MSTCN-VAE: An unsupervised learning method for micro gesture recognition based on skeleton modality

Wenxuan Yuan[1], Shanchuan He[2,†] and Jianwen Dou[3,†]

[123] Taiyuan University of Technology

Abstract

We propose a novel unsupervised model for micro-gesture classification, called MSTCN-VAE, which follows the VAE structure by adding the Multi-scale TCN and hidden feature extraction block to the encoder, and the decoder is embedded with the Temporal Deconvolution block. The MSTCN-VAE model collects more temporal information from the input action sequences due to the advanced time series information integration method and thus exhibits better classification performance. By evaluation of the iMiGUE dataset, our approach outperforms the current state-of-the-art unsupervised methods in micro-gesture classification and is comparable to the accuracy of slightly earlier supervised models. Also, we validate the effectiveness of our model on the SMG dataset.

Keywords

Multi-Scale TCN, Unsupervised Network, Temporal Deconvolution, VAE structure

1. Introduction

The recognition of human gestures and actions plays a crucial role in various domains, ranging from human-computer interaction [1][2][3] to video surveillance [4][5][6] and robotics [7][8][9][10]. Over the years, there has been significant progress in the field of skeleton-based action recognition [11][12], where the skeletal representation of human body movements is utilized for analyzing and understanding human gestures. Skeleton-based approaches [13][14][15] offer a compact and informative representation that captures the spatial and temporal dynamics of human actions, enabling the efficient processing of gestures and facilitating the extraction of relevant features for recognition tasks. While skeleton-based action recognition has achieved remarkable success, there is a growing interest in exploring micro-gesture recognition, which focuses on recognizing subtle and fine-grained hand movements. Micro-gestures are characterized by intricate hand poses and subtle temporal variations, making them challenging to capture and understand. To address this research frontier, many methods have been proposed. In supervised micro-gesture recognition, approaches like deep learning-based convolutional neural networks (CNNs) [16][17] and attention convolutional networks [18] have been widely employed. On the other hand, unsupervised methods aim to learn representations or discover patterns from unlabeled or weakly labeled data. Some articles explore techniques such as hidden Markov models [19], sparse coding [20] , and local temporal features [21]. Unsupervised methods play a crucial role in scenarios where annotated training data is scarce or unavailable, allowing for the discovery of meaningful micro-gesture representations directly from raw data.

Micro-gesture recognition networks in the mainstream are predominantly supervised [22][23], relying on labeled data for training. However, collecting micro-gesture datasets presents challenges due to the difficulty of capturing and annotating subtle hand movements. This process often results in multiple labels for the same sample, introducing ambiguity. To address the limitations of supervised methods, researchers have explored unsupervised approaches for micro-gesture datasets. One notable method is Predict & Cluster framework [24], which provides a way to automatically recognize actions from skeletal data with the special form of GRU and shows promising results on multiple benchmark datasets. Another one is unsupervised S-VAE (U-S-VAE) [25], which indicates the effectiveness of using multi-layer BLSTM to extract information from a skeleton-based dataset. These studies highlight the significance of temporal modeling and latent space representations in unsupervised micro-gesture recognition. However, both GRU (Gated Recurrent Unit) [26] and LSTM (Long Short-Term Memory) [27] have certain limitations because of the computational complexity and limited time information integration when it comes to effectively modeling long-term dependencies and integrating time information. These limitations have led to the development of alternative architectures like TCN (Temporal Convolutional Network) [28], which benefits from the inherent parallelism of convolutional operations

MiGA@IJCAI23: International IJCAI Workshop on Micro-gesture Analysis for Hidden Emotion Understanding, August 21, 2023, Macao, China.

[†] These authors contributed equally.

2799782134@qq.com (W. Yuan); 3081146253@qq.com (S. He); 484298512@qq.com (J. Dou)

0009-0006-7496-2140 (W. Yuan); 0009-0003-2484-6991 (S. He); 0009-0000-9084-8211 (J. Dou)

and the ability to increase the receptive field exponentially with depth to learn patterns over extended time horizons and the stability of gradient. These qualities make TCN a promising alternative for tasks involving time series data.

To address these issues, an innovative unsupervised network model for gesture classification is proposed in this paper. The network is based on a VAE structure [29] using a temporal convolutional network (TCN) [28] and a multiscale temporal convolutional network (MSTCN) as the encoder and a TDCN (Temporal Deconvolutional Network) as the decoder. By conducting experiments on the original skeleton data as well as on the data after extraction of angular information, we demonstrate the advantages of the model, such as label dependence on the dataset and capturing the hidden feature vectors that are crucial for micro-gesture classification. Different variants of our network are evaluated on the now popular micro-gesture dataset iMiGUE [25] and compared with state-of-the-art supervised unsupervised methods. The wide applicability of the network is validated on the SMG dataset [30]. The potential of our approach in advancing skeleton-based micro-gesture recognition and further improving human-computer interaction is highlighted.

