**Skeleton-Based Action Recognition Using Few Shot Learning**

**Wangbin Ding1,2, Shengqin Lin2, Zhenze Dai2**

1Shanghai JJ Software Technology Co 2Fuzhou Institute of Technology

Abstract

This paper presents a new skeleton-based action recognition method using a few shot learning. Consider the sequence data contain the temporal and spatial information; the proposed method encodes each of skeleton as an RGB image. Nothing more but a naïve normalization is engaged to each channel of the encoded skeleton image. In order to acquire the discriminative skeleton image feature, a serial of dilated-dense layers is adopted in our model to both extend the receptive field of feature points and capture diversity representation of the skeleton image. After that, a prototypical network is introduced to recognize the specific action of the feature stands for. The skeleton image feature will be mapped into a metric space in which action classification can be performed by the nearest neighbor search. Benefited from the nature of the few shot learning, our model can be trained with only a few labeled samples. Moreover, it could deal with the samples from unseen classes that have not presented during the training phase. We evaluated our method with the seen and unseen class of sample, experiment result shows, the method achieved state-of-the-art performance on benchmark datasets.

**Index Terms:** skeleton sequence, dilated-dense layer, few shot learning

# **Introduction**

Human action recognition has been widely researched for a few decades. A lot of recognition methods are developed to serve for entertainment, surveillance, and video analysis. At present, the action recognition algorithm has made a great step forward by the wave of the deep learning model. However, these methods always consume large-scale datasets during the training period. As human actions are varied, the demand for train sample will explode. Besides, when samples with unseen labels acquired, the deep model needs to be re-trained to fit the changing distribution of the data label. A sort of few-shot (one-shot, zero-shot) learning methods are proposed to resolve these issues. Non-parameter model from nearest neighbor to metric learning [13] have played a significant role in the progress of this field. Mishra [14] present a generative framework for Zero-Shot or Few-Shot action recognition. Yang [15] introduce a new example-based action detection on the Matching Network. Nevertheless, seldom attention had paid to apply the few shot learning model to skeleton sequence yet. To address this challenge, we propose an action recognition method based on the few shot learning. Because we had engaged a serial of CNN layers to extract the feature of skeleton sequence, our method also can benefit from the performance of CNN. And The main contributions of this method are：

1. Only a few skeleton sequences are adequate to train an efficient action recognition model.
2. With few support samples, it is enough to recognize the action that had never seen before.
3. Dilated-dense layer is embedded to extract the feature maps, which could enhance the robustness and diversity(dense) of the feature representation.

The remainder of the paper is organized as follows. In Sec.2 we briefly review methods proposed to deal with skeleton representation. In Sec.3 the representation of the skeleton using dilated-dense layer is introduced. Then the inference and training algorithm of our few shot learning based model will be fully described. In Sec.4 we report the experiments results on a series of datasets to show the performance of the method. Finally, in Sec.5 we discuss research directions in the future work.

# **Related Work**

From the perspective of feature representation, skeleton-based action recognition methods can be divided into three main streams.

The first streams usually encode the skeleton sequence into a skeleton-image. Different skeleton-image encoders are proposed to capture the feature of actions. Wang[1] introduce a compact and effective method to encode spatiotemporal information carried in 3D skeleton sequences into Joint Trajectory Maps (JTM). Pham[2] and Li[3] rearrange the pixels in RGB skeleton-images to obtain a better representation of the movement. Furthermore, Liu[4] design a skeleton visualization method to represent a skeleton sequence as a series of visual and motion enhanced color images. Having encoded skeleton sequences into skeleton images, a variety of CNN networks(CNN[1], multi-scale CNN[5], DeepResidualNeuralNetworks[2], multi-stream CNN[4]) are constructed to classify the indeed action of the skeleton image.

