Deep Embedding Features for Action Recognition on Raw Depth Maps

Jacek Trelinski and Bogdan Kwolek

AGH University of Science and Technology
30 Mickiewicza, 30-059 Krakow, Poland
{tjacek,bkw}@agh.edu.pl,

Abstract. In this paper we present an approach for embedding features for action recognition on raw depth maps. Our approach demonstrates high potential when amount of training data is small. A convolutional autoencoder is trained to learn embedded features, encapsulating the content of single depth maps. Afterwards, multichannel 1D CNN features are extracted on multivariate time-series of such embedded features to represent actions on depth map sequences. In the second stream the dynamic time warping is used to extract action features on multivariate streams of statistical features from single depth maps. The output of the third stream are class-specific action features extracted by TimeDistributed and LSTM layers. The action recognition is achieved by voting in an ensemble of one-vs-all weak classifiers. We demonstrate experimentally that the proposed algorithm achieves competitive results on UTD-MHAD dataset and outperforms by a large margin the best algorithms on 3D Human-Object Interaction Set (SYSU 3DHOI).

Keywords: Data scarcity, convolutional neural networks, feature embedding.

1 Introduction

People have an innate tendency to recognize and even predict other people's intentions based on their actions [1] and understanding actions and intentions of other people is one of most vital social skills we have [2]. In recent years, deep learning-based algorithms have shown high potential in modeling high-level abstractions from intricate data in many areas such as natural language processing, speech processing and computer vision [3]. After seminal works [4,5] that showed potential and effectiveness of deep learning in human activity recognition, many related studies have been published in this area [6,7]. Most of the present state-of-the-art methods for action recognition either aims at improving the recognition performance through modifications of the backbone CNN network, or they investigate different trade-offs between computational efficiency and performance, c.f. work done in Amazon [8]. However, while deep learning-based algorithms have achieved remarkable results, putting this technology into practice can be difficult in many applications for human activity analysis because training deep models

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9

requires large datasets and specialized and energy-intensive equipment. In order to cope with such challenges, massively parallel processing capabilities offered by photonic architectures were investigated recently to achieve energy-efficient solutions for real-time action recognition [9].

Several recent methods treat the problem of human action recognition as a generic classification task and try to transfer best practice from ImageNet classification with difference that the input are frame sequences instead of single frames. However, human activities are complex, ambiguous, have different levels of granularity and differ in realization by individuals, including action dynamics. Difficulties in recognition involve many factors such as non-rigid shape of humans, temporal structure, body movement, and human-object interaction, etc. Due to such factors, environmental complexities and plenty another challenges, current algorithms have poor performance in comparison to human ability to recognize human motions and actions [10,11]. As shown in a recent study [12], humans predict actions using grammar-like structures, and this may be one of the reasons of not sufficient recognition performance of current end-to-end approaches that neglect such factors. Moreover, as showed in the discussed study, losing time-information is a feature that can help grouping actions together in the right way. One of the important conclusions of this work is that time may rather confuse than help in recognition and prediction.

3D-based approaches to human action recognition provide higher accuracy than 2D-based ones. Most of the present approaches to action recognition on depth maps are based on the skeleton [13,14]. The number of approaches based on depth maps only, particularly deep learning-based is very limited [11]. One reason of lower interest on such research direction is that depth data is difficult as well as the presence of noise in raw depth map sequences. Despite that skeleton-based methods usually achieve better results than algorithms using only depth maps, they can fail in many scenarios due to skeleton extraction failure. Moreover, in scenarios involving interaction with objects, where detection of objects shapes, 6D poses, etc., is essential, skeleton only-based methods can be less useful. Depth maps acquired from wall-mounted or ceiling-mounted sensors permit accurate detection of patient mobility activities and their duration in intensive care units [15] as well as events like human falls [16].

Traditional approaches to activity recognition on depth maps rely on the handcrafted feature-based representations [17,18]. In contrast to handcrafted representation-based approaches, in which actions are represented by engineered features, learning-based algorithms are capable of discovering the most informative features automatically from raw data. Such deep learning-based methods permit processing images/videos in their raw forms and thus they are capable of automating the process of feature extraction and classification. These methods employ trainable feature extractors and computational models with multiple layers for action representation and recognition.

In this work we propose an approach that, despite limited amount of data, permits achieving high classification scores in action recognition on the basis of raw depth data. To cope with limited and difficult data for learning the ac-

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9

tion classifier we utilize multi-stream features, which are extracted using DTW, TimeDistributed and LSTM layers (TD-LSTM), and convolutional autoencoder followed by a multi-channel, temporal CNN (1D-CNN). In order to improve model uncertainty the final decision is taken on the basis of several models that are simpler but more robust to the specifics of noisy data sequences.