2. Related Work

Skeleton-based action recognition has been a growing area of focus in computer vision research. The related works broadly cover methodologies from hand-crafted features to deep learning models.

An early approach [31], in which a skeleton-based representation called Actionlet Ensemble is used for action recognition. It identifies and groups related parts of the skeletons that form meaningful sub-actions, termed Actionlets. The advent of deep learning has significantly improved the performance of skeleton-based action recognition. Additionally, a hierarchical RNN [32] was proposed for skeleton-based recognition. The model hierarchically constructs five parts of the body and then connects them in a temporal recurrent layer. More recent works leverage attention mechanisms to focus on discriminative joints or frames. Moreover, an attention mechanism in Long Short-Term Memory (LSTM) networks [33] is proposed, which can selectively focus on informative joints in the skeleton.

Although there are already many excellent supervised skeleton-based methods to recognize, these methods rely on labels that we have made. Manual annotation not only requires a lot of manpower and financial resources, and accuracy cannot be guaranteed. If we take a supervised approach, we must classify this set of actions into the types of actions we already know, and there may be some kinds we can't discern. Besides, when we train a supervised neural network, we can only use data that we have labeled, and more unmarked data will be wasted.

The unsupervised method based on skeleton data has come into view to conquer the aforementioned problems. Graph convolutional neural networks are widely used in graph correlation recognition [34][35], but action recognition depends on long-term information. Most of the frameworks are based on recurrent neural networks (RNNs), convolutional neural networks (CNNs), or graph-based CNNs. Employing methods directly tends to ignore the most important information in action recognition, which includes the interrelationship and timing of the movements. Different from the above framework, a novel model-aware gesture-to-gesture translation method is proposed, which presents novel approaches, called Self-Attention Network (SAN) [36]. Furthermore, a Focal and Global Spatial-Temporal Transformer network (FG-STFormer)[37].

Different from the previous form of network optimization, a new unsupervised model [24] is based on an encoder-decoder system. The encoder is responsible for feature extraction from the original data to obtain a feature vector that can be separated. The decoder needs to restore the extracted features to the original action sequence. They set up an evaluation system to measure the difference between the original data and the restored data. The cluster used the intermediate feature vectors generated by the encoder. Specifically, the encoder is a multi-layered bidirectional Gated Recurrent Unit (GRU) and the decoder is a uni-directional GRU. Afterward, another structure U-S-VAE [25], which is different from [5] in that BLSTM is used instead of BI-GRU. Our structure is also similar to several approaches [24][25]. The encoder-decoder system is also a vital part of our network structure. We adopt the hidden representation from the encoder as our classification feature vector. Furthermore, we incorporated TCN and multi-scale TCN in the encoder to integrate temporal information. In addition, due to the excellent performance of deconvolutional neural networks in GAN networks for data generation [38], we embedded TDCN (temporal deconvolutional network) in the decoder.

3. Methods

3.1. Preliminary

Data angle information extraction: The skeleton data is a sequence X_T^{3K} of T frames, and each frame is the 2D location information and confidence about the K-th joints node:

$$X_T^{3K} = \{x_1, x_2, ...x_t, ..., x_T\}$$
$$x_t = \{x_t^1, y_t^1, c_t^1, x_t^2, y_t^2, c_t^2, \dots, x_t^K, y_t^K, c_t^K\}$$

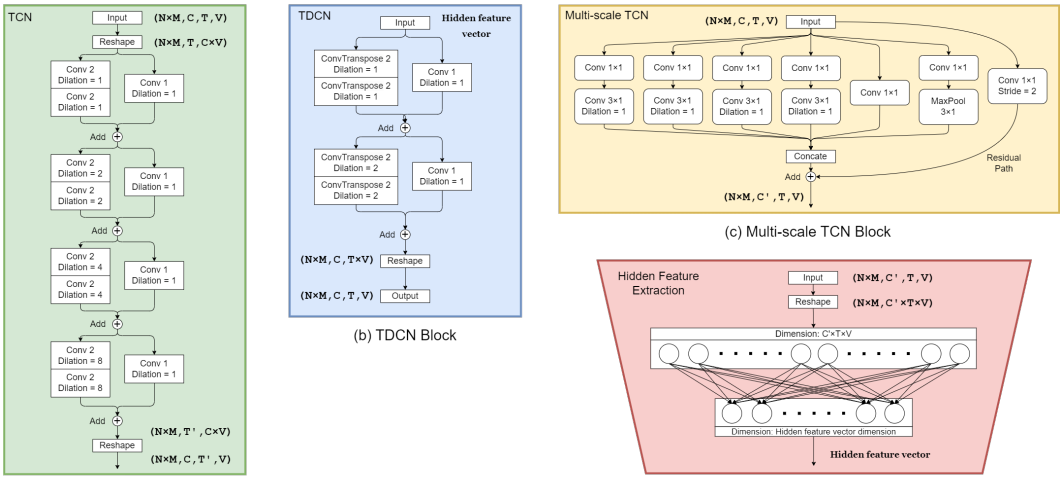

Figure 1: BLocks for MSTCN-VAE model and TCN-VAE model. 'TCN' and 'TDCN' denotes temporal convolutional blocks and temporal deconvolutional blocks, respectively. N, M, C, T, V represents batch size, number of people per frame, number of channels in the data set, frame length after downsampling, number of joints.

