Journal of Computers Vol. 37 No. 5, October 2026, pp. 443-457
https://doi.org/10.63367/19911599202610370502 7

Leveraging Mamba for Enhanced Multimodal Few-Shot Action Recognition

Zhaochuan Wei[1], Rongtian Yang[2], Xiaodong Cai[1], and Yuanfa Ji[2*]

[1] School of Information and Communication Guilin University of Electronic Technology, Guilin, China
{weizc, caixiaodong}@guet.edu.cn

[2] Guangxi Key Laboratory of Precision Navigation Technology and Application, Guilin University of Electronic Technology
{1208313279, 104806156}@qq.com

Received 15 July 2025; Revised 27 January 2026; Accepted 28 January 2026

Abstract. Few-Shot Action Recognition (FSAR) widely use in Action Recognition domains with limited labeled data, such as medical surgery. Existing FSAR methods often rely on single-modal approaches, which suffer from limited feature representation and a lack of semantic understanding. To overcome these limitations, we propose MoMA, a novel Mamba-based multimodal FSAR framework, designed to optimize the learning of multimodal information and improve the performance of few-shot action recognition tasks. MoMA integrates a multimodal large language model (MLLM) to generate semantically rich representations from both textual and video data, which are then aligned and fused with visual features. To address the challenges of domain-specific prior knowledge and intra-class variance, we introduce a dual-branch contrastive learning architecture that reinforces category distinctions and mitigates the accumulation of intra-class heterogeneity. Our experimental results show that MoMA outperforms state-of-the-art methods on the HMDB51, Kinetics, and SSv2 datasets, achieving improvements of 1.5%, 1.9%, and 3.5%, respectively, in the 1-shot setting.

Keywords: few-shot action recognition, multimodal large language models, Mamba, contrastive learning

1 Introduction

Action recognition is a crucial task in computer vision, focused on identifying the category of actions in a given video or video clip. However, data collection can be particularly challenging in certain domains, such as medical surgery, where only a limited number of samples are available. To address this challenge, Few-Shot Action Recognition (FSAR) was introduced, enabling models to learn to recognize new action categories with only a few labeled samples—ranging from 1 to 5 samples per category. This approach is broadly used in applications such as autonomous driving, human-computer interaction, and smart city management [1-4].

Early few-shot action recognition primarily focused on a single visual modality, where a universal distance metric function was learned to bring similar actions closer in the feature space while separating dissimilar actions. For instance, TRX [10] proposed a temporal-relational cross Transformer for temporal modeling; PAL [11] utilized a prototype-centered attention mechanism; Cao et al. [12] achieved few-shot video classification via temporal alignment; Zhang et al. [13] focused on implicit temporal alignment; and TA2N [14] introduced a two-stage action alignment network. However, single-modal methods suffer from limited feature representation capabilities and a lack of semantic information.

To overcome these limitations, researchers began exploring multimodal information fusion strategies. ActionCLIP [20], proposed by Wang et al., utilized action category labels as text inputs to construct a video-text contrastive learning framework. Xiang et al. [21] leveraged large language models to generate descriptive text based on action labels, providing prior knowledge for skeleton action recognition. Xu et al. [22] enhanced skeleton action recognition by using language knowledge-assisted representation learning. However, these approaches are mostly based on simple text modalities and unimodal large language models, generating text features solely from action labels. This limits their generalization ability for unseen action categories. The advent of multimodal large language models, such as BLIP [23], BLIP2 [24], MiniGPT-4 [25], and LLaMA-AdapterV2 [26], provides new opportunities for generating richer multimodal semantic representations. These methods are typically combined with Transformer architectures for temporal modeling. However, due to the high

* Corresponding Author

computational complexity of Transformers, they can only process short subsequences (around 8 frames) and struggle with long-term temporal dependencies.

The Mamba architecture [5] is a novel state-space model that introduces input-dependent state transitions through selective parameterization, optimizing the restrictions of traditional state-space models. Recently, several studies have applied Mamba to computer vision tasks. For example, ViM [27] integrated Mamba into the Vision Transformer model, while VMamba [28] built a hierarchical structure using Mamba and ResNet [29]. Additionally, EfficientVMamba [30] introduced a selective scanning mechanism to improve efficiency. However, these methods still rely on abundant labeled data for few-shot action recognition tasks involving multimodal information and perform poorly when dealing with unseen action categories.

Our basic idea is to use Mamba to optimize the inference speed of multimodal FSAR. However, this approach faces two main challenges. First, although Mamba excels at capturing long-range dependencies, it lacks mechanisms for incorporating domain-specific prior knowledge. For example, it does not consider biomechanical constraints on human joint motion or the temporal dependencies between different phases of an action. This limitation can lead to ambiguous feature representations that are not sufficiently tailored to the task. Second, variations in recording conditions (e.g., viewpoint) and post-processing artifacts introduce intra-class heterogeneity, even among well-aligned action samples (e.g., "Cliff Diving"). As the sequence length increases, these disparities accumulate, exacerbating intra-class variance and diminishing the separability between different action classes. As a result, the model struggles to form compact clusters for semantically similar actions and fails to maintain clear decision boundaries between distinct categories, ultimately degrading classification accuracy [19].

To address these challenges, we propose MoMA (Modeling of Multimodal Action Recognition), a novel Mamba-based multimodal FSAR framework. First, we leverage a MLLM (multimodal large language model) to generate semantically rich textual descriptions and high-level representations from videos. These representations are then processed through dedicated textual and structured data encoders and fused with visual features extracted from image-based models to achieve aligned multimodal information. Next, we integrate feedforward networks and normalized residual structures into the Mamba backbone to compensate for the lack of domain-specific prior knowledge in the original Mamba, thereby facilitating the training of fused multimodal representations. Finally, we introduce a joint multimodal contrastive learning paradigm that integrates modality-aware contrast and class-supervised contrast. This paradigm is designed to mitigate the accumulation of intra-class variance and inter-class ambiguity, ultimately improving the robustness of the learned representations.