The second steam is inspired by the RNN network, whose recurrent structure can boil a sequence data down into a high-level understanding []. For better performance, they tend to adopt the LSTM to process the skeleton sequence data. Zhu [6] introduce an end-to-end fully connected deep LSTM network with a designed regularization, through which can learn the co-occurrence feature of the skeleton joints. Based on LSTM, Liu [7] design a skeleton tree traversal method and a new gating mechanism to achieve a robust representation of the input sequence data. To further, Liu [8] also add attention ability to the LSTM network, which is capable of focusing on the informative joints of the skeleton.

The last one is based on graph convolutional network [9], which can be applied directly to the raw skeleton data. Shi [10] present a novel two-stream nonlocal graph convolutional network for the recognition task. Li [11] introduce multi-scale graphical convolutional kernels to encode motion variations and input state for feature extracting. Si [12] propose a novel Attention Enhanced Graph Convolutional LSTM Network, which can not only capture discriminative features in spatial configuration and temporal dynamics but also explore the co-occurrence relationship between spatial and temporal domains.

# Proposed Method

## Skeleton Image Encoder

It is common knowledge that a skeleton sequence can be represented as an RGB image []. Consider an frame action , each frame contains joints . And each joint is a 3-D coordinate point, which is corresponding to the RGB channel of a pixel in a skeleton image. Thus, the skeleton sequence can be encoded as an skeleton image. a sort of variant RGB encoder had proposed to achieve the translation-scale invariant representation of the skeleton sequences [4 16,17]. But our method hasn’t taken advantage of these mechanisms. Only a naive normalization, proposed by [18], is adopted.

Where and are the maximum and minimum value of the k-th channel (x; y; z) of a skeleton. In this way, each action sample can be encoded into a skeleton image .



Figure 1: the workflow of extracting a skeleton image from a skeleton sequence

## Prototypical Network

A prototypical network learns a metric space in which classification can be performed by computing distances to prototype representations of each class [19]. Given K classes of labeled support sets, we define the k-th support set as：

The support set will be mapped by a function and the prototype of k-th class is the mean of the mapped support samples that belong to it.

When feeding an unlabeled skeleton image , the class of the image is decided by the nearest neighbour prototype.

## Dilated-Dense Layer

The learnable part of our model is the mapping function . For the sake of processing the RGB skeleton image, we construct it as a CNN network. There are two things that we considered before build the network. On is that, unlike natural image, each pixel scattered in the image has equally interpretive meaning for the final action decision, we trend to expands the size of the convolution kernel, so that is could convolve as much as pixels in the skeleton image. The other one is that, both the movement of the skeleton joints between different frames and the distribution of the joints in spatial space are the significate features of action. For these reasons, we considered adding dilated-dense layers, whose dilated convolution kernel could enlarge the receptive field of the feature point when constructing the CNN network. Besides, the full connectivity of dense layers could lead to a variety of joints and movement features descriptors.

C:\Users\Administrator\PycharmProjects\human_action_recognition\data\Skeleton2\MSRAction3DSkeleton(20joints)\y_a04_s06_e03_skeleton.txt.bmp 

Figure Each pixel in skeleton image represent a skeleton joint, even the pixel around the corner has equally importance to the one located in the center.



Figure 2: CNN architecture of the mapping function : Three dilated-desnse layers is introduced to enhance the robust representation of the skeleton image. The parameter of the mapping function is corresponding to the weights and biases of the CNN network.

## Inference Phase

The output of the prototypical network is SoftMax over the distance; it is defined as a:

where, is the prototype of the model, is the labeled skeleton image sample. the inference phase of our model is similar to the nearest neighbor. Testing samples are divided into the support and test sets. Samples from support set will be mapped into metric space by . The prototype of our model is estimated by the mean of the different class of mapped support samples. When a sample from the testing set is feed into our model, it will be mapped by the same .Then the assignment of the sample is depended on the prototype, from whom it had the shortest distance. The algorithm 1 is a brief pseudo-code of the inference phase.