Where c_t^k is the confidence of the 2D location information (x_t^1, y_t^1), which is about the k-th joints node in the t frame.

To overcome the difference in the position information of the character in the view, we choose to use Angle information instead of the original 2D Cartesian coordinate data.The Angel data is a sequence X_T^{2A} of T frames, and each frame is the angular formation of three nodes in sequence and confidence:

$$X_T^{2A} = \{x_1, x_2, ...x_t, ..., x_T\}$$
$$x_t = \{a_t^1, c_t^1, a_t^2, c_t^2, ..., a_t^A, c_t^A\}$$

Where c_t^i is the confidence of a_t^i.*TheAngle* is the order of the three adjacent nodes (e.g., right shoulder, right elbow, right hand). To deal with the unity of left and right angles, we take counterclockwise or clockwise angles on both sides (for example, the angles of the right shoulder, right elbow, and right hand is counterclockwise, and the angles of the left hand, left elbow and left hand is clockwise).

While we use the angular information,we should note that we can no longer convert Angle data to coordination data.Therefore, when this transformation happens, we lose some information that we can't be sure of useful.So we propose two data supplement solutions.Both methods add distance information to the original Angle data.The first is the distance from the center of the Angle to the center of the body, which is the shoulder center, and the other is the length of the second side formed by the Angle.

$$X_T^{3A} = \{x_1, x_2, ... x_t, ..., x_T\}$$

$$x_t = \{a_t^1, d_t^1, c_t^1, a_t^2, d_t^2, c_t^2, ..., a_t^A, d_t^A, c_t^A\}$$

For the sake of convenience in the later part of this paper, we will refer to the data extracted from the angle information as AE data, while the data in the dataset that has not been changed in any way, i.e. original skeleton data, will be referred to as OS data.

3.2. Model Architecture

MSTCN-VAE network structure: The advantage of unsupervised methods over fully supervised methods is that they do not require manually labeled data. In this paper, referring to the VAE unsupervised model proposed by previous researchers [25][24], an encoder-decoder model is introduced to learn unlabeled micro-gesture sequence data (key points-based or angle-information-extracted). However, compared to existing unsupervised models our network has the following key differences: 1) We put TCN and Multi-scale TCN (MSTCN) into the encoder for integration of temporal information, respectively. This is because TCN has been shown to have better integration of temporal information compared to RNNs, LSTMs, and GRUs [39]. Also, MSTCN will collect more information than TCN due to the joint effect of different size receptive fields [40]. 2) We embed a temporal deconvolutional network module in the decoder to generate an initial sequence of gesture actions based on hidden features. On the one hand, deconvolutional neural networks are widely used in Generative adversarial networks to generate data [41],[42], and on the other

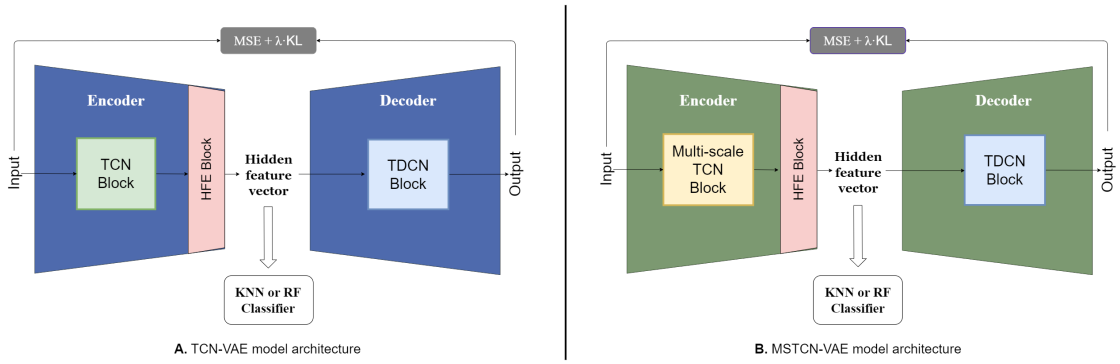

Figure 2: Overview of TCN-VAE and MSTCN-VAE model.

hand to use operations in the decoder similar to those in the encoder as a way to better generate the original micro-gesture sequence data. In terms of the loss function, similar to U-S-VAE [25] we use a linear combination of L_r and L_k as the plausible loss. L_r computes the MSE loss between the decoder-generated vector and the input vector, and this term aims to make the decoder-generated result as similar as possible to the input action sequence data. L_k computes the *Kullback-Leibler* (KL) divergence, and the KL divergence norm term is to ensure a closer approximation to the joint distribution and the product of the marginals, i.e. makes the encoder-generated hidden variables conform to the standard normal distribution as much as possible.