Organization of the paper. The remainder of this paper is organized as follows. Section 2 reviews related work on few-shot action recognition, multimodal learning, and Mamba-based sequence modeling. Section 3 presents the proposed MoMA framework, including multimodal information construction, the multimodal Mamba module, and the dual-branch contrastive objective. Section 4 reports experimental settings and results on HMDB51, Kinetics, and SSv2, together with ablation and robustness analyses. Finally, Section 5 concludes the paper and discusses future directions.

Our main contributions are summarized as follows:

(1) We introduced Mamba into the multimodal FSAR framework, demonstrating that multimodal semantic information can compensate for Mamba's prior knowledge. This approach enhances its learning capacity, improves training stability, and ensures consistent reasoning speed.

(2) We proposed MoMA, a method that leverages the multimodal learning capabilities of MLLMs to extract both text and video information, thereby augmenting Mamba's prior knowledge. To further boost performance, we incorporated a dual-branch contrastive learning architecture. This architecture aligns cross-modal features and reinforces category distinctions, effectively improving the model's generalization ability in small sample environments.

(3) Our experimental results show that MoMA outperforms existing state-of-the-art methods on the HMDB51, Kinetics, and SSv2 datasets. Specifically, MoMA achieves improvements of 5.67%, 5.00%, and 6.77%, respectively, in the 1-shot setting.

2 Related Work

2.1 Few-Shot Action Recognition

FSAR primarily employs a meta-learning approach based on metric learning to learn an effective distance metric.

Early methods like OTAM [11] utilized the dynamic time warping (DTW) algorithm to calculate the similarity between query and support video subsequences, matching them through temporal alignment. The ITANe [12] model proposed an intrinsic temporal attention network to enhance action representation by learning the temporal dependencies between frames. T2AN [13] introduced a time-to-time attention mechanism that captures action patterns across different time scales, while STRM [14] employed a spatiotemporal relationship modeling network, accounting for both the spatial structure and temporal evolution of actions and modeling spatiotemporal dependencies through a relational reasoning module. However, these methods suffer from the limitation of insufficient semantic information extracted from a single modality, which can hinder the model's recognition accuracy.

To address these challenges, some studies have introduced additional auxiliary information to enrich feature representations. For example, Fu et al. [15] proposed a depth-guided adaptive meta-fusion network that leverages geometric information from depth maps to enhance video representations, effectively integrating RGB and depth information through a multimodal fusion strategy. Optical flow information [16] has also been used to capture motion patterns, enhancing the temporal characteristics of actions by analyzing pixel-level motion trajectories. Furthermore, motion information enhancement methods [17-18] explicitly model the motion of human joints or overall motion patterns, providing richer descriptions of actions. However, the fusion of different modalities in these models can be rigid and may even lead to negative transfer in some cases.

The introduction of MLLMs has advanced the field. For example, ActionCLIP [20] generates semantically rich text embeddings from action labels via the pre-trained visual-language model CLIP. Xiang et al. [21] guided a large language model to produce descriptive text containing details such as action specifics and body part involvement through prompt engineering, which was then encoded into feature vectors for enhanced skeleton action recognition. Similarly, Xu et al. [22] used a language knowledge-assisted representation learning strategy, extracting action-related semantic knowledge from a pre-trained language model and integrating it into skeleton action recognition. With the rise of multimodal large language models like BLIP [23], BLIP2 [24], MiniGPT-4 [25], and LLaMA-AdapterV2 [26], these models generate detailed textual descriptions directly from visual content, eliminating the reliance on predefined action labels. However, many of these methods still depend on Transformer architectures for temporal modeling. Due to the high computational complexity of Transformers, they can typically process only short temporal sequences, making it difficult to model long temporal dependencies in extended sequences. Additionally, existing approaches face challenges like intra-class variance accumulation in few-shot scenarios.

2.2 Mamba Architecture

Mamba is a novel state-space model (SSM) architecture that introduces input-dependent state transitions through selective parameterization, enabling context-aware sequence modeling with linear computational complexity [5-7]. An increasing number of studies have begun applying Mamba to computer vision tasks. For instance, ViM [27], inspired by the Vision Transformer, was the first to integrate the Mamba architecture into a visual model. By utilizing bidirectional state-space modeling, it captures spatial dependencies between image patches while maintaining linear complexity, thus enabling efficient processing of high-resolution images. Similarly, inspired by the ResNet architecture, VMamba [28] developed a hierarchical structure based on Mamba to handle multi-scale visual features. EfficientVMamba [30] further improved upon VMamba by introducing a selective scanning mechanism. This method uses a gating function to determine which spatial positions should be updated, thus avoiding the full calculation of all positions. However, these methods still lack prior knowledge when applied to few-shot action recognition tasks that involve multimodal information.

2.3 Contrastive Learning

As a paradigm of self-supervised representation learning, contrastive learning trains models by maximizing the similarity between semantically similar instances while minimizing the similarity between unrelated ones [36]. This approach is designed to enhance the ability of deep learning models to learn general representations from unlabeled data [30-31]. SupCon [32] introduced label information into the contrastive learning framework, maximizing the similarity between samples of the same class and minimizing the similarity between samples of different classes, thereby learning feature representations with stronger class discrimination. Gidaris et al. [33] combined contrastive learning with self-supervised tasks, such as rotation prediction, allowing the

model to learn rich feature representations even with limited labeled data. Zheng et al. [34] proposed a few-shot action recognition method that combines hierarchical matching with contrastive learning, enhancing the model's discriminative ability through multi-level feature comparisons. Su et al. [35] explored the conditions and mechanisms for effective self-supervised learning to improve few-shot learning. Their work found that contrastive learning effectively reduces intra-class variance in the feature space while increasing inter-class separation.