2 The Algorithm

A characteristic feature of the proposed approach is that it does not require skeleton. Thanks to using depth maps only, our algorithm can be employed on depth data provided by stereo cameras, which can deliver the depth data for persons being at larger distances to the sensors. It is well known that the Kinect sensor fails to estimate the skeleton in several scenarios. In the next Section, we demonstrate experimentally that despite no use of the skeleton, our algorithm achieves better accuracies than several skeleton-based algorithms. In the proposed approach, various features are learned in different domains, like single depth map, time-series of embedded features, time-series warped by DTW (dynamic time warping), and final decision is taken on the basis of voting of one-vs-all weak classifiers. In the proposed approach multi-stream features are processed to extract action features in sequences of depth maps. Action features are extracted using DTW, TimeDistributed and LSTM layers (TD-LSTM), and convolutional autoencoder followed by a multi-channel, temporal CNN (1D-CNN). In consequence, to cope with variability in the observations as well as limited training data, particularly in order to improve model uncertainty the final decision is taken on the basis of several models that are simpler but more robust to the specifics of the noisy data sequences.

The algorithm was evaluated on UTD-MHAD and SYSU 3DHOI datasets. Since in SYSU 3DHOI dataset the performers are not extracted from depth maps, we extracted the subjects. For each depth map we determined a window surrounding the person, which has then been scaled to the required input shape.

In Subsection 2.1 we present features describing the person's shape in single depth maps. Afterwards, in Subsection 2.2 we outline features representing multivariate time-series. Then, in Subsection 2.3 we detail embedding actions using neural network with TimeDistributed and LSTM layers. In Subsection 2.4 we discuss multi-class classifiers to construct ensemble. Finally, in Subsection 2.5 we describe the ensemble as well as our algorithm that for each classified action determines classifiers for voting.

2.1 Embedding Action Features Using CAE and Multi-channel, Temporal CNN

Embedding Frame-Features. Since current datasets for depth-based action recognition have insufficient number of sequences to learn deep models with adequate generalization capabilities, we utilize a convolutional autoencoder (CAE)

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9

operating on single depth maps to extract informative frame-features. Time-series of such features representing actions in frame sequences are then fed to multi-channel, temporal CNN that is responsible for extraction embedded features. Because the number of frames in the current benchmark datasets for RGB-D-based action recognition is pretty large, deep feature representations can be learned. Given an input depth map sequence $x = \{x_1, x_2, \ldots, x_T\}$, we encode each depth map x_i using a CNN backbone f into a feature $f(x_i)$, which results in a sequence of embedded feature vectors $f(x) = \{f(x_1), f(x_2), \ldots, f(x_T)\}$. The dimension of such embedding for a depth map sequence is $T \times D_f$, where D_f is size of the embedded vector.

An autoencoder is a type of neural network that projects a high-dimensional input into a latent low-dimensional code (encoder), and then carries out a reconstruction of the input using such a latent code (the decoder) [19]. To achieve this the autoencoder learns a hidden representation for a set of input data, by learning how to ignore less informative information. This means that the autoencoder tries to generate from such a reduced encoding an output representation that is close as possible to its input. When the hidden representation uses fewer dimensions than the input, the encoder carries out dimensionality reduction. An autoencoder consists of an internal (hidden) layer that stores a compressed representation of the input, as well as an encoder that maps the input into the code, and a decoder that maps the code to a reconstruction of the original input. The encoder compresses the input and produces the code, whereas the decoder reconstructs the input using only this code. Learning to replicate its input at its output is achieved by learning a reduction side and a reconstructing side. Autoencoders are considered as unsupervised learning technique since no explicit labels are needed to train them. Once such a representation with reduced dimensionality is learned, it can then be taken as input to a supervised algorithm that can then be trained on the basis of a smaller labeled data subset.