Hidden feature vector clustering: A vital feature in our network architecture is that we use two fully connected layers after multi-scale convolution in the temporal dimension and use this to form feature clusters. In other words, the feature clusters consist of hidden features integrated by temporal convolution [43]. Such a strategy is effective and promising when unsupervised methods are used for clustering multidimensional sequences, such as in body and gesture junction sequences [44][25]. It has been experimented with and displayed that fully connected layers are extensively applicable to RNN architectures [24], and in our demonstration, it can be found that fully connected layers under VAE structures will also help temporal convolution extract hidden features to some extent. Therefore, we put a hidden feature extraction (HFE) block which consists of two fully connected layers into the end of the encoder to extract the multi-nodal temporal information after MSTCN integration. In this way, we implement a codec system, called Multi-Scale Temporal Convolutional variational autoencoder (MSTCN-VAE), in which the original time series is input to the encoder and the encoder passes the low-dimensional hidden feature vectors to the decoder.

MSTCN-VAE Motion Prediction: Our proposed MSTCN-VAE network framework is depicted in detail in **Figure** 1 and **Figure** 2. A four-dimensional data X of the shape $(N \cdot M, C, T, V)$, where N is equal to the size of the batch size, M represents the number of people in each frame, C represents the number of channels, T represents the frame length of each action sample, and V represents the number of human features in each frame. It is important to clarify that $C = 3$ in OS data and consists of the x-coordinate and y-coordinate of the joint point and the confidence level of that point, and $C = 3$ in AE data and consists of the angle value of the pinch angle, the length of the line segment of the corresponding joint point, and the corresponding confidence level. X is first extracted by the MSTCN module of the encoder with different scales of convolutional kernels for temporal information, and then the data is stitched according to the C dimension for data stitching and then shaped into $(N \cdot M, C' \cdot T \cdot V)$ data (C' represents the size of C dimension after stitching). Subsequently, a HFE block consisting of two fully connected layers performs dimensionality reduction on this data, reducing the computational effort for clustering while not losing information as much as possible. For the dimensionality reduction, the data is then shaped into $(N \cdot M, C, T \cdot V)$ three-dimensional data $\tilde{X}$ after the deconvolution module and the initial time series data X which reshaped as $(N \cdot M, C, T \cdot V)$ is used to calculate the Loss value by $L = L_r + \lambda \cdot L_k$, where $L_r = \|X - \tilde{X}\|^2$, λ is used to describe the weight of the kl-divergence loss.

3.3. Classification methods

Unsupervised K-nearest neighbors classifier: In order to evaluate our action classification effect more explicitly, for the hidden feature vectors generated by the encoder, we use the K-nearest neighbors classifier (KNN). In other words, all the sequence data in the training set are forward propagated in the current training

network to obtain the hidden feature vectors of all the training data when calculating the accuracy, and this is used to form the KNN classification space. After the same forward propagation for each sample in the test set, the KD-tree algorithm is used to quickly search for the neighboring samples in the just-formed classification space. It is worth noting that although the composition of the KNN classification space uses the labels of the training set, the labels are only used to assign categories and are not involved in model training.

Supervised random forest classifier: The supervised classification method was used to evaluate the performance of our model from multiple perspectives. Specifically, for the hidden feature vector generated by the encoder, we put it into a Random Forest classifier (RF classifier). All training sets are also forward propagated under the current network to obtain the hidden feature vectors. The RF classifier is used to fit these vectors and the accuracy is calculated on the test data after forward propagation in the evaluation phase. It is important to clarify that since the RF classifier uses the label information to form the classification space, the model at this point belongs to the supervised network.

4. Experiment

4.1. Dataset

iMiGUE: iMiGUE dataset [25] focuses on unconscious micro-gesture movements without identity information. The dataset uses the OpenPose video dataset [45] gesture estimation toolbox to extract 18499 action samples from 359 post-race press conference videos, which are categorized into 31 micro-gesture action categories as well as one non-micro-gesture category. Each frame of the skeleton map consists of V = 22 upper body joints as nodes, and the coordinates of each point consist of data in three dimensions: two-dimensional spatial coordinates and prediction confidence scores. In MiGA Workshop & Challenge 2023, the entire dataset was divided into a training dataset consisting of 13670 samples and a test dataset consisting of 4562 samples.