3 Method

3.1 Problem Definition

In the context of few-shot learning, the widely used N-way K-shot configuration consists of two sets: the support set $S_k^N = \{S_1^1, ..., S_k^1, ..., S_1^N, ..., S_k^N\}$ and the query set $Q_M^N = \{q_1^1, ..., q_M^1, ..., q_1^N, ..., q_M^N\}$. Here, N represents the number of action classes, K denotes the number of video samples per class in the support set, and M corresponds to the number of video samples per class in the query set. The objective of few-shot action recognition is to enable a model to make accurate predictions using only a limited number of labeled examples. Specifically, given a query video V_i from an unseen class, the model is expected to classify it by comparing it with the most similar samples s_i^k in the support set, thereby identifying the correct action category. The formula for classification is:

$$\hat{y} = \arg\min_{y\in\{1,\ldots,N\}} \sum_{i=1}^{M} d\left(q_i^y, s_i^y\right) \tag{1}$$

where $d(\cdot,\cdot)$ represents a distance metric, and $\hat{y}$ is the predicted class for the query video. As illustrated in Fig. 1, our framework adopts a dual-branch architecture that integrates a multimodal Mamba branch and a joint contrastive learning branch to address the challenges of few-shot action recognition. The Mamba branch enhances the original architecture by incorporating feedforward networks with normalized residual connections, which improves its ability to model action patterns. Simultaneously, the contrastive learning branch introduces a novel dual-path mechanism. This mechanism is designed to align cross-modal feature distributions and enhance the model's discriminative power. Specifically, it combines modality-aware contrastive learning, which aligns features from different modalities (e.g., video and text), with class-supervised contrastive learning, which strengthens intra-class compactness and inter-class separability. These two complementary mechanisms work in tandem to optimize the learning process, enabling end-to-end training of the entire framework.

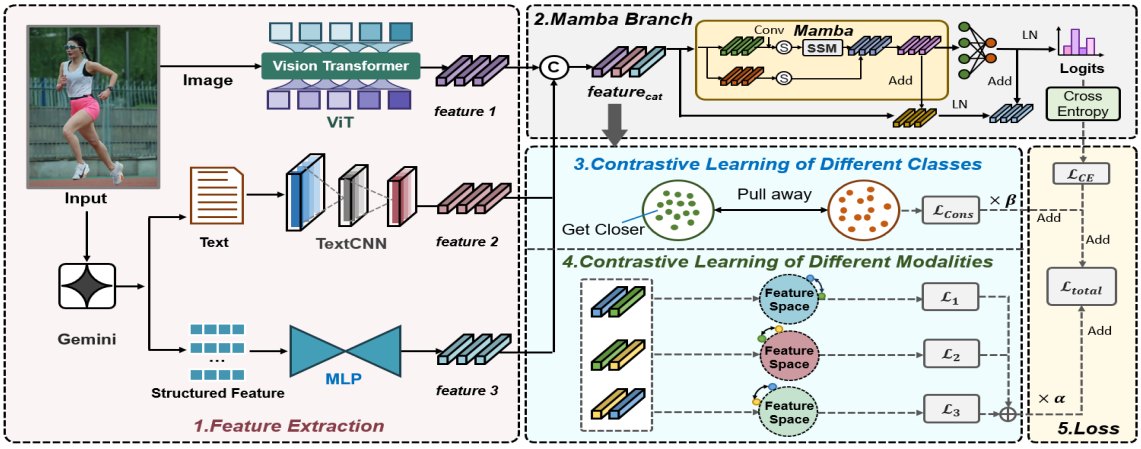


Fig 1. General architecture of multimodal mamba and comparative learning framework

3.2 Multimodal Information Construction

In the task of few-shot action recognition, traditional unimodal methods often lead to a lack of semantic information and limited model accuracy. To overcome these limitations, we propose a multimodal information construction strategy based on large model feature enhancement. By leveraging the semantic understanding ability of MLLM, we extract rich multimodal representations from raw video data. Given an original video sequence $V = \{f_1, f_2, \ldots, f_T\}$, where T represents the total number of video frames, we first apply a uniform sampling strategy to extract key frames from the video $\mathcal{F} = \{f_{i1}, f_{i2}, ..., f_{iK}\}$, where i_j is calculate as:

$$i_j = | j \cdot \frac{T}{K} | \tag{2}$$

where K denotes the number of sampled frames. This uniform sampling strategy ensures the integrity of the temporal information while controlling computational complexity. The sampled image data is represented as $x_1 \in \mathbb{R}^{B\times C\times H\times W}$, where B is the batch size, C is the number of channels, H is the height of the image, and W is the width of the image. We use the MLLM to perform semantic analysis on the sampled frames and generate corresponding text descriptions. Specifically, for each input frame f_i, the MLLM generates a natural language description d_i containing information such as action semantics, scene description, and object interaction. We also designed prompt templates to guide the Multimodal Large Language Model (MLLM) in task execution. Descriptive Caption Prompt: "Describe the primary human action in the given frame in one concise sentence ($\leq$ 20 words) [8], including key interacting objects and scene context." The mathematical representation of the text features is given by $x_2 \in \mathbb{R}^{B\times T}$, where T represents the maximum length of the text sequence. The text generation process can be formalized as follows:

$$d_i = \mathrm{MLLM}_{\mathrm{text}}(f_i; \theta_{\mathrm{text}}) \tag{3}$$

where θ_{text} denotes the parameters of the text generation module. In addition to generating descriptive text, we also directly obtain a structured high-dimensional embedding from the multimodal large language model (MLLM) and treat it as the native embedding output of the model. Specifically, conditioned on the fixed instruction prompt ϕemb and the sampled frame Ii, the MLLM returns a ds-dimensional global multimodal embedding vector that encapsulates high-level semantic information (including actions, objects, scenes, and key body parts). We use fixed instruction prompts across all datasets and all episodes: "Focus on action semantic information: action type, involved objects, scene context, and key body parts. Generate a compact semantic representation. "The structured feature extraction process is:

$$e_i = \mathrm{MLLM}_{\mathrm{embed}}(f_i; \theta_{\mathrm{embed}}) \tag{4}$$

To convert the raw data from different modalities into a unified feature space, we design three specialized encoding modules. The visual encoder ε_v maps the raw image data to the feature space as follows:

$$x_{1'} = \mathcal{E}_v(x_1; \theta_v) \in \mathbb{R}^{B\times F} \tag{5}$$

where θ_v represents the visual encoder parameters and F is the unified feature dimension. The visual encoder adopts the Vision Transformer (ViT) architecture, which is effective at extracting spatial hierarchical features. The text encoder ε_t processes the natural language descriptions and maps them to the same feature space:

