Micro-gesture Online Recognition using Learnable Query Points

Pengyu Liu[1], Fei Wang[1], Kun Li[4,*], Guoliang Chen[1], Yanyan Wei[1], Shengeng Tang[1], Zhiliang Wu[4] and Dan Guo[1,2,3,5,*]

[1]School of Computer Science and Information Engineering, School of Artificial Intelligence, Hefei University of Technology (HFUT)
[2]Key Laboratory of Knowledge Engineering with Big Data (HFUT), Ministry of Education
[3]Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, China
[4]CCAI, Zhejiang University, China
[5]Anhui Zhonghuitong Technology Co., Ltd.

Abstract

In this paper, we briefly introduce the solution developed by our team, HFUT-VUT, for the Micro-gesture Online Recognition track in the MiGA challenge at IJCAI 2024. The Micro-gesture Online Recognition task involves identifying the category and locating the start and end times of micro-gestures in video clips. Compared to the typical Temporal Action Detection task, the Micro-gesture Online Recognition task focuses more on distinguishing between micro-gestures and pinpointing the start and end times of actions. Our solution ranks **2nd** in the Micro-gesture Online Recognition track.

Keywords

Micro-gesture, action online recognition, video understanding, Mamba

1. Introduction

Humans can express emotions and communicate with others through various non-verbal forms, among which gestures play a crucial role in emotional expression and communication [1, 2, 3, 4, 5]. Examples include "cover face", "fold arms", and "cross fingers", which convey human emotions to the outside world. Additionally, these micro-gestures (MGs) are often not spontaneous but occur unconsciously in specific environments. Unlike macro gestures intended for communication, non-spontaneous MGs better reflect genuine human emotions, making the study of MGs more meaningful in understanding human emotions. SMG [2] and iMiGUE [6] are the datasets to assess and analyze human emotional states through MGs information. These

The 2nd Workshop & Challenge on Micro-gesture Analysis for Hidden Emotion Understanding, Aug 3–9, 2024, Jeju, South Korea
*Corresponding author.
lpynow@gmail.com (P. Liu); jiafei127@gmail.com (F. Wang); kunli.hfut@gmail.com (K. Li); guoliangchen.hfut@gmail.com (G. Chen); weiyy@hfut.edu.cn (Y. Wei); tangsg@hfut.edu.cn (S. Tang); wu_zhiliang@zju.edu.cn (Z. Wu); guodan@hfut.edu.cn (D. Guo)
0000-0002-3396-3108 (P. Liu); 0009-0004-1142-6434 (F. Wang); 0000-0001-5083-2145 (K. Li); 0009-0002-7984-3184 (G. Chen); 0000-0001-8818-6740 (Y. Wei); 0000-0001-6313-2543 (S. Tang); 0000-0002-6597-8048 (Z. Wu); 0000-0003-2594-254X (D. Guo)

datasets provide a stronger representation of human emotions, significantly contributing to a deeper understanding of genuine human feelings.

Compared to common macro gestures, Micro-gesture Online Recognition is more challenging because MGs appear more irregularly and randomly than existing action or gesture recognition datasets. Additionally, there may be co-occurrence relationships between different classes of actions, and transformations may occur between different MGs. Moreover, the finer distinctions between different categories of MGs make it more difficult to determine the start and end times of actions due to their smaller movement amplitudes.

In this challenge, we adopt PointTAD [7] as the baseline. The main contributions of our method are as follows:

- We introduce the Mamba-MHSA block for Micro-gesture Online Recognition, which better distinguishes and locates action categories compared to the baseline model.
- In the Micro-gesture Online Recognition challenge, our solution achieves an F1 score of 14.34 on the test set, securing 2nd in the competition. The experimental results demonstrate that our model can effectively distinguish and locate MGs.

2. Related Work

Current research predominantly focuses on common macro gestures or actions [8, 9], which have limited capability in reflecting human emotions. This is because humans can subjectively control their gestures and actions to hide their true emotions. In contrast, MGs typically occur involuntarily and uncontrollably, providing a more accurate reflection of genuine human emotions, which is crucial for understanding behavior and emotions. Here, we review the related technologies: micro-gesture datasets, temporal action detection, and Mamba.