We extracted frame-features using encoder/decoder paradigm proposed in [20]. We implemented a convolutional autoencoder in which the input depth map is first transformed into a lower dimensional representation through successive convolution operations and rectified linear unit (ReLU) activations and afterwards expanded back to its original size using deconvolution operations. The mean squared error, which measures how close the reconstructed input is to the original input has been used as the loss function in the unsupervised learning. The network has been trained using Adam optimizer with learning rate set to 0.001. After training, the decoding layers of the network were excluded from the convolutional autoencoder. The network trained in such a way has been used to extract low dimensional frame-features. The depth maps acquired by the sensor were projected two 2D orthogonal Cartesian planes to represent top and side view of the maps. On training subsets we trained a single CAE for all classes. The convolutional autoencoder has been trained on depth maps of size $3 \times 64 \times 64$. The CAE network architecture is shown in Fig. 1. The network consists of two encoding layers and two associated decoding layers. The size of depth map embedding is equal to 100.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9

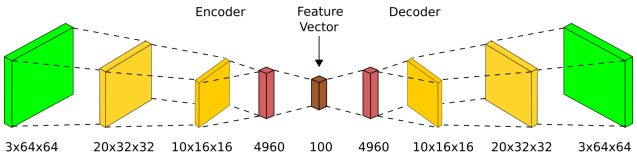

Fig. 1. Architecture of convolutional autoencoder.

Features of Time-series. Embedding Action Features Using Multi-channel, Temporal CNN. On the basis of depth map sequences representing human actions the CAE that was discussed above produces multivariate time-series. Having on regard that depth map sequences differ in length, such variable length time-series were interpolated to a common length. In multi-channel, temporal CNNs (MC CNNs) the 1D convolutions are applied in the temporal domain. In this work, the time-series (TS) of frame-features that were extracted by the CAE have been used to train a multi-channel 1D CNN. The number of channels is equal to 100, see Fig. 1. The multivariate time-series were interpolated to the length equal to 64. Cubic-spline algorithm has been utilized to interpolate the TS to such a common length.

The first layer of the MC CNN is a filter (feature detector) operating in time domain. Having on regard that the amount of the training data in current datasets for depth-based action recognition is quite small, the neural network consists of two convolutional layers, each with 8×1 filter, 4×1 and 2×1 max pools, and strides set to 1 with no padding, respectively, see Fig. 2. The number of neurons in the dense layer is equal to 100. The number of output neurons is equal to number of the classes. Nesterov Accelerated Gradient (Nesterov Momentum) has been used to train the network, in 1000 iterations, with momentum set to 0.9, dropout equal to 0.5, learning rate equal to 0.001, and L1 parameter set to 0.001. After the training, the output of the dense layer has been used to embed the features, which are referred to as 1D-CNN features.

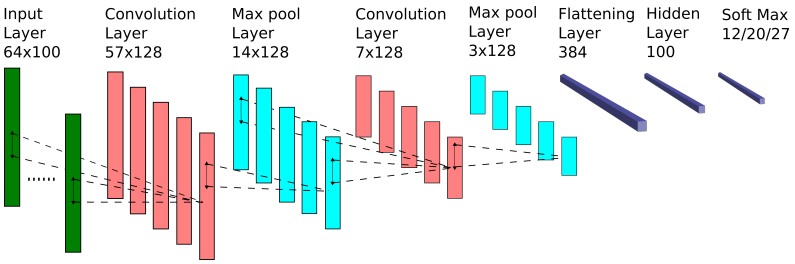

Fig. 2. Flowchart of the multi-channel CNN for multivariate time-series modeling.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9

2.2 DTW-based Action Features

Frame-Feature Vector. For each depth frame we calculate also handcrafted features describing the person's shape. Similarly to learned frame-features that have been described in Subsection 2.1, we project the acquired depth maps onto three orthogonal Cartesian views to capture the 3D shape and motion information of human actions. Only pixels representing the extracted person in depth maps are utilized for calculating the features. The following vectors of frame-features were calculated on such depth maps:

1. correlation (xy, xz and zy axes),
2. $x-$coordinate for which the corresponding depth value represents the closest pixel to the camera, $y-$coordinate for which the corresponding depth value represents the closest pixel to the camera.

This means that the person shape in each depth map is described by 3 and 2 features, respectively, depending on the chosen feature set. A human action represented by a number of depth maps is described by a multivariate time-series of length equal to number of frames and dimension 2 or 3 in dependence on the chosen feature set.

DTW-based Features. Dynamic time warping (DTW) is an effective algorithm for measuring similarity between two temporal sequences, which may vary in speed and length. It calculates an optimal match between two given sequences, e.g. time series [21]. In time-series classification one of the most effective algorithms is 1-NN-DTW, which is a special k-nearest neighbor classifier with $k = 1$ and a dynamic time warping for distance measurement. In DTW the sequences are warped non-linearly in time dimension to determine the best match between two samples such that when the same pattern exists in both sequences, the distance is smaller. Let us denote $D(i, j)$ as the DTW distance between subsequences $x[1 : j]$ and $y[1 : j]$. Then the DTW distance between x and y can be determined by the dynamic programming algorithm according to the following iterative equation:

$$D(i, j) = \min\{D(i-1, j-1), D(i-1, j), D(i, j-1)\} + |x_i, y_j| \qquad (1)$$

The time complexity of calculation of DTW distance is $O(nm)$, where n and m are the length of x and y, respectively.