$$x_{2'} = \mathcal{E}_t(x_2; \theta_t) \in \mathbb{R}^{B\times F} \tag{6}$$

The text encoder utilizes the TextCNN architecture, which captures long-distance dependencies in text sequences through a self-attention mechanism. The structured encoder ε_s processes the high-dimensional embedding vectors:

$$x_{3'} = \mathcal{E}_s(x_3; \theta_s) \in \mathbb{R}^{B \times F} \tag{7}$$

The encoder uses a multi-layer perceptron (MLP) structure to achieve nonlinear mapping from the high-dimensional embedding space to the unified feature space. After processing by their respective encoders, the features from the three modalities are aligned in the same feature space with dimension F. We then apply a feature stacking strategy to combine these modalities:

$$X = \text{concat}(x_{1'}, x_{2'}, x_{3'}) \in \mathbb{R}^{B \times 3 \times F} \tag{8}$$

This feature stacking ensures that the features from each modality remain relatively independent, avoiding potential information loss from early fusion. This provides a structured input format for the subsequent multimodal Mamba layer.

3.3 Mamba Branch

The original Mamba architecture lacks the inductive bias necessary for action recognition tasks, making it insufficient for effectively utilizing domain-specific knowledge, such as human kinematic constraints. As shown in Fig. 2, to address these limitations we designed a multimodal Mamba branch that incorporates enhancement, residual, and normalization mechanisms. Compared with the standard module design, our proposed module has the following characteristics: it adds an explicit feed-forward network (FFN) sublayer for channel mixing and high-order interactions; it adopts two independent "LayerNorm + residual connection" paths (one is placed after the Mamba operation, and the other is placed after the FFN operation); it retains the advantage of linear-time complexity — because the computational costs of both Mamba and FFN scale with the parameter size at a rate of O(Ld). These mechanisms are aimed at improving the representation capability of the model and enhancing its training stability, while preserving the linear complexity advantage inherent to Mamba. The entire multimodal Mamba module can be described as a composite function of three stages:

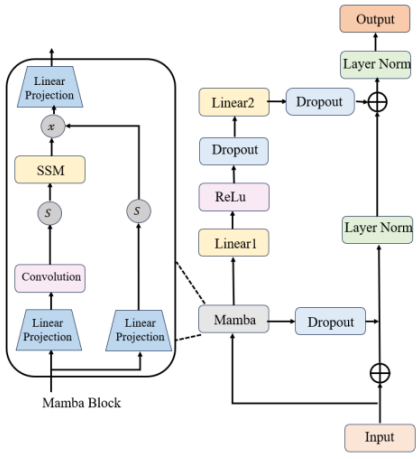


Fig 2. Architecture of multimodal Mamba

$$X = \text{LayerNorm}_2(H_1 + \text{Drop}_2(\text{FFN}(H_1))) \tag{9}$$

Where H_1 can be formulate as:

$$H_1 = \text{LayerNorm}_1(X + \text{Drop}_1(\text{Mamba}(X))) \tag{10}$$

The state-space model, as a type of linear time-invariant (LTI) system, maps the input stimulus $x(t)\in\mathbb{R}^L$ to the output response through the hidden state space $h(t)\in\mathbb{R}^N$. This relationship is modeled by the following system of linear ordinary differential equations:

$$\frac{dh(t)}{dt} = Ah(t) + Bx(t) \tag{11}$$

$$y(t) = Ch(t) \tag{12}$$

where $A \in\mathbb{R}^{N\times N}$ is the state evolution parameter matrix, which governs the intrinsic dynamics of the hidden state; $B \in\mathbb{R}^{N\times 1}$ is the input projection parameter vector, which maps the input to the state space; and $C \in \mathbb{R}^{1\times N}$ is the output projection parameter vector, which maps the state to the output space. In practice, the continuous-time model needs to be discretized to obtain a discrete-time state-space model, which is given by:

$$h_k = \overline{A}h_{k-1} + \overline{B}x_k,\ y_k = Ch_k \tag{13}$$

Where:

$$\overline{A} = \exp(\Delta A), \quad \overline{B} = (\Delta A)^{-1}(\exp(\Delta A) - I)\cdot \Delta B \tag{14}$$

In this discretization, Δ represents the sampling step size. While traditional state-space models use fixed parameter matrices, Mamba introduces a selective mechanism that allows parameters to be adaptively adjusted according to the input content. This selective mechanism is a key innovation of Mamba over conventional state-space models. Specifically, the adaptive parameters are as follows:

$$B_k = s_B(x_k), \quad C_k = s_C(x_k), \quad \Delta_k = s_\Delta(x_k) \tag{15}$$

Where s_B, s_C, and s_Δ are learnable selection functions, typically implemented through simple linear layers. This selectivity mechanism allows Mamba to adjust its parameters based on the input, making it more flexible and effective for diverse tasks. The feed-forward network (FFN) in Mamba is responsible for enhancing the nonlinear representation capacity of the model. The FFN consists of two linear layers, with a ReLU activation function in between. The transformations in each layer are as follows. First layer transformation:

$$H_{ffn} = \mathrm{ReLU}(X_1W_1 + b_1) \in \mathbb{R}^{B\times N\times d_{ff}} \tag{16}$$

Second layer transformation:

$$F = H_{ffn}W_2 + b_2 \in \mathbb{R}^{B\times N\times D} \tag{17}$$

Where d_{ff} = 2048 is the extended dimension used for high-dimensional space transformation. $W_1\in\mathbb{R}^{D\times dff}$ is the weight matrix for the first layer, $W_2\in\mathbb{R}^{dff\times D}$ is the weight matrix for the second layer, and b_1, b_2 are the corresponding bias vectors for each layer. Layer normalization is applied to ensure the stability and consistency of the feature distributions. The formula for LayerNorm is as follows:

$$\mathrm{LayerNorm}(x) = \gamma\cdot\frac{x-\mu}{\sqrt{\sigma^2+\epsilon}} + \beta \tag{18}$$

Where μ is the mean, σ^2 is the variance, and γ,β are learnable scaling and offset parameters. In our

architecture, we use two LayerNorm layers: the first is applied after the Mamba output to ensure the numerical stability of the state-space model output, while the second is applied after the FFN output to maintain the distribution characteristics of the final output.