SMG: SMG dataset [30] is a novel spontaneous micro-gesture dataset. From 414 long video instances containing 40 participants, the SMG dataset extracted 3712 micro-gesture action clips, where the average length of these clips was 51.3 frames, and labeled them with 16 micro-gesture action categories as well as one non-micro-gesture category. Based on the authors' suggestion, we evaluated our proposed model with 610 test samples in the body skeleton data model of this dataset. Following the convention of some articles [46] [47], we show the accuracy of Top1 and Top5 on this dataset.

iMiGUE dataset

	Methods	Top1	Top5
Supervised	S-VAE	27.38	60.44
	ST-GCN	46.97	84.09
	Shift-GCN	51.51	88.18
	MS_G3D	**54.91**	**89.98**
	TCN_VAE(with RFC) (OS data)(***Our***)	39.11	48.55
	MSTCN_VAE(with RFC) (OS data)(***Our***)	41.23	51.64
	MSTCN_VAE(with RFC) (AE data)(***Our***)	35.73	45.20
	MSTCN_VAE(with RFC) (AE data + OS data)(***Our***)	47.69	56.36
Unsupervised	P&C	31.67	**64.93**
	U-S-VAE	32.43	64.30
	TCN_VAE (with out HFE) (OS data) (***Our***)	24.44	39.54
	TCN_VAE (OS data)(***Our***)	28.50	44.91
	MSTCN_VAE (OS data)(***Our***)	30.84	45.22
	MSTCN_VAE (AE data + OS data)(***Our***)	**35.38**	50.07

Table 1
Comparison of micro-gesture recognition accuracy (%) with state-of-the-art algorithms on the iMiGUE dataset (best supervised method: **Black with bold**, best unsupervised method: **Blue with bold**). HFE and RFC denotes hidden feature extraction block and random forest classifier.

SMG dataset

	Methods	Top1	Top5
Supervised	ST-GCN	41.48	86.07
	Shift-GCN	55.31	87.34
	MS_G3D	**64.75**	**91.48**
	MSTCN_VAE(with RFC) (OS data)(***Our***)	42.59	49.54
Unsupervised	MSTCN_VAE(OS data) (***Our***)	**30.06**	**45.28**

Table 2
Validation on the SMG dataset and Comparison with currently known methods(best supervised method: **Black with bold**, best unsupervised method: **Blue with bold**).RFC denotes random forest classifier.

4.2. Implementation Details

To train the network, each action sample was downsampled by up to 100 frames. The joint point data in each skeleton map were also normalized. For the optimization hyperparameters, unless otherwise stated, all models were optimizer: SGD, batch size: 32, an initial learning rate: 0.0001, epoch: 200, LR decay rate: 0.1, LR decay step:

(100, 150). By random hyperparametric grid search, 1) for the encoder that only uses TCN blocks, setting the following network structure: Encoder: using one TCN block, each of which convolves T-dimension into 75, 50, 25, and 1, Gradually; Decoder: by one TDCN block, which deconvolutes T-dimension of sizes 50 and 100, Gradually. 2) for the encoder that uses TCN block and HFE block, the following network structure is set: Encoder: consists of one TCN block and one HFE block, TCN block convolves T-dimension into 75, 50, 25, 1, Gradually; Decoder: one TDCN block, deconvolutes T-dimension into 50, 100, Gradually. 3) For the encoder using one MSTCN block and HFE, the following network structure is set: Encoder: similar to the setting for MSTCN blocks [48], utilizing one MSTCN block and one HFE block; Decoder: consists of one TDCN block, which deconvolutes the T-dimension into 50, 100, Gradually.

For the above 1) and 2) models, the hidden feature vectors used for classification are all 66-dimensional, and for the 3) model, the hidden feature vectors are 128-dimensional. In order to avoid gradient explosion during the training process, gradient truncation will be performed when the maximum norm is greater than 10. In calculating the loss and performing backpropagation, we found that the overall training effect of the model was best when the value of λ in the loss function was 0.8 after several experiments. For the iMiGUE dataset, both the original skeleton data and the pre-processed data with the above angular information were input to the model to demonstrate the improvement of the model accuracy with the new data. However, for the SMG dataset, we only validated the effectiveness of our model on the raw skeleton data.

In the model evaluation session, for our different MSTCN-VAE variants (a combination of the methods described in Section 3.2), Top1 accuracy and Top5 accuracy are calculated uniformly using k=1 under the KNN classifier and k=1, 2, 3, 4, 5 combined, and random_state =1 under the Random forest classifier Top1 accuracy and random_state =1, 2, 3, 4, 5 are used to calculate Top5 accuracy. In the top5 calculation, for the classification results under five different parameters of the classifier, the prediction is considered correct as long as it contains the correct category. All experiments are based on an RTX 3080 (10GB) GPU and a 12 vCPU Intel(R) Xeon(R) Platinum 8255C CPU for training and evaluation.