We calculate the DTW distance between all depth maps sequences in the training subset. For each depth map sequence the DTW distances between multivariate time-series were calculated for the feature sets 1 and 2. The DTW distances between a given sequence and all remaining sequences from the training set were then used as features. This means that the resulting feature vector has size $n_t \times 2$, where n_t denotes the number of training depth map sequences. The DTW distances have been calculated using library [22].

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9

2.3 Embedding Actions Using Neural Network Consisting of TimeDistributed and LSTM Layers

The neural network operates on depth map sequences, where each sample has a 64×64 data format, across 30 time-steps. The frame batches of size 30 were constructed by sampling with replacement. In first three layers we employ TimeDistributed wrapper to apply the same Conv2D layer to each of the 30 time-steps, independently. The first TimeDistributed layer wraps 32 convolutional filters of size 5×5, with padding set to 'same'. The second TimeDistributed layer wraps 32 convolutional filters of size 5×5. The third TimeDistributed layer wraps the max pooling layer in window of size 4×4. Afterwards, TimeDistributed layer wraps the flattening layer. Next, two TimeDistributed layers wrap dense layers with 256 and 128 neurons, respectively. At this stage the output shape is equal to (None, 30, 128). Finally, we utilize 64 LSTMs and then 64 global average pooling filters, see Fig. 3. The resulting features are called TD-LSTM. The neural networks have been trained using adadelta with learning rate set to 0.001. The loss function was categorical crossentropy and the models were trained as one-vs-all. The motivation of choosing such approach is due to redundant depth maps, i.e. the same human poses in different actions.

2.4 Multi-class Classifiers to Construct Ensemble

The features described in Subsections 2.1 –2.3 were used to train multi-class classifiers with softmax encoding, see Fig. 3. Having on regard that for each class an action-specific classifier to extract depth map features has been trained, the number of such classifiers is equal to the number of actions to be recognized. The convolutional autoencoder operating on sequences of depth maps delivers time-series of CAE-based frame-features, on which we determine 1D-CNN features (Subsect. 2.1). Similarly to features mentioned above, the DTW-based features (Subsect. 2.2) are also common features for all classes. The base networks of TimeDistributed-LSTM network (Subsect. 2.3) operating on sequences of depth maps deliver class-specific action features. The discussed TD-LSTM features are of size 64, see Fig. 3, and they are then concatenated with action features mentioned above. The multi-class classifiers delivering at the outputs the softmax-encoded class probability distributions are finally used in an ensemble responsible for classification of actions.

2.5 Ensemble of Classifiers

Figure 3 depicts the ensemble for action classification. The final decision is calculated on the basis of voting of the classifiers. In essence, the final decision is taken using an ensemble of individual models. One advantage of this approach is its interpretability. Because each class is expressed by one classifier only, it is possible to gain knowledge about the discriminative power of individual classifiers. As we can see, for each class the action features that are common for