3.4 Model Optimization Objectives

In the task of FSAR, intra-class variance accumulation impacts the recognition accuracy of the model. Contrastive learning addresses this issue by optimizing the geometric structure of the feature space, bringing positive samples closer and pushing negative samples farther apart. Additionally, it handles the alignment problem of multimodal data, making it an ideal tool for multimodal FSAR. We propose two contrastive learning branches to tackle these challenges: the multimodal contrastive learning branch, which focuses on cross-modal feature alignment, and the class-supervised contrastive learning branch, which aims to enhance intra-class aggregation and inter-class separation. Given an input instance x_i, the corresponding image embedding and text embedding are denoted as v_i and t_i, respectively. The goal of multimodal contrastive learning is to bring embeddings from the same instance closer, while pushing representations from different instances further apart. The multimodal contrastive learning loss function is defined as:

$$\mathcal{L}_{\text{multimodal}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(s(v_i,t_i)/\tau)}{\sum_{j=1}^{N}\exp(s(v_i,t_j)/\tau)} \tag{19}$$

where $s(v_i, t_i)$ is the similarity score between the embeddings v_i and t_i, and τ is the temperature parameter that controls the smoothness of the distribution. The batch size is denoted by N. To calculate the similarity, we use cosine similarity as the basic metric:

$$s(v_i,t_j) = \frac{v_i \cdot t_j}{\| v_i\|_2 \| t_j\|_2} \tag{20}$$

where v_i and t_j are the image and text embeddings, respectively, and $\|\cdot\|_2$ denotes the L2 norm. On the other hand, class-supervised contrastive learning aims to explicitly optimize the category structure of the feature space by utilizing label information. This loss function is defined as:

$$\mathcal{L}_{\text{supervised}} = \sum_{i=1}^{N}\frac{-1}{|P(i)|}\sum_{p\in P(i)}\log\frac{\exp(z_i \cdot z_p/\tau)}{\sum_{a\in A(i)}\exp(z_i \cdot z_a/\tau)} \tag{21}$$

where z_i is the embedding of instance x_i after backbone network encoding, $P(i)$ is the set of all samples in the same category as the anchor sample x_i, and $A(i)$ represents the set of all samples in the batch, excluding the anchor x_i. This loss function explicitly forces the model to cluster instances of the same category while separating those of different categories. Finally, the overall loss function of the MoMA framework is a weighted sum of several loss components, which balances the contributions from classification, contrastive learning, and consistency regularization. The total loss is formulated as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{ce}} + \beta\mathcal{L}_{\text{CONS}} + \alpha(\mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3) \tag{22}$$

Here, $\mathcal{L}_{\text{ce}}$ is the cross-entropy loss that provides basic classification supervision, $\mathcal{L}_{\text{CONS}}$ is the consistency loss that ensures the alignment of multimodal predictions, and $\mathcal{L}_1$, $\mathcal{L}_2$, and $\mathcal{L}_3$ are the contrastive learning losses for the three modalities (image, text, and structured embeddings). The terms α and β are the weight coefficients that control the contributions of the different loss components.

4 Experiments

4.1 Experiment Setting

Dataset: We employ three widely used benchmarks for the FSAR task: SSv2 [37], Kinetics [38], and HMDB51 [39]. The SSv2 dataset is a large-scale video dataset that contains diverse human actions, while Kinetics provides a broad range of action categories across various video clips, and HMDB51 focuses on a smaller set of 51 action categories with more challenging video samples. When decoding the videos, the sampling interval was set to 1 frame per second. The datasets are split into training, validation, and testing sets. The objective of the FSAR task is to classify the query sample Q_r into the corresponding class of the support set S. Based on TSN [40], each frame is resized to 3×256×256, and F frames are extracted sequentially from the video stream. During training, data augmentation techniques include random cropping to 3×224×224 and horizontal flipping, while only center cropping is used during testing. Since many actions in the SSv2 dataset are lateral, horizontal flipping was excluded to preserve the directional integrity of actions like "pushing S from right to left" [11].

Environment Configuration: Experiments were conducted on a server equipped with an Intel Xeon Platinum 8358 CPU, a 32GB NVIDIA Tesla V100 PCIe GPU, and a 10TB NVMe SSD RAID array. The operating system is Ubuntu 22.04 LTS, and Python version 3.9.16 was used.

Baseline: The models involved in the comparison can be grouped into several categories based on their methodologies. Temporal modeling-based methods include STRM, TRX, and HyESM, which focus on capturing temporal dependencies through various attention mechanisms. Feature alignment and fusion techniques are represented by SloshNet, GCSM, and GgHM, which aim to align and integrate multimodal features for improved recognition. Contrastive learning-based methods such as SA-CT, MoLo, and Lite-MKD leverage contrastive learning strategies for learning robust feature representations. Additionally, action recognition methods based on graph convolution and attention mechanisms include AmeFu-Net, MTFAN, and AMFAR, each using specialized graph structures for action modeling.

Hyperparameters: In our experiments, we employed two commonly used few-shot learning configurations: 5-way 1-shot and 5-way 5-shot. To ensure a comprehensive comparison, we selected TextCNN, ViT, and MLP as the backbone architectures. The extracted features were represented as 2048-dimensional vectors (D = 2048). For training, the larger-scale SSv2 dataset required 75,000 instances, while the remaining datasets were trained using 15,000 few-shot tasks. Optimization was performed using stochastic gradient descent (SGD) with an initial learning rate of 10^{-3}. Hyperparameter tuning was carried out using the validation set (D_{val}), with the temperature parameter set to τ = 0.07 and the loss weight factors set to α = 0.1 and β = 0.1. During the testing phase, the average accuracy was reported over 10,000 random tasks sampled from the test set.

4.2 Main Results

In this study, we conducted experiments on SSv2, Kinetics, and HMDB51, using 1-shot and 5-shot few-shot learning settings. The objective was to compare the Accuracy performance of our proposed MoMA method against the current state-of-the-art methods such as AMFAR, STRM, SloshNet, and others.