Figure : The structure of the skeleton-based action recognition model. Sample is assigned to the class whose prototype had the shortest distance from it

|  |
| --- |
| **Algorithm 1: the inference process of action recognition model.** |
| **Input：** classes of support set , where each class of support set contains labeled samples. the unlabeled sample  **Output:** the assignment of the unlabeled sample  **Foreach i in**    **End** |

## Training Phase

The parameter of our model can be divided into two parts. One is the support sets , it can be provided by a few labeled samples in real recognition scenarios. The other one is the parameter of the mapping function , which can be solved via SGD optimization. The purpose of our training algorithm is to address the learnable parameters of . The process of training phase is provided in algorithm 2. Analogy to the inference, the training samples are divided into support and query sets. Support sets is used to estimate the prototype of each classes. The query sets, mapped into embedding space, will be used to adjust the parameter . Figure 4 illustrates the detail computation of the cross-entropy loss of the model. Having the loss function defined, the parameters can be updated via the gradient descent method on it.



Figure : The computational graph of the model.

|  |
| --- |
| **Algorithm 2: the training process of action recognition model.** |
| **Input：** classes of action training set , where the i-th class of training sample is labeled as and contains samples. and are the number of support and query sample.  **Output:**    **Repeat:**  **Foreach i in**    **End**  **Foreach i in**        **End**  **Foreach i in**  **For each in**  **End**  **End**    **END** |

# **Experiments**

The whole architecture of our model is provide in Figure 3, as mentioned before, we construction the as a serial of dilated-dense layers. The structure of network is illustrated in Figure2. It composed of three dilated-dense layers. We implemented our method on Tensorflow with GeForce 920MX and evaluated it on a series of datasets. A reproducible piece of code is also provided in github **https://github.com/NanYoMy/human\_action\_recognition**

## UTD-MHAD

The UTD-MHAD dataset contains 27 classes of actions performed by 8 subjects (4 females and 4 males), Each subject repeated each action 4 times[20]. So, we get 32 samples for each action. From each class, we select 8(4 support, 4 query) samples for model training, and leave out 24 samples for testing. Table1 shows the recognition accuracy of different methods on UTD-MHAD dataset.

Table 1: comparison of different action recognition methods on UTD-MHAD.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **method** | **Exp1** | **Exp2** | **Exp3** | **Exp4** |
| ELC-KSVD | 76.19% | | | |
| kinect & Inertial | 79.10% | | | |
| Cov3DJ | 85.58% | | | |
| SOS | 86.97% | | | |
| JTM | 87.90% | | | |
| TSIIM-MSDCNN[21] | **96.27%** | | | |
| Our method 5-support | **97.07%** | **95.99%** | **96.91%** | **96.14%** |
| Our method 1-support | **96.14%** | **95.22%** | **95.83%** | **96.61%** |

## unseen classes

To investigate the performance of the model when dealing with the samples from the unseen class, we must evaluate it with the action sequences that have not presented during the training phase. For this purpose, the UTD-MHAD dataset is split into two uncorrelated sets. The training set holds 10 classes of the actions from the whole dataset, and the remaining 17 is allotted to the inference set. During the training phase, we select ( support, query) samples from each type of action , with which the parameters of the model are optimized. During the inference phase, we randomly select 5 samples from each remained class to estimate the prototype of them. And the rest 27 samples are used to validate the accuracy of our model. In this situation, the sample labels are totally different between the training and inference sets. Table2 gives the recognition accuracy of our model with different training and testing settings.

Tables 2: the classification accuracies of the model with different training setting

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Setting** | **Exp1** | **Exp2** | **Exp3** | **Exp4** |
| support=5 n\_class =5 | **92.97%** | **93.86%** | **94.41%** | **95.58%** |
| support=1 n\_class = |  |  |  |  |
| support=3 n\_class = |  |  |  |  |

# Conclusion

We present an action recognition method based on a few shot learning. It could handle the unseen classes with a few support samples. But, the classification ability of the model is limited; it could only classify a sample into a fixed number of types. For future work, we will focus on to develop the model to work well with an arbitrary number of output classes. Besides, our method cannot be applied to the long-term skeleton sequence, which may contain a different type of actions. We will extend our action recognition task to the segmentation of the long-term skeleton sequence based on the few shot learning.