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9

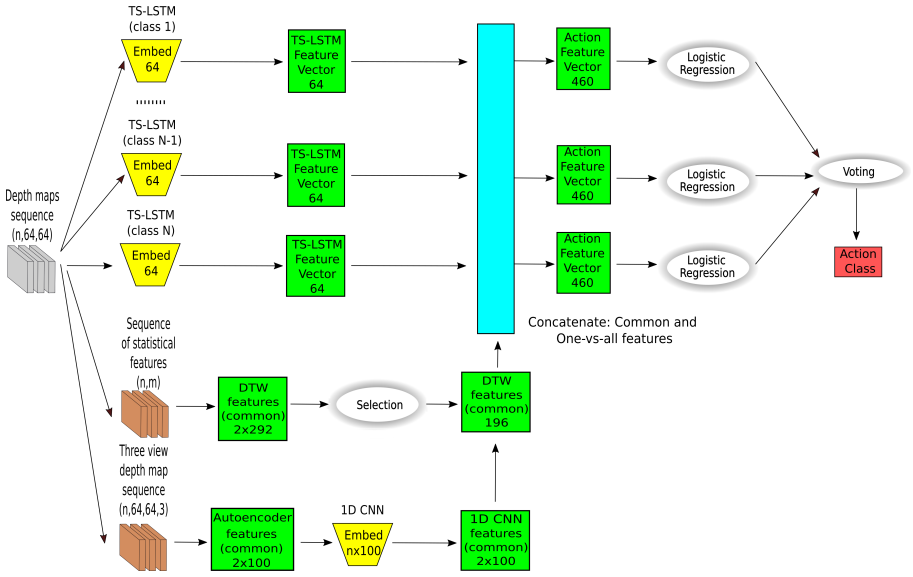

Fig. 3. Ensemble operating on features extracted by DTW, features embedded by CAE and 1D-CNN, which are then concatenated with class-specific features that are embedded by TimeDistributed and LSTM neural networks.

all actions are concatenated with class-specific features, and then used to train multi-class classifiers.

Having on regard that not all classifiers do not contribute equally in decision making we selected the classifiers individually for each testing example. During the classification of each action we initially perform a classification using all classifiers trained in advance and additionally we employ a k-NN classifier. The k-NN classifiers operate only on TD-LSTM features, whereas the logistic regression (LR) classifiers operate on the concatenated features. We consider each class-specific LR classifier with corresponding k-NN with k set to three and inspect their decisions. If decisions of both classifiers are the same then the LR classifier will take in the final voting about the action category. In a case when less than three LR classifiers were selected for the final voting then all LR classifiers attend in the final voting. The discussed algorithm has been compared with an algorithm based on differential evolution (DE), which is responsible for determining the weights for the soft voting.

3 Experimental Results

The proposed algorithm has been evaluated on two publicly available benchmark datasets: UTD-MHAD dataset [23] and SYSU 3D Human-Object Interaction Set (SYSU 3DHOI) [24]. The datasets were selected having on regard their

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9

frequent use by action recognition community in the evaluations and algorithm comparisons.

The UTD-MHAD dataset contains 27 different actions performed by eight subjects (four females and four males). All actions were performed in an indoor environment with a fixed background. Each performer repeated each action four times. The dataset consists of 861 data sequences and it was acquired using the Kinect sensor and a wearable inertial sensor.

The SYSU 3D Human-Object Interaction (3DHOI) dataset was recorded by the Kinect sensor and comprises 480 RGB-D sequences from 12 action classes, including calling with cell phone, playing with a cell phone, pouring, drinking, wearing backpack, packing a backpack, sitting on a chair, moving a chair, taking something from a wallet, taking out a wallet, mopping and sweeping. Actions were performed by 40 subjects. Each action involves a kind of human-object interactions. Some motion actions are quite similar at the beginning since the subjects operate or interact with the same objects, or actions start with the same sub-action, such as standing still. The above mentioned issues make this dataset challenging following the evaluation setting in [25], in which depth map sequences with the first 20 subjects were used for training and the rest for testing.

Table 1 presents experimental results that were achieved on the UTD-MHAD dataset. As we can observe, the ensemble consisting of weak classifiers operating on only one-vs-all features, which were embedded using the LSTMs achieves relatively low accuracy in comparison to remaining results, i.e. the recognition performances in row #3 are lower than remaining performances. The DTW features if used alone or when combined with the features embedded by the LSTMs permit achieving better results in comparison to results presented in row #3. The features embedded by CAE and 1D-CNN, see results in first row, permit to achieve better results in comparison to results, which we discussed above. Concatenating the above mentioned features with the features embedded by LSTMs leads to slightly better results, cf. results in the first and fourth row. The best results were achieved by the ensemble consisting of weak classifiers operating on one-vs-all features (LSTM-based), concatenated with features embedded by CAE and 1D-CNN, and concatenated with DTW features. Although the features

Table 1. Recognition performance on UTD-MHAD dataset.

common	one-vs-all	Accuracy	Precision	Recall	F1-score
1D-CNN	-	0.8558	0.8593	0.8558	0.8474
DTW	-	0.7930	0.8096	0.7930	0.7919
-	TD-LSTM	0.6419	0.6833	0.6419	0.6322
1D-CNN	TD-LSTM	0.8581	0.8649	0.8581	0.8504
DTW	TD-LSTM	0.8256	0.8455	0.8256	0.8242
DTW 1D-CNN	TD-LSTM	**0.8814**	**0.8844**	**0.8814**	**0.8747**

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9

embedded by LSTMs achieve relatively poor results, when combined with the other features they improve the recognition accuracy significantly. The discussed results were achieved by the logistic regression classifiers. They were obtained on the basis of soft voting in the ensemble, which gave slightly better results in comparison to hard voting. Logistic regression returns well calibrated predictions by default as it directly optimizes the Log loss and therefore it has been chosen to built the ensemble. Figure 4 depicts the confusion matrix for the best results achieved on the discussed dataset.