As shown in Table 1, our MoMA method demonstrates improvements in accuracy across all three datasets. For example, in the 1-shot setting on the SSv2 dataset, MoMA improves the accuracy of the state-of-the-art method AMFAR from 56.6% to 60.1%, a 3.5% increase. Similar improvements are observed across the other datasets under multiple few-shot learning settings, with notable increases in accuracy on Kinetics and HMDB51 datasets as well. These results confirm that MoMA achieving enhanced recognition accuracy under few-shot learning conditions.

An interesting finding from the experiments is the impact of frame-level noise on model performance. To evaluate the robustness of the model against temporal corruption, we construct a noisy frame sequence by replacing a subset of sampled frames with irrelevant frames while keeping the sequence length unchanged. Given T sequentially sampled frames $\{f_t\}_{t=1}^{T}$ from a video clip, we first select a noise ratio $\rho \in \{0, 0.1, 0.2, 0.3, 0.4\}$, and then obtain a set of indices $\Omega \subset \{1, ..., T\}$ containing $M = \lfloor \rho T \rfloor$ elements through uniform sampling. For each index $t \in \Omega$, the original frame f_t is replaced with a noisy frame f_t^{noise}; this noisy frame is extracted from a random video of a different action class in the same dataset split (training/validation/test set) and sampled at a random timestamp of that video. The resulting noisy sequence satisfies the following condition: $\tilde{f}_t = f_t$ when $t \notin \Omega$ $\tilde{f}_t = f_t^{\text{noise}}$ when $t \in \Omega$. As shown in Fig. 3 and Fig. 4, the experimental results indicate that the accumulation

of intra-class variance—especially over long subsequences—exerts a negative impact on model performance. Specifically, methods such as TRX, HyRSM and MoLo exhibit a noticeable decline in accuracy when dealing with frame-level noise. In contrast, our proposed MoMA method demonstrates stronger resilience, maintaining high accuracy even in the presence of noisy frames.

Table 1. Comparison of experimental results of various models

Methods	SSV2		Kinetics		HMDB51	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
STRM [14]	N/A	68.1	N/A	86.7	N/A	76.3
SloshNet [41]	46.5	68.3	N/A	87.0	N/A	77.5
SA-CT [42]	48.9	69.1	71.9	87.1	61.2	76.9
GCSM [43]	N/A	N/A	74.2	88.2	61.3	79.3
GgHM [44]	54.5	69.2	74.9	87.4	61.2	76.9
TRX [9]	53.8	68.8	74.9	85.9	63.5	75.8
HyRSM [45]	54.1	68.7	73.5	86.2	60.1	76.2
MoLo [17]	56.6	70.7	74.2	85.7	67.1	77.3
AmeFu-Net [15]	N/A	N/A	74.1	86.8	60.2	75.5
MTFAN [18]	45.7	60.4	74.6	87.4	59.0	74.6
Lite-MKD [46]	55.7	69.9	75.0	87.5	66.9	74.7
MoMA (Ours)	60.1	73.8	77.3	88.9	68.6	78.3

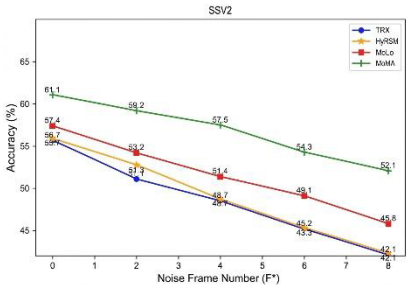


Fig 3. Comparison of SSV2 datasets under more complex tasks

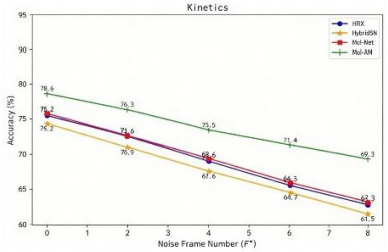


Fig 4. Comparison of Kinetics datasets under more complex tasks

The experimental results are summarized in Table 2, which assess the efficiency of MoMA in terms of inference time, training time, memory usage, and floating-point operations (FLOPs) relative to the state-of-the-art methods. As observed in the results, MoMA outperforms the fastest baseline method, Lite-MKD, in inference time, processing 10,000 videos in only 4.62 hours compared to 5.67 hours for Lite-MKD. This corresponds to an 18.6% improvement in inference time. More importantly, this advantage becomes even more pronounced as the sequence length increases, indicating that MoMA is better equipped to handle longer video sequences efficiently. Additionally, MoMA demonstrates lower memory usage (6.1 GB) and fewer FLOPs (67.8) than other methods, depicts its computational efficiency.

Table 2. Model training speed comparison

Method	Inference time	Training time	Memory	FLOPs
TRX	8.67	2.8	8.7	125.4
AMFAR	9.71	3.5	11.2	178.9
MoLo	7.83	2.6	7.9	112.6
HyRSM	6.33	3.1	9.4	145.7
Lite-MKD	5.67	2.4	7.2	98.3
MoMA	4.62	1.8	6.1	67.8

4.3 Ablation Experiment

Impact of Different Modules: In this study, we perform an ablation analysis of the proposed MoMA model to assess the individual impact of each component and determine how they work together to enhance performance. The experimental results, as shown in Fig. 5 and Fig. 6, demonstrate that incorporating all key components of MoMA leads to performance gains. For instance, in the 1-shot setting on the Kinetics dataset, applying the full MoMA model improves accuracy from 66.3% to 77.3%, a 11% increase. Similarly, in the 5-shot setting on SSv2, the full MoMA model improves accuracy from 58.8% to 73.8%, a 15% improvement. These results clearly indicate that combining all components of MoMA yields the best performance. The multimodal information construction module (MI) and the attention mechanism module (AM) both show a noticeable impact on performance, particularly when combined with contrastive learning (CL). However, when any one of these components is removed, performance declines. For example, removing the contrastive learning module (w/o CL) leads to a decrease in performance, which is more noticeable in the 1-shot setting. This depicts that contrastive learning enhances feature alignment and reduces intra-class variance. Similarly, when the attention mechanism module (AM) is removed, performance suffers slightly, demonstrating the importance of spatial and temporal attention mechanisms in modeling action recognition tasks. Overall, MoMA's design effectively combines the strengths of each module to optimize action recognition performance. The combination of multimodal information, contrastive learning, and attention mechanisms enables MoMA to reduce computational complexity, mitigate the effects of intra-class variance.