4.3. Evaluation and Comparison

State-of-the-art supervised and unsupervised action recognition methods based on skeleton data have been applied to the iMiGUE dataset and the SMG dataset, e.g. [48],[49],[50],[51],[24],[25]. To highlight the advantages of our model, the above methods and the detailed accuracy of our method on these two data are presented comparatively in table 1 and table 2.

In table 1, first, compared between different MSTCN-VAE variants. We find that the hidden feature extraction block has about 3% improvement in the accuracy of the model, and Multi-scale has a 7% positive impact on the model. AE data has a negative impact on the model compared to OS data, but when AE data and OS data are judged together it brings a 4% improvement. In addition, the application of supervised classification in the VAE network structure brings a significant 10% increase in the model. Second, compared to supervised algorithms that are also based on skeleton recognition, our supervised model is at a considerable disadvantage since the graph connectivity property in the skeleton data is not taken into account. It is worth mentioning that for the supervised algorithm S-VAE, which also does not consider this property, our model has a considerable improvement in prediction. Finally, compared to similar unsupervised algorithms, the use of temporal convolution gives better classification results for skeleton-based data under the encoder-decoder system. However, since the process of calculating the Top5 of P&C and U-S-VAE in article [25] is ambiguous, this leads to the accuracy of our top5 and the top5 of the two models mentioned above not being directly comparable.

Of course, by validating the results on the SMG dataset in table 2, it can be found that the supervised and unsupervised MSTCN-VAE models are equally effective for other micro-gesture datasets. It is worth noting that our approach is the first to use a completely unsupervised temporal convolution method in skeleton-based recognition, and the results validate the effectiveness of our approach.

5. Conclusion

In this paper, we propose a novel skeleton-based micro-gesture recognition method. Our model connects a multi-scale temporal convolutional network with a hidden feature extraction block as an encoder to aggregate out hidden feature vectors and uses a temporal deconvolutional network in the decoder to generate action sequences from the hidden feature vectors. Through experiments on the iMiGUE dataset, we continuously improve and demonstrate the improvement of the MSTCN-VAE model over previous unsupervised methods, in addition to validation on the SMG dataset to further illustrate the effectiveness of our model.

Acknowledgments

Thanks to the developers of MS_G3D https://github.com/kenziyuliu/ms-g3d, P&C https://github.com/shlizee/Predict-Cluster and TCN https://github.com/locuslab/

TCN. And thanks to MiGA Workshop & Challenge 2023 https://cv-ac.github.io/MiGA2023/ for providing the baseline code.

References

[1] L. Chang, Y.-P. Tan, H.-C. Chua, Detection and removal of rainboweffect artifacts, in: 2007 IEEE International Conference on Image Processing, volume 1, 2007, pp. I – 297–I – 300. doi:10.1109/ICIP.2007.4378950.

[2] S. S. Rautaray, A. Agrawal, Vision based hand gesture recognition for human computer interaction: a survey, Artificial Intelligence Review 43 (2012) 1 – 54.

[3] Y. Sun, C. Xu, G. Li, W. Xu, J. Kong, D. Jiang, B. Tao, D. Chen, Intelligent human computer interaction based on non redundant emg signal, Alexandria Engineering Journal 59 (2020). doi:10.1016/j.aej.2020.01.015.

[4] R. Iltis, Array beamforming for slow fh spread-spectrum wlans, in: Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers, 2002., volume 2, 2002, pp. 1683–1687 vol.2. doi:10.1109/ACSSC.2002.1197062.

[5] P. Xu, A real-time hand gesture recognition and human-computer interaction system (2017).

[6] Y. Zhu, Z. Yang, B. Yuan, Vision based hand gesture recognition, in: 2013 International Conference on Service Sciences (ICSS), 2013, pp. 260–265. doi:10.1109/ICSS.2013.40.

[7] S. Gulati, R. K. Bhogal, Comprehensive review of various hand detection approaches, in: 2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET), 2018, pp. 1–5. doi:10.1109/ICCSDET.2018.8821238.

[8] M. Hasanuzzaman, T. Zhang, V. Ampornaramveth, M. Bhuiyan, Y. Shirai, H. Ueno, Gesture recognition for human-robot interaction through a knowledge based software platform 3211 (2004) 530–537.

[9] V. K. Thakur, Priyadarshni, Robust hand gesture recognition for human machine interaction system, Journal of Global Research in Computer Sciences 5 (2014) 14–19.

[10] G. Pozzato, S. Michieletto, E. Menegatti, F. Dominio, G. Marin, S. Milani, P. Zanuttigh, Human-robot interaction with depth-based gesture recognition, 2014.

[11] S. Yeung, O. Russakovsky, G. Mori, L. Fei-Fei, End-to-end learning of action detection from frame glimpses in videos, 2017. arXiv:1511.06984.