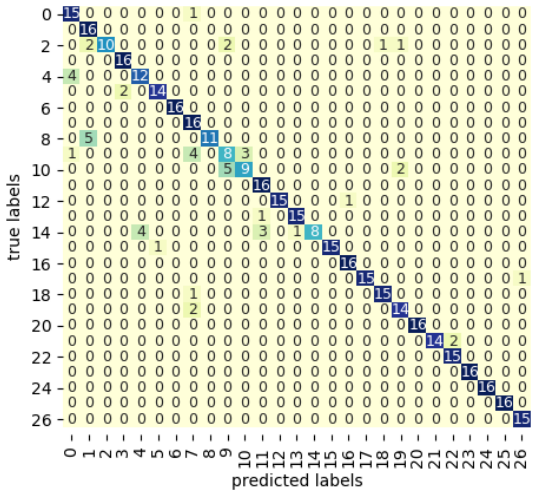

Fig. 4. Confusion matrix on UTD-MHAD dataset.

Table 2 presents experimental results that were achieved using feature selection. As we can notice, our feature selection algorithm permits achieving better results in comparison to results shown in Tab. 1. The Differential Evolution allows achieving the best classification performance.

Table 2. Recognition performance on UTD-MHAD dataset with selected classifiers for voting.

	voting using selected classifiers				differential evolution(DE)			
common	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
-	0.6535	0.6992	0.6535	0.6426	0.6721	0.7112	0.6721	0.6601
TD-LSTM	0.8628	0.8688	0.8628	0.8548	0.8581	0.8649	0.8581	0.8504
DTW	0.8326	0.8557	0.8326	0.8304	0.8302	0.8497	0.8302	0.8279
DTW TD-LSTM	0.8860	0.8883	0.8860	0.8784	**0.8907**	**0.8919**	**0.8907**	**0.8833**

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9

Table 3 presents the recognition performance of the proposed method compared with previous methods. Most of current methods for action recognition on UTD-MHAD dataset are based on skeleton data. Methods based on skeleton modality usually achieve better results in comparison to methods relying on depth data only. Despite the fact that our method is based on depth modality, we evoked the recent skeleton-based methods to show that it outperforms many of them.

Table 3. Comparative recognition performance of the proposed method with recent algorithms on MHAD dataset.

Method	Modality	Accuracy [%]
JTM [26]	skeleton	85.81
SOS [27]	skeleton	86.97
Kinect & inertial [23]	skeleton	79.10
Struct. SzDDI [28]	skeleton	89.04
WHDMMs+ConvNets [29][28]	depth	73.95
Proposed method	depth	**89.07**

Table 4 illustrates results that were achieved on the 3DHOI dataset. As we can observe, the ensemble consisting of weak classifiers operating on only one-vs-all features, which were embedded using the LSTMs achieves comparable results with results that were obtained using DTW features, and whose performances are lower in comparison to remaining results. Combining DTW features with features embedded by LSTMs leads to better results in comparison to results achieved using only features embedded by LSTMs, compare results in row #5 with results in row #3. The features embedded by CAE and 1D-CNN, see results in first row, permit to achieve better results in comparison to results, which we discussed above. Combining features embedded by CAE and 1D-CNN with features embedded by LSTMs leads to further improvement of the recognition performance, see results in row #4. The best results were achieved by the ensemble consisting of weak classifiers operating on one-vs-all features (LSTM-based) concatenated with features embedded by CAE and 1D-CNN, and concatenated with DTW features. The discussed results were achieved by the logistic regression classifiers. They were obtained on the basis of soft voting in the ensemble, which gave slightly better results in comparison to hard voting. Figure 5 illustrates the confusion matrix.

Table 5 presents results that were obtained using feature selection. As we can observe, both our algorithm and DE improve results presented in Tab. 4. Results achieved by our algorithm are superior in comparison to results achieved by differential evolution.