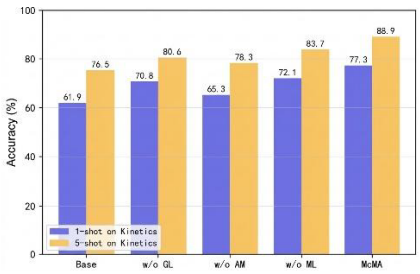


Fig 5. Comparison of key components under the Kinetics dataset

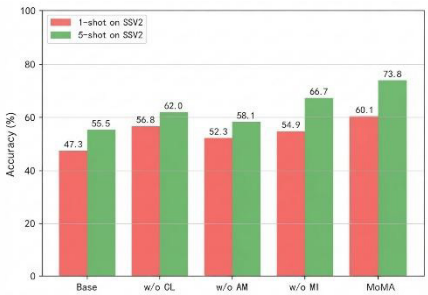


Fig 6. Comparison of key components under the SSV2 dataset

Impact of Different Weight Factors: In this experiment, we evaluate the impact of weight factors α and β on the performance of the proposed model. These weight factors are included in the loss function to scale the values of $\mathcal{L}_1$ to $\mathcal{L}_3$ and $\mathcal{L}_{\text{CONS}}$ to be on the same scale as the cross-entropy loss $\mathcal{L}_{\text{ce}}$. Initially, both α and β were set to 0 as a baseline, and the corresponding accuracy was observed across different datasets. The goal was to find the most suitable values for α and β to optimize performance. As shown in Table 3, the performance of the model gradually improves as the values of α and β increase. Specifically, when $\alpha = 0.1$ and $\beta = 0.1$, the model consistently reaches its best performance across both the 1-shot and 5-shot settings for the Kinetics and SSv2 datasets. For example, on the Kinetics dataset, the accuracy improves from 74.2% to 77.3% in the 1-shot setting, and from 86.8% to 88.9% in the 5-shot setting. On the SSv2 dataset, the accuracy improves from 57.3% to 60.1% in the 1-shot setting, and from 70.4% to 73.8% in the 5-shot setting. These improvements correspond to 3.1%, 2.0%, 2.8%, and 3.4% increases in accuracy for the respective datasets and shot settings. However, when α and β values are increased or decreased beyond 0.1, performance slightly declines, indicating that $\alpha = 0.1$ and $\beta = 0.1$ provide the optimal balance between loss terms.

Table 3. Comparison of various weight factors

α	β	Kinetics		SSV2	
		1-shot	5-shot	1-shot	5-shot
0	0	74.2	86.8	57.3	70.4
0.1	0	74.8	87.3	58.4	71.2
0	0.1	75.4	97.6	58.6	71.5
0	0.2	74.6	86.7	58.8	70.9
0.05	0.1	73.6	87.4	59.7	71.8
0.2	0.1	72.9	87.1	59.5	71.4
0.1	0.1	77.3	88.9	60.1	73.8
0.1	0.05	73.7	87.6	59.8	71.6
0.1	0.2	72.5	87.3	59.6	71.5

Impact of Different Temperature: In this study, we investigate the effect of the hyperparameter temperature τ on the performance of the MoMA model. The temperature τ controls the smoothness of the distribution in the softmax function. We conducted multiple experiments by adjusting τ from 0.01 to 0.11, and the results are summarized in Fig. 7. As observed in the experimental results, as the temperature τ increases, model performance improves. For example, in the 1-shot and 5-shot settings on the Kinetics dataset, the accuracy increases from 75.2% to 77.3% and from 87.1% to 88.9%, respectively, as τ moves from 0.01 to 0.07. Similarly, on the SSv2 dataset, the accuracy improves from 58.9% to 60.1% in the 1-shot setting and from 72.6% to 73.8% in the 5-shot setting as τ increases. These improvements correspond to a 5% increase in performance for both the Kinetics and SSv2 datasets at optimal $\tau = 0.07$. These results indicate that while increasing τ improves model performance up to a certain point, further increases lead to diminishing returns or even performance degradation. Therefore, $\tau = 0.07$ is selected as the optimal temperature hyperparameter.

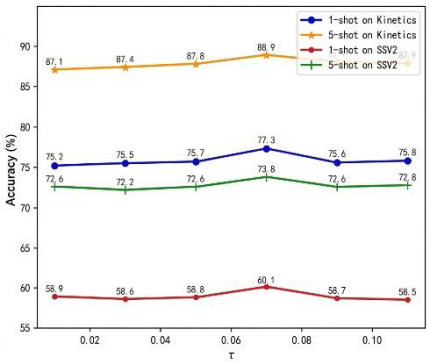


Fig 7. Comparison of various temperature τ

5 Conclusion

In this paper, we proposed MoMA, a Mamba-based multimodal framework for few-shot action recognition. The key contributions can be summarized as follows: first, we leverage an MLLM to provide semantically rich descriptive texts together with a structured semantic embedding, and align them with visual representations to construct a unified multimodal feature sequence; second, we design a tailored multimodal Mamba module by integrating an explicit feed-forward enhancement and normalized residual connections, enabling more effective fusion of multimodal cues while preserving the linear-time modeling advantage of Mamba; third, we introduce a dual-branch contrastive learning objective that jointly considers modality-aware and class-supervised contrasts, which mitigates intra-class heterogeneity accumulation and strengthens inter-class separability under few-shot supervision; finally, extensive experiments on HMDB51, Kinetics, and SSv2 validate the effectiveness of MoMA, showing consistent improvements over state-of-the-art baselines in both 1-shot and 5-shot settings and exhibiting stronger robustness under frame-level noise. In future work, we plan to further improve noise robustness and explore more efficient multimodal prompting/encoding strategies to enhance stability and scalability.