Micro-gesture Datasets. The iMiGUE [6] dataset is the first publicly available dataset, aimed at recognizing and understanding suppressed or hidden emotions through MGs. It includes 359 videos with a total duration of 2092 minutes, collected from 72 subjects from 28 countries. The dataset is annotated with 18,499 MG samples across 32 categories, averaging 51 MG actions per video, with each MG instance ranging from 0.18 seconds to 80.92 seconds, and an average duration of 2.55 seconds. The SMG [2] dataset focuses on naturally occurring MGs under stress, collected from 40 participants of various ages, genders, and racial backgrounds, divided into 16 types of MGs. The SMG dataset has been applied in various studies on micro-gesture recognition and emotion analysis, demonstrating its utility in these research fields.

Micro-gesture Online Recognition. Guo *et al.* [10] proposed a novel deep network combining graph convolution and Transformer encoders to extract motion features from 2D skeleton sequences. This combination leverages the strengths of both graph convolution and Transformer. Their contributions collectively advance the state-of-the-art in micro-gesture recognition, providing a robust framework for emotion analysis based on MGs.

Temporal Action Detection. Temporal action detection has been studied as a multi-label frame-wise classification problem in previous literature. Early models [11] mainly focused on modeling the temporal relationships between frames using Gaussian filters in the time dimension. Current research primarily deals with processing information at different scales and integrating spatiotemporal attention during processing. Tirupattur *et al.* [12] introduced

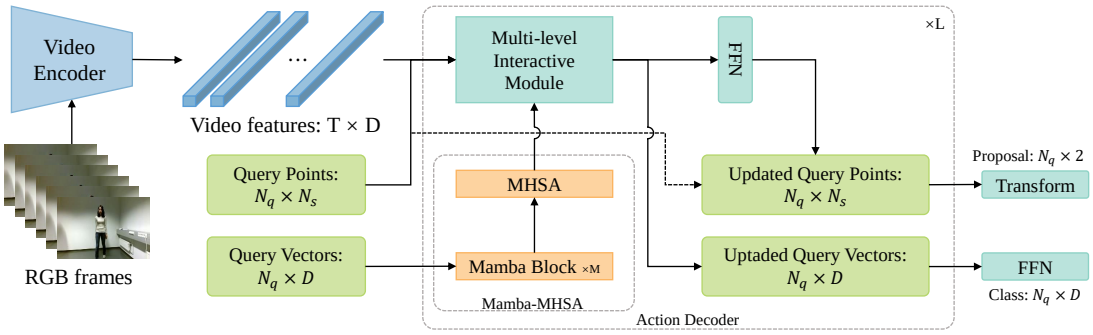

Figure 1: The proposed model consists of a video encoder, which extracts video features from continuous RGB frames, and an action decoder.

an attention-based Multi-label Action Dependency layer (MLAD) in their model, significantly improving the co-occurrence dependencies and temporal dependencies of actions. Dai *et al.* [13] proposed a novel ConvtransFormer network named MS-TCT that incorporates global and local time relationship encoders and a time-scale mixer for effective multi-scale feature fusion [14], addressing the complexities of temporal relationships. Tan *et al.* [7] presented an end-to-end action detection model named PointTAD that leverages learnable query points for precise localization and differentiation of actions in multi-label videos. These studies provide valuable insights for micro-gesture online recognition.

Mamba. The Transformer architecture and its core self-attention mechanism [15, 16, 17, 18] achieve significant success in deep learning. However, the Transformer faces inefficiency issues when processing long sequences. Structured State Space Models (SSMs) [19] [20], combining characteristics of Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs), have shown potential in certain data modalities. SSMs perform well on continuous signal data but less effectively on discrete and information-dense data. To address these shortcomings, Mamba introduces a selection mechanism that allows SSM parameters to adjust dynamically based on input data, improving model performance on discrete modalities. Mamba has notable advantages in inference speed and sequence length scalability. Thus, we incorporate Mamba into our model, combining Mamba [21] [22] with self-attention to better model different semantics.

3. Method

3.1. Task Definition

We formulate the Micro-gesture Online Recognition task as a set prediction problem. Given a continuous video clip with T frames, we predict a set of action instances $\phi = \{\phi_n = (t_n^s, t_n^e, c_n)\}_{n=1}^{N_q}$, where N_q is the number of learnable queries, t_n^s and t_n^e are the starting and ending timestamps of the n-th detected instance, and c_n is its action category. The ground truth action set to detect is denoted as $\hat{\phi} = \left\{\widehat{\phi_n} = \left(\widehat{t_n^s}, \widehat{t_n^e}, \widehat{c_n}\right)\right\}_{n=1}^{N_g}$, where $\widehat{t_n^s}$ and $\widehat{t_n^e}$ are the starting and ending timestamps of the n-th action, $\widehat{c_n}$ is the ground truth action category, and N_g is the number of ground truth actions.