Table 6 presents results achieved by recent algorithms on 3DHOI dataset in comparison to results achieved by our algorithm. As we can observe, our algorithm achieves the best results on this challenging dataset. It is worth noting that

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9

Table 4. Recognition performance on SYSU 3DHOI dataset.

common	one-vs-all	Accuracy	Precision	Recall	F1-score
1D-CNN	-	0.8114	0.8197	0.8114	0.8104
DTW	-	0.4781	0.4889	0.4781	0.4546
-	TD-LSTM	0.4781	0.4800	0.4781	0.4627
1D-CNN	TD-LSTM	0.8553	0.8591	0.8553	0.8550
DTW	TD-LSTM	0.5044	0.5318	0.5044	0.4872
DTW 1D-CNN	TD-LSTM	**0.8947**	**0.8953**	**0.8947**	**0.8941**

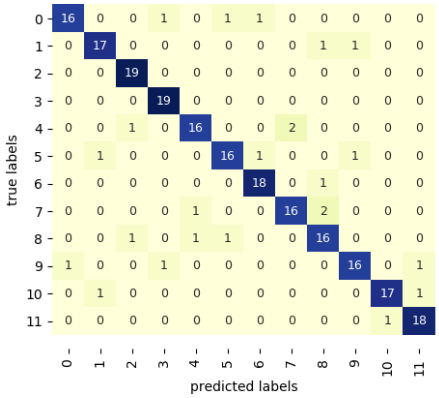

Fig. 5. Confusion matrix on 3DHOI dataset.

Table 5. Recognition performance on 3D HOI dataset with selected classifiers for voting.

	voting using selected classifiers				differential evolution(DE)			
common	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
-	0.5482	0.5521	0.5482	0.5318	0.5263	0.5325	0.5263	0.5140
1D-CNN	0.8640	0.8709	0.8640	0.8638	0.8684	0.8743	0.8684	0.8688
DTW	0.5351	0.5619	0.5351	0.5151	0.5351	0.5602	0.5351	0.5172
DTW 1D-CNN	**0.9035**	**0.9033**	**0.9035**	**0.9024**	0.8904	0.8909	0.8904	0.8895

method [30] relies on depth and skeleton modalities, whereas [25] additionally utilizes RGB images jointly with the skeleton data.

4 Conclusions

In this paper we presented an approach to encapsulate the content of raw depth maps sequences with human actions. The algorithm has been designed to recog-

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9

Table 6. Comparative recognition performance of the proposed method with recent algorithms on 3DHOI dataset.

Method	Modality	Acc. [%]
MSRNN [25]	depth+RGB+skel.	79.58
PTS [30]	depth+skeleton	87.92
Proposed method	depth	**90.35**

nize actions in scenarios when amount of training data is small. It achieves considerable gain in action recognition accuracy on challenging 3D Human-Object Interaction Set (SYSU 3DHOI). On UTD-MHAD dataset it outperforms recent methods on raw depth maps and outperforms most recent methods on skeleton data. The novelty of the proposed method lies in multi-stream features, which are extracted using dynamic time warping, TimeDistributed and LSTM layers, and convolutional autoencoder followed by a multi-channel, temporal CNN. The main methodological results show that despite data scarcity the proposed approach builds classifiers that are able to cope with difficult data and outperforms all the other methods in terms of accuracy.

Acknowledgment. This work was supported by Polish National Science Center (NCN) under a research grant 2017/27/B/ST6/01743.

References

1. Blakemore, S.J., Decety, J.: From the perception of action to the understanding of intention. Nature Reviews Neuroscience **2**(8) (2001) 561–567
2. Blake, R., Shiffrar, M.: Perception of human motion. Annual Review of Psychology **58**(1) (2007) 47–73
3. Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M.P., Shyu, M.L., Chen, S.C., Iyengar, S.S.: A survey on deep learning: Algorithms, techniques, and applications. ACM Computing Surveys **51**(5) (2018)
4. Yang, J.B., Nguyen, M.N., San, P., Li, X.L., Krishnaswamy, S.: Deep convolutional neural networks on multichannel time series for human activity recognition. In: Proc. of the 24th Int. Conf. on Artificial Intelligence, AAAI Press (2015) 3995–4001
5. Lane, N.D., Georgiev, P.: Can deep learning revolutionize mobile sensing? In: 16th Int. Workshop on Mobile Comp. Syst. and Appl., ACM (2015) 117–122
6. Beddiar, D.R., Nini, B., Sabokrou, M., Hadid, A.: Vision-based human activity recognition: A survey. Multimedia Tools and Appl. **79**(41) (2020) 30509–30555
7. Majumder, S., Kehtarnavaz, N.: Vision and inertial sensing fusion for human action recognition: A review. IEEE Sensors Journal **21**(3) (2021) 2454–2467
8. Martinez, B., Modolo, D., Xiong, Y., Tighe, J.: Action recognition with spatial-temporal discriminative filter banks. In: IEEE/CVF Int. Conf. on Computer Vision (ICCV), IEEE Computer Society (2019) 5481–5490
9. Antonik, P., Marsal, N., Brunner, D., Rontani, D.: Human action recognition with a large-scale brain-inspired photonic computer. Nature Machine Intelligence **1**(11) (2019) 530–537