# Acknowledgment

This work was supported in part by the Educational research projects for young and middle-aged teachers of Fujian under Grants JAT170796

# Reference

1 Wang, P., Li, Z., Hou, Y., & Li, W. (2016). Action Recognition Based on Joint Trajectory Maps Using Convolutional Neural Networks. *Acm on Multimedia Conference*.

2 Pham, Huy-Hieu, Khoudour, L., Crouzil, A., Zegers, P., & Velastin, S. A. (2018). Learning and recognizing human action from skeleton movement with deep residual neural networks.

3 Li, C. , Zhong, Q. , Xie, D. , & Pu, S. . (2017). Skeleton-based action recognition with convolutional neural networks.

4 Liu, M. , Liu, H. , & Chen, C. . (2017). Enhanced skeleton visualization for view invariant human action recognition. Pattern Recognition, 68, 346-362.

5 Bo, L., He, M., Cheng, X., Chen, Y., & Dai, Y. (2017). Skeleton based action recognition using translation-scale invariant image mapping and multi-scale deep cnn.

6 Zhu, W. , Lan, C. , Xing, J. , Zeng, W. , Li, Y. , & Shen, L. , et al. (2016). Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks.

7 Liu, J. , Shahroudy, A. , Xu, D. , Kot Chichung, A. , & Wang, G. . (2017). Skeleton-based action recognition using spatio-temporal lstm network with trust gates. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1-1.

8 Liu, J., Wang, G., Duan, L., Abdiyeva, K., & Kot, A. C. (2018). Skeleton-Based Human Action Recognition With Global Context-Aware Attention LSTM Networks. IEEE Transactions on Image Processing, 27(4), 1586-1599.

9 Bruna, J. , Zaremba, W. , Szlam, A. , & Lecun, Y. . (2013). Spectral networks and locally connected networks on graphs. Computer Science.

10 Shi, L. , Zhang, Y. , Cheng, J. , & Lu, H. . (2018). Non-local graph convolutional networks for skeleton-based action recognition.

11 Li, C. , Cui, Z. , Zheng, W. , Xu, C. , & Yang, J. . (2018). Spatio-temporal graph convolution for skeleton based action recognition.

12 Si, C., Chen, W., Wang, W., Wang, L., & Tan, T. (2019). An attention enhanced graph convolutional lstm network for skeleton-based action recognition.

13 Davis, J. V., Kulis, B., Jain, P., Sra, S., & Dhillon, I. S. (2007). Information-theoretic metric learning. Icml 07: International Conference on Machine Learning.

14 Mishra, A. , Verma, V. K. , Reddy, M. S. K. , Arulkumar, S. ,. , Rai, P. , & Mittal, A. . (2018). A generative approach to zero-shot and few-shot action recognition.

15 Yang, H., He, X., & Porikli, F. (2018). One-Shot Action Localization by Learning Sequence Matching Network. computer vision and pattern recognition.

16 Ke, Q., Bennamoun, M., An, S., Sohel, F. A., & Boussaid, F. (2017). A New Representation of Skeleton Sequences for 3D Action Recognition. computer vision and pattern recognition, 4570-4579.

17 Liu, M., Liu, H. W., & Chen, C. (2017). Enhanced skeleton visualization for view invariant human action recognition. Pattern Recognition, 68(68), 346-362.

18 Li, B., Chen, H., Chen, Y., Dai, Y., & He, M. (2017). Skeleton boxes: Solving skeleton based action detection with a single deep convolutional neural network. international conference on multimedia and expo, 613-616.

19 Snell, J., Swersky, K., & Zemel, R. S. (2017). Prototypical Networks for Few-shot Learning. neural information processing systems, 4077-4087.

20 Chen, C., Jafari, R., & Kehtarnavaz, N. (2015). UTD-MHAD: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor. international conference on image processing.

21 Li, B., Dai, Y., Cheng, X., Chen, H., Lin, Y., & He, M. (2017). Skeleton based action recognition using translation-scale invariant image mapping and multi-scale deep CNN. international conference on multimedia and expo, 601-604.