3.2. Overall Architecture

The overall architecture of our model is shown in Figure 1. The model consists of a video encoder and an action decoder. For each video sequence, we select an RGB sequence of length T, a set of learnable query points $P = \{P_i\}_{i=1}^{N_q}$, and query vectors $Q = \mathbb{R}^{T \times D}$. The learnable query points are used to locate the positions of action boundaries, and the query vectors decode action semantics and positions from the features input to the model. The action decoder comprises L stacked decoder layers. Each layer of the action decoder takes video features, the latest query points P, and the latest query vectors Q as input. Each action decoder layer includes two parts: 1) the Mamba-MHSA block models the relationships among query vectors and the potential relationships between different action categories; 2) the Multi-level Interactive Module dynamically models the relationships based on query vectors between point-level and same action categories. Finally, we use a Feed-Forward Network(FFN) to decode the action labels from the query vectors and convert the query points into detection outputs.

3.3. Video Encoder

We use the I3D network [23] as our model's video encoder, integrating the video encoder with the action decoder for end-to-end training. To facilitate model deployment and speed up feature extraction, we avoid using the optical flow part of the I3D backbone network. Finally, the temporal stride of the encoded video features is 4, and the spatiotemporal representations are compressed into temporal features through spatial average pooling.

3.4. Learnable Query Points

Using only the start and end times to represent an action instance limits its boundary and content description. Therefore, to improve the representation flexibility, a point-based representation method is used to learn keyframes of action boundaries and semantics within instances. For each query, the point-based representation is $P = \{t_i\}_{i=1}^{N_s}$, where t_i is the time position of the i-th query point, and the number of points per query is N_s. During training, query points are initially placed at the midpoint of the input video sequence and are then refined through iterations in the action decoder layers by the query vectors Q, gradually approaching their final positions. Specifically, at each layer, the offsets of query points are predicted from the updated query vectors via linear projection. In action decoder layer l, the representation of a query's query points is $P^l = \left\{t_i^l\right\}_{i=1}^{N_s}$, with the offsets denoted as $\left\{\Delta t_i^l\right\}_{i=1}^{N_s}$. This operation can be summarized as:

$$P^{l+1} = \left\{\left(t_i^l + \Delta t_i^l \cdot s^l \cdot 0.5\right)\right\}_{i=1}^{N_s}, \tag{1}$$

where $s^l = \max\left(t_i^l\right) - \min\left(t_i^l\right)$. For relatively short actions, the update step size of the query points is smaller, aiding in the localization of short actions. Additionally, the action query points updated by the previous action decoder layer become the input to the next action decoder layer after passing through a layer of FFN.

3.5. Mamba-MHSA Block

Compared to Transformers [24, 25, 26], the recently proposed Mamba has demonstrated powerful capabilities in sequence modeling. Therefore, we introduce Mamba into our model and combine it with the Multi-Head Self-Attention (MHSA) to model the relationships of query vectors, forming the Mamba-MHSA block. Our Mamba-MHSA module consists of M of Mamba blocks and an MHSA. The Mamba block processes the query vectors Q^m of the m-th Mamba block based on a selective state space model.

Mamba is designed based on state space models (SSMs) and requires defining three key parameters $A \in \mathbb{R}^{D\times D}$, $B \in \mathbb{R}^{D\times 1}$, and $C \in \mathbb{R}^{1\times D}$. The SSMs are defined by the following differential equations:

$$h'(t) = Ah(t) + BQ^m(t), \tag{2}$$

$$y(t) = Ch(t). \tag{3}$$

We need to discretize the above equations. The discretized SSMs include a time parameter Δ, which converts the continuous parameters A and B into discrete parameters. The specific formulas are as follows:

$$A_x = \exp(\Delta A), \tag{4}$$

$$B_x = (\Delta A)^{-1}(\exp(\Delta A) - I)\Delta A. \tag{5}$$

After discretization, the block can be expressed as:

$$h_t = A_x h_{t-1} + B_x Q_t^m, \tag{6}$$

$$y_t = Ch_t. \tag{7}$$

Next, we use a global convolution operation to obtain the output Q^{m+1} by convolving the input sequence Q^m with a structured convolutional kernel K. The convolution kernel K is precomputed from the parameters A, B, and C, and its calculation method is as follows:

$$Q^{m+1} = Mamba(Q^m) = Q^m \times K = Q^m \times (CB, CAB, \ldots, CA^{D-1}B). \tag{8}$$

After passing through M of Mamba blocks, the query vectors Q^M are input into a Multi-Head Self-Attention block to obtain the output. With the Mamba-MHSA block, the model gains stronger selectivity and perceptual capability for the input query vectors, allowing it to better model the relationships between different action instances.