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9

10. Liang, B., Zheng, L.: A survey on human action recognition using depth sensors. In: Int. Conf. on Digital Image Comp.: Techn. and Appl. (2015) 1–8
11. Wang, L., Huynh, D.Q., Koniusz, P.: A comparative review of recent Kinect-based action recognition algorithms. IEEE Trans. Image Process. **29** (2020) 15–28
12. Wörgötter, F., Ziaeetabar, F., Pfeiffer, S., Kaya, O., Kulvicius, T., Tamosiunaite, M.: Humans predict action using grammar-like structures. Scientific Reports **10**(1) (2020) 3999
13. Ali, H.H., Moftah, H.M., Youssif, A.A.: Depth-based human activity recognition: A comparative perspective study on feature extraction. Future Computing and Informatics Journal **3**(1) (2018) 51 – 67
14. Ren, B., Liu, M., Ding, R., Liu, H.: A survey on 3D skeleton-based action recognition using learning method. arXiv, 2002.05907 (2020)
15. Yeung, S., Rinaldo, F., Jopling, J., Liu, B., Mehra, R., Downing, N.L., Guo, M., Bianconi, G.M., Alahi, A., Lee, J., Campbell, B., Deru, K., Beninati, W., Fei-Fei, L., Milstein, A.: A computer vision system for deep learning-based detection of patient mobilization activities in the ICU. npj Digital Medicine **2**(1) (2019)
16. Haque, A., Milstein, A., Fei-Fei, L.: Illuminating the dark spaces of healthcare with ambient intelligence. Nature **585**(7824) (2020) 193–202
17. Yang, X., Zhang, C., Tian, Y.L.: Recognizing actions using depth motion maps-based histograms of oriented gradients. In: Proc. of the 20th ACM Int. Conf. on Multimedia, ACM (2012) 1057–1060
18. Xia, L., Aggarwal, J.: Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera. In: CVPR. (2013) 2834–2841
19. Hinton, G., Salakhutdinov, R.: Reducing the dimensionality of data with neural networks. Science **313**(5786) (2006) 504 – 507
20. Masci, J., Meier, U., Cireşan, D., Schmidhuber, J.: Stacked convolutional auto-encoders for hierarchical feature extraction. In: ICANN, vol. I. (2011) 52–59
21. Paliwal, K., Agarwal, A., Sinha, S.: A modification over Sakoe and Chiba's dynamic time warping algorithm for isolated word recognition. Signal Proc. **4**(4) (1982) 329 – 333
22. Meert, W., Hendrickx, K., Craenendonck, T.V.: DTAIdistance, ver. 2.0. `https://zenodo.org/record/3981067` (2021)
23. Chen, C., Jafari, R., Kehtarnavaz, N.: UTD-MHAD: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor. In: IEEE ICIP. (2015) 168–172
24. Hu, J., Zheng, W., Lai, J., Zhang, J.: Jointly learning heterogeneous features for RGB-D activity recognition. In: CVPR. (2015) 5344–5352
25. Hu, J., Zheng, W., Ma, L., Wang, G., Lai, J., Zhang, J.: Early action prediction by soft regression. IEEE Trans. PAMI **41**(11) (2019) 2568–2583
26. Wang, P., Li, W., Li, C., Hou, Y.: Action recognition based on joint trajectory maps with convolutional neural networks. Knowledge-Based Syst. **158** (2018) 43 – 53
27. Hou, Y., Li, Z., Wang, P., Li, W.: Skeleton optical spectra-based action recognition using convolutional neural networks. IEEE Trans. CSVT **28**(3) (2018) 807–811
28. Wang, P., Wang, S., Gao, Z., Hou, Y., Li, W.: Structured images for RGB-D action recognition. In: ICCV Workshops. (2017) 1005–1014
29. Wang, P., Li, W., Gao, Z., Zhang, J., Tang, C., Ogunbona, P.: Action recognition from depth maps using deep convolutional neural networks. IEEE Trans. on Human-Machine Systems **46**(4) (2016) 498–509
30. Wang, X., Hu, J.F., Lai, J.H., Zhang, J., Zheng, W.S.: Progressive teacher-student learning for early action prediction. In: CVPR. (2019) 3551–3560

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77967-2_9