[12] G. Luo, S. Yang, G. Tian, C. Yuan, W. Hu, S. Maybank, Learning human actions by combining global dynamics and local appearance, Pattern Analysis and Machine Intelligence, IEEE Transactions on 36 (2014) 2466–2482. doi:10.1109/TPAMI.2014.2329301.

[13] L. Shi, Y. Zhang, J. Cheng, H. Lu, Skeleton-based action recognition with directed graph neural networks, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 7904–7913. doi:10.1109/CVPR.2019.00810.

[14] A. Zhu, Q. Ke, M. Gong, J. Bailey, Adaptive local-component-aware graph convolutional network for one-shot skeleton-based action recognition, 2022. arXiv:2209.10073.

[15] L. Wu, C. Zhang, Y. Zou, Spatiotemporal focus for skeleton-based action recognition, Pattern Recognition 136 (2023) 109231. URL: https://www.sciencedirect.com/science/article/pii/S0031320322007105. doi:https://doi.org/10.1016/j.patcog.2022.109231.

[16] M. Peng, C. Wang, T. Chen, G. Liu, X. Fu, Dual temporal scale convolutional neural network for micro-expression recognition, Frontiers in Psychology 8 (2017). URL: https://www.frontiersin.org/articles/10.3389/fpsyg.2017.01745. doi:10.3389/fpsyg.2017.01745.

[17] B. Dekker, S. Jacobs, A. Kossen, M. Kruithof, A. Huizing, M. Geurts, Gesture recognition with a low power fmcw radar and a deep convolutional neural network, in: 2017 European Radar Conference (EURAD), 2017, pp. 163–166. doi:10.23919/EURAD.2017.8249172.

[18] H.-X. Xie, L. Lo, H.-H. Shuai, W.-H. Cheng, Au-assisted graph attention convolutional network for micro-expression recognition, in: Proceedings of the 28th ACM International Conference on Multimedia, MM '20, Association for Computing Machinery, New York, NY, USA, 2020, p. 2871–2880. URL: https://doi.org/10.1145/3394171.3414012. doi:10.1145/3394171.3414012.

[19] R. Melani, Hand gesture recognition using hidden markov model algorithm, MATICS 9 (2017) 7. doi:10.18860/mat.v9i1.4126.

[20] S. Bhattacharya, P. Nurmi, N. Hammerla, T. Plötz, Using unlabeled data in a sparse-coding framework for human activity recognition, Pervasive and Mobile Computing 15 (2014) 242–262. URL: https://doi.org/10.1016%2Fj.pmcj.2014.05.006. doi:10.1016/j.pmcj.2014.05.006.

[21] Y.-J. Liu, B.-J. Li, Y.-K. Lai, Sparse mdmo: Learning a discriminative feature for micro-expression recognition, IEEE Transactions on Affective Computing 12 (2021) 254–261. doi:10.1109/TAFFC.2018.2854166.

[22] R. Zhi, H. Xu, M. Wan, T. Li, Combining 3d convolutional neural networks with transfer learning by supervised pre-training for facial micro-expression recognition, IEICE TRANSACTIONS on Information and Systems 102 (2019) 1054–1064.

[23] R. Zhi, J. Hu, F. Wan, Micro-expression recognition with supervised contrastive learning, Pattern Recognition Letters 163 (2022) 25–31. URL: https://www.sciencedirect.com/science/article/pii/S0167865522002690. doi:https://doi.org/10.1016/j.patrec.2022.09.006.

[24] K. Su, X. Liu, E. Shlizerman, Predict & cluster: Unsupervised skeleton based action recognition, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 9628–9637. doi:10.1109/CVPR42600.2020.00965.

[25] X. Liu, H. Shi, H. Chen, Z. Yu, X. Li, G. Zhao, imigue: An identity-free video dataset for micro-gesture understanding and emotion analysis, arXiv preprint arXiv:2107.00285, 2021. URL: https://arxiv.org/abs/2107.00285, [cs.CV].

[26] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using rnn encoder-decoder for statistical machine translation, 2014. arXiv:1406.1078.

[27] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (1997) 1735–1780.

[28] S. Bai, J. Z. Kolter, V. Koltun, An empirical evaluation of generic convolutional and recurrent networks for sequence modeling, arXiv preprint arXiv:1803.01271 (2018).

[29] D. P. Kingma, M. Welling, Auto-encoding variational bayes, arXiv preprint arXiv:1312.6114 (2013).

[30] H. Chen, H. Shi, X. Liu, X. Li, G. Zhao, Smg: A micro-gesture dataset towards spontaneous body gestures for emotional stress state analysis, International Journal of Computer Vision 131 (2023) 1–21. doi:10.1007/s11263-023-01761-6.

[31] S. Maji, L. Bourdev, J. Malik, Action recognition from a distributed representation of pose and appearance, in: CVPR 2011, 2011, pp. 3177–3184. doi:10.1109/CVPR.2011.5995631.