3.6. Multi-Level Interactive Module

Previous temporal action detectors often have deficiencies in decoding sampled frames, as they typically aggregate semantics from different aspects and levels infrequently. Thus, we consider a multi-level interactive module to aggregate multi-level semantics.

Point-Level Local Semantic Extraction We use the deformable convolution [27, 28] to extract point-level features within a local neighborhood. For the i-th query point, considering that more time offsets can more precisely cover the area around the sub-points, thereby capturing more information, but they also increase the computational cost, we predict 4 time offsets

$\{\Delta p_i\}_{i=1}^4$ and corresponding weights $\{w_i\}_{i=1}^4$ from the position of this point. Using the query point at frame t_i as the center point, we add time offsets to form four deformable sub-points. These sub-points represent the local area around the center point. The features at the sub-points are extracted through bilinear interpolation and multiplied by the weight values to obtain the point-level feature x_i. This process can be represented as:

$$x_i = \sum_{i=1}^{4} (t_i + \Delta p_i) \times w_i. \tag{9}$$

The offsets and weights are generated by linear projection from the query vector q. This process can be represented as:

$$\Delta q = \text{Linear}(q) \in \mathbb{R}^{N_q \times 4}, \tag{10}$$

$$w = \text{Softmax}(\text{Linear}(q)) \in \mathbb{R}^{N_q \times 4}. \tag{11}$$

Instance-Level Semantic Mixing Since actions can occur simultaneously, modeling only the temporal aspect may cause overlapping actions to have similar representations, leading to classification errors. Therefore, dynamic convolution is used to mix semantics across frames and channels. The mixed features of the query points use $x \in \mathbb{R}^{N_s \times D}$. Given the query vector q, the parameters for frame mix and channel mix are generated:

$$\theta_f = \text{Linear}(q) \in \mathbb{R}^{N_s \times N_s}, \theta_{c,1} = \text{Linear}(q) \in \mathbb{R}^{D \times D'}, \theta_{c,2} = \text{Linear}(q) \in \mathbb{R}^{D' \times D}. \tag{12}$$

Frame mix is performed by projecting and then activating with LayerNorm and ReLU across N_s points to explore intra-instance relationships:

$$x_f = \text{ReLU}(\text{LayerNorm}(x^T \theta_f)) \in \mathbb{R}^{D \times N_s}. \tag{13}$$

Channel mix enhances action semantics using dynamic projection along the channel dimension:

$$x_c = \text{ReLU}(\text{LayerNorm}(\text{ReLU}(\text{LayerNorm}(x\theta_{c,1}))\theta_{c,2})) \in \mathbb{R}^{N_s \times D}. \tag{14}$$

These two features are then concatenated along the channel and compressed through a linear layer to the size of the query vector. The query vector is updated to obtain the query vector for the next layer input q^{l+1}. This process can be represented as:

$$q^{l+1} = q^l + \text{Linear}(\text{Concat}(x_f^T, x_c)). \tag{15}$$

4. Experiments

4.1. Dataset and Evaluation Metric

Dataset. The spontaneous Micro-Gesture (SMG) dataset [2] consists of 3,692 samples of 17 MGs. The dataset employs a cross-subject evaluation protocol by dividing the 40 subjects into a training group consisting of long sequences from 35 subjects and a testing group of sequences from 5 subjects. We only use RGB sequences as input.

Evaluation Metric. We jointly evaluate the detection and classification performances of algorithms using the $F1$ score measurement defined below:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{16}$$

Given a long video sequence that needs to be evaluated, Precision is the fraction of correctly classified MGs among all gestures retrieved in the sequence by the algorithms, while Recall (or sensitivity) is the fraction of MGs that have been correctly retrieved over the total amount of annotated MGs.

4.2. Implementation Details

We use the I3D backbone network to extract video frames at a rate of 10 fps. A sliding window mechanism is employed to preprocess video sequences, with the window size(β) set to 128 frames to accommodate most action categories. During training, the overlap ratio is set to 0.75, while for inference, the overlap ratio is 0. We set N_q to 48 and N_s to 30. The I3D backbone uses pre-trained weights from Kinetics400 [29]. The batch size is set to 1, and the initial learning rate is 1e-4, halved every 10 epochs, for a total of 50 epochs.

Table 1
The top-3 results of Micro-gesture Online Recognition on the SMG test set. Data is provided by the Kaggle competition page[1].

Rank	Team	F1 Score
1	NPU-MUCIS	27.57
2	**HFUT-VUT(ours)**	**14.34**
3	JDY203	9.28

4.3. Experimental Results

As shown in Table 1, we report the results of the top three teams on the SMG dataset test set. Our team secured the second place. Although there remains a notable performance disparity between our method and the first-place "NPU-MUCIS" team, our method significantly exceeds the performance of the third-place "JDY203" team by 54.52%.

4.4. Ablation Study

Study on the Number of Query Points (α). In Table 2a, we conduct an ablation study on different numbers of query points. We observe that the model's performance improves as the number of query points increases when the number is less than 30. However, when the number of query points exceeds 30, the model's performance starts to decrease. Therefore, we choose 30 as the default number of query points for our model.

[1]The Kaggle competition page: https://www.kaggle.com/competitions/2nd-miga-ijcai-challenge-track2/leaderboard

Table 2
The ablation experiments of our method on the SMG dataset.

a. Query Points in action detectors parameter α

α	F1-score
25	11.33
27	13.48
30	**14.34**
31	13.49
32	13.43
35	9.81

b. Window size in action detectors parameter β

β	F1-score
16	7.33
32	8.45
64	9.27
128	**14.34**
200	7.95

c. Action decoder parameter L

L	F1-score
2	8.3
3	11.96
4	**14.34**
5	10.01

d. Mamba Block parameter M

M	F1-score
1	8.42
2	**14.34**
3	9.22

Study on Window Size (β). We examine the potential impact of different window sizes on the model results. We consider five different initializations for window size to accommodate the majority of action lengths in the dataset. As shown in Table 2b, the model achieves the best performance when the window size is set to 128. Thus, we set the window size to 128.

Study on the number of layers in the Action Decoder (L). We investigate the influence of different numbers of layers in the action decoder on the model. According to the results in Table 2c, increasing the number of layers in the action decoder allows the model to learn deeper information, thereby improving its performance. However, when the number of layers exceeds 4, the model's performance begins to decrease.

Study on the number of Mamba Blocks (M). To balance computational resources, we study the impact of the number of Mamba blocks on the model. As indicated in Table 2d, the model performs best when M is set to 2. Additionally, when the number of Mamba blocks exceeds 2, the model encounters issues with gradient explosion.

5. Conclusion

In this paper, we present a solution for the Micro-gesture Online Recognition (MiGA) challenge at IJCAI 2024. Our approach is based on the PointTAD baseline, enhanced with Mamba-MHSA to improve the model's ability to model sequences. This module effectively enhances the model's capability for Micro-gesture Online Recognition, achieving an experimental result of 14.34 on the SMG dataset. In future work, we will consider incorporating skeletal data into the model to enhance its recognition ability for Micro-gesture Online Recognition.

Acknowledgments

This work was supported by the National Key R&D Program of China (2022YFB4500601), the National Natural Science Foundation of China (62272144,72188101,62020106007 and U20A20183), the Major Project of Anhui Province (202203a05020011), and the Fundamental Research Funds for the Central Universities (JZ2024HGTG0309).

References

[1] H. Chen, X. Liu, X. Li, H. Shi, G. Zhao, Analyze spontaneous gestures for emotional stress state recognition: A micro-gesture dataset and analysis with deep learning, in: 2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019), 2019, pp. 1–8.

[2] 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) 1346–1366.

[3] K. Li, D. Guo, G. Chen, X. Peng, M. Wang, Joint skeletal and semantic embedding loss for micro-gesture classification, arXiv preprint arXiv:2307.10624 (2023).

[4] D. Guo, K. Li, B. Hu, Y. Zhang, M. Wang, Benchmarking micro-action recognition: Dataset, methods, and applications, IEEE Transactions on Circuits and Systems for Video Technology 34 (2024) 6238–6252.