[32] Y. Du, W. Wang, L. Wang, Hierarchical recurrent neural network for skeleton based action recognition, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1110–1118. doi:10.1109/CVPR.2015.7298714.

[33] J. Liu, G. Wang, P. Hu, L.-Y. Duan, A. C. Kot, Global context-aware attention lstm networks for 3d action recognition, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 3671–3680. doi:10.1109/CVPR.2017.391.

[34] D. Miki, S. Chen, K. Demachi, Weakly supervised graph convolutional neural network for human action localization, in: 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), 2020, pp. 642–650. doi:10.1109/WACV45572.2020.9093551.

[35] L. Huang, Y. Huang, W. Ouyang, L. Wang, Part-level graph convolutional network for skeleton-based action recognition, Proceedings of the AAAI Conference on Artificial Intelligence 34 (2020) 11045–11052. doi:10.1609/aaai.v34i07.6759.

[36] S. Cho, M. H. Maqbool, F. Liu, H. Foroosh, Self-attention network for skeleton-based human action recognition, in: 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), 2020, pp. 624–633. doi:10.1109/WACV45572.2020.9093639.

[37] Z. Gao, P. Wang, P. Lv, X. Jiang, Q. Liu, P. Wang, M. Xu, W. Li, Focal and global spatial-temporal transformer for skeleton-based action recognition, in: L. Wang, J. Gall, T.-J. Chin, I. Sato, R. Chellappa (Eds.), Computer Vision – ACCV 2022, Springer Nature Switzerland, Cham, 2023, pp. 155–171.

[38] S. Addepalli, G. Nayak, A. Chakraborty, R. Babu, Degan: Data-enriching gan for retrieving representative samples from a trained classifier, Proceedings of the AAAI Conference on Artificial Intelligence 34 (2020) 3130–3137. doi:10.1609/aaai.v34i04.5709.

[39] S. Bai, J. Z. Kolter, V. Koltun, An empirical evaluation of generic convolutional and recurrent networks for sequence modeling, ArXiv abs/1803.01271 (2018).

[40] J. Zhang, Y. Wang, J. Tang, J. Zou, S. Fan, Ms-tcn: A multiscale temporal convolutional network for fault diagnosis in industrial processes, in: 2021 American Control Conference (ACC), 2021, pp. 1601–1606. doi:10.23919/ACC50511.2021.9482728.

[41] A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, CoRR abs/1511.06434 (2015).

[42] M. Arjovsky, S. Chintala, L. Bottou, Wasserstein generative adversarial networks, ICML'17, JMLR.org, 2017, p. 214–223.

[43] M. Farrell, S. Recanatesi, G. Lajoie, E. Shea-Brown, Recurrent neural networks learn robust representations by dynamically balancing compression and expansion, in: Real Neurons & Hidden Units: Future directions at the intersection of neuroscience and artificial intelligence @ NeurIPS 2019, 2019. URL: https://openreview.net/forum?id=BylmV7tI8S.

[44] K. Su, E. Shlizerman, Clustering and recognition of spatiotemporal features through interpretable embedding of sequence to sequence recurrent neural networks, 2020. arXiv:1905.12176.

[45] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, Y. Sheikh, Openpose: Realtime multi-person 2d pose estimation using part affinity fields, IEEE Transactions on Pattern Analysis and Machine Intelligence 43 (2021) 172–186. doi:10.1109/TPAMI.2019.2929257.

[46] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, A. Zisserman, The kinetics human action video dataset (2017).

[47] S. Yan, Y. Xiong, D. Lin, Spatial temporal graph convolutional networks for skeleton-based action recognition, AAAI'18/IAAI'18/EAAI'18, AAAI Press, 2018.

[48] Z. Liu, H. Zhang, Z. Chen, Z. Wang, W. Ouyang, Disentangling and unifying graph convolutions for skeleton-based action recognition, 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020) 140–149.

[49] H. Shi, X. Liu, X. Hong, G. Zhao, Bidirectional long short-term memory variational autoencoder, in: British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, September 3-6, 2018, BMVA Press, 2018, p. 165. URL: http://bmvc2018.org/contents/papers/0963.pdf.

[50] S. Yan, Y. Xiong, D. Lin, Spatial temporal graph convolutional networks for skeleton-based action recognition, Proceedings of the AAAI Conference on Artificial Intelligence 32 (2018). doi:10.1609/aaai.v32i1.12328.

[51] K. Cheng, Y. Zhang, X. He, W. Chen, J. Cheng, H. Lu, Skeleton-based action recognition with shift graph convolutional network, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 180–189. doi:10.1109/CVPR42600.2020.00026.

A. Online Resources

The sources code for the MSTCN-VAE model are available via

- MSTCN-VAE

The data set used in this article is available at

- iMiGUE
- SMG