[5] S. Tang, R. Hong, D. Guo, M. Wang, Gloss semantic-enhanced network with online back-translation for sign language production, in: Proceedings of the 30th ACM International Conference on Multimedia, 2022, pp. 5630–5638.

[6] 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, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 10631–10642.

[7] J. Tan, X. Zhao, X. Shi, B. Kang, L. Wang, Pointtad: Multi-label temporal action detection with learnable query points, Advances in Neural Information Processing Systems 35 (2022) 15268–15280.

[8] K. Li, D. Guo, M. Wang, Proposal-free video grounding with contextual pyramid network, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2021, pp. 1902–1910.

[9] K. Li, D. Guo, M. Wang, Vigt: proposal-free video grounding with a learnable token in the transformer, Science China Information Sciences 66 (2023) 202102.

[10] X. Guo, W. Peng, H. Huang, Z. Xia, Micro-gesture online recognition with graph-convolution and multiscale transformers for long sequence (2023).

[11] A. Piergiovanni, M. S. Ryoo, Learning latent super-events to detect multiple activities in videos, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5304–5313.

[12] P. Tirupattur, K. Duarte, Y. S. Rawat, M. Shah, Modeling multi-label action dependencies for temporal action localization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 1460–1470.

[13] R. Dai, S. Das, K. Kahatapitiya, M. S. Ryoo, F. Brémond, Ms-tct: Multi-scale temporal convtransformer for action detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 20041–20051.

[14] Z. Wu, K. Zhang, H. Xuan, J. Yang, Y. Yan, Dapc-net: Deformable alignment and pyramid context completion networks for video inpainting, IEEE Signal Processing Letters 28 (2021) 1145–1149.

[15] Z. Wu, C. Sun, H. Xuan, G. Liu, Y. Yan, Waveformer: Wavelet transformer for noise-robust video inpainting, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 38, 2024, pp. 6180–6188.

[16] Z. Wu, C. Sun, H. Xuan, Y. Yan, Deep stereo video inpainting, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 5693–5702.
[17] J. Zhou, D. Guo, Y. Zhong, M. Wang, Advancing weakly-supervised audio-visual video parsing via segment-wise pseudo labeling, arXiv preprint arXiv:2406.00919 (2024).
[18] Y. Wei, Z. Zhang, M. Xu, R. Hong, J. Fan, S. Yan, Robust attention deraining network for synchronous rain streaks and raindrops removal, in: Proceedings of the 30th ACM International Conference on Multimedia, 2022, pp. 6464–6472.
[19] A. Gu, K. Goel, C. Ré, Efficiently modeling long sequences with structured state spaces, arXiv preprint arXiv:2111.00396 (2021).
[20] A. Gu, I. Johnson, K. Goel, K. Saab, T. Dao, A. Rudra, C. Ré, Combining recurrent, convolutional, and continuous-time models with linear state space layers, Advances in neural information processing systems 34 (2021) 572–585.
[21] A. Gu, T. Dao, Mamba: Linear-time sequence modeling with selective state spaces, arXiv preprint arXiv:2312.00752 (2023).
[22] S. Shams, S. S. Dindar, X. Jiang, N. Mesgarani, Ssamba: Self-supervised audio representation learning with mamba state space model, arXiv preprint arXiv:2405.11831 (2024).
[23] J. Carreira, A. Zisserman, Quo vadis, action recognition? a new model and the kinetics dataset, in: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6299–6308.
[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Proceedings of the Advances in Neural Information Processing Systems 30 (2017).
[25] F. Wang, D. Guo, K. Li, M. Wang, Eulermormer: Robust eulerian motion magnification via dynamic filtering within transformer, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 38, 2024, pp. 5345–5353.
[26] F. Wang, D. Guo, K. Li, Z. Zhong, M. Wang, Frequency decoupling for motion magnification via multi-level isomorphic architecture, arXiv preprint arXiv:2403.07347 (2024).
[27] Z. Wu, H. Xuan, C. Sun, W. Guan, K. Zhang, Y. Yan, Semi-supervised video inpainting with cycle consistency constraints, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 22586–22595.
[28] Z. Wu, C. Sun, H. Xuan, K. Zhang, Y. Yan, Divide-and-conquer completion network for video inpainting, IEEE Transactions on Circuits and Systems for Video Technology 33 (2023) 2753–2766.
[29] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, et al., The kinetics human action video dataset, arXiv preprint arXiv:1705.06950 (2017).