6 Acknowledgement

This paper was supported by the National Natural Science Foundation of China (U23A20280, 62161007, 62471153), the Guangxi Graduate Education Innovation Program (YCSW2025355), and the project of Guangxi Key Laboratory of Precision Navigation Technology and Application (No. DH202312).

References

[1] K. Hamid, C. Cui, K.A. Akbar, Z. Wang, N. Liang, FSDAM: Few-Shot Driving Attention Modeling via Vision-Language Coupling. <https://arxiv.org/abs/2511.12708>, 2025 (accessed 15.05.2025). https://doi.org/10.48550/arXiv.2511.12708

[2] S. Ahn, Y. Jo, K. Lee, S. Kwon, I. Hong, S. Park, AnyAnomaly: Zero-Shot Customizable Video Anomaly Detection with LVLM, in: Proc. IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2026. https://doi.org/10.1109/WACV61042.2026.00296

[3] Y. Qian, K. Geng, K. Chen, S. Cheng, L. Xu, F. Meng, Q. Wu, H. Li, Few-Shot Continual Learning for Activity Recognition in Classroom Surveillance Images, in: Proc. 3rd International Conference on Intelligent Perception and Computer Vision (CIPCV), 2025. https://doi.org/10.1109/CIPCV65863.2025.00019

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is All you Need, in: Proc. Conference on Neural Information Processing Systems (NIPS), 2017. https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[5] A. Gu, T. Dao, Mamba: Linear-Time Sequence Modeling with Selective State Spaces, in: Proc. First Conference on Language Modeling (COLM), 2024. https://openreview.net/forum?id=tEYskw1VY2

[6] T. Dao, A. Gu, Transformers are SSMs: Generalized Models and Efficient Algorithms Through Structured State Space Duality, in: Proc. 41st International Conference on Machine Learning (ICML), 2024. https://proceedings.mlr.press/v235/dao24a.html

[7] A. Gu, I. Johnson, A. Timalsina, A. Rudra, C. Ré, How to Train Your HiPPO: State Space Models with Generalized Orthogonal Basis Projections, in: Proc. International Conference on Learning Representations (ICLR), 2023. https://openreview.net/forum?id=klK17OQ3KB

[8] Y. Chen, Convolutional Neural Network for Sentence Classification, [Master's thesis], University of Waterloo: Waterloo, Ontario, Canada, 2015. https://hdl.handle.net/10012/9592

[9] S.K. Dwivedi, V. Gupta, R. Mitra, S. Ahmed, A. Jain, ProtoGAN: Towards Few Shot Learning for Action Recognition, in: Proc. IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), 2019. https://doi.org/10.1109/ICCVW.2019.00166

[10] T. Perrett, A. Masullo, T. Burghardt, M. Mirmehdi, D. Damen, Temporal-Relational CrossTransformers for Few-Shot Action Recognition, in: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021. https://doi.org/10.1109/CVPR46437.2021.00054

[11] X. Zhu, A.S. Toisoul, J.-M. Perez-Rua, L. Zhang, B. Martinez, T. Xiang, Few-shot Action Recognition with Prototype-centered Attentive Learning, in: Proc. 32nd British Machine Vision Conference (BMVC), 2021. https://doi.org/10.5244/C.35.94

[12] K. Cao, J. Ji, Z. Cao, C.-Y. Chang, J.C. Niebles, Few-Shot Video Classification via Temporal Alignment, in: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020. https://doi.org/10.1109/CVPR42600.2020.01063

[13] S. Zhang, J. Zhou, X. He, Learning Implicit Temporal Alignment for Few-shot Video Classification, in: Proc. Thirtieth International Joint Conference on Artificial Intelligence (IJCAI), 2021. https://doi.org/10.24963/ijcai.2021/181

[14] S. Li, H. Liu, R. Qian, Y. Li, J. See, M. Fei, X. Yu, W. Lin, TA2N: Two-Stage Action Alignment Network for Few-Shot Action Recognition, in: Proc. AAAI Conference on Artificial Intelligence, 2022. https://doi.org/10.1609/aaai.v36i2.20029

[15] A. Thatipelli, S. Narayan, S. Khan, R.M. Anwer, F.S. Khan, B. Ghanem, Spatio-Temporal Relation Modeling for Few-Shot Action Recognition, in: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022. https://doi.org/10.1109/CVPR52688.2022.01933

[16] Y. Fu, L. Zhang, J. Wang, Y. Fu, Y.-G. Jiang, Depth Guided Adaptive Meta-Fusion Network for Few-shot Video Recognition, in: Proc. 28th ACM International Conference on Multimedia, 2020. https://doi.org/10.1145/3394171.3413502

[17] Y. Wanyan, X. Yang, C. Chen, C. Xu, Active Exploration of Multimodal Complementarity for Few-Shot Action Recognition, in: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023. https://doi.org/10.1109/CVPR52729.2023.00628

[18] X. Wang, S. Zhang, Z. Qing, C. Gao, Y. Zhang, D. Zhao, N. Sang, MoLo: Motion-Augmented Long-Short Contrastive Learning for Few-Shot Action Recognition, in: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023. https://doi.org/10.1109/CVPR52729.2023.01727

[19] J. Wu, T. Zhang, Z. Zhang, F. Wu, Y. Zhang, Motion-modulated Temporal Fragment Alignment Network For Few-Shot Action Recognition, in: Proc. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022. https://doi.org/10.1109/CVPR52688.2022.00894

[20] M. Wang, J. Xing, Y. Liu, ActionCLIP: A New Paradigm for Video Action Recognition. <https://arxiv.org/abs/2109.08472>, 2021 (accessed 15.05.2025). https://doi.org/10.48550/arXiv.2109.08472

[21] W. Xiang, C. Li, Y. Zhou, B. Wang, L. Zhang, Generative Action Description Prompts for Skeleton-based Action Recognition, in: Proc. IEEE/CVF International Conference on Computer Vision (ICCV), 2023. https://doi.org/10.1109/ICCV51070.2023.00943

[22] H. Xu, Y. Gao, Z. Hui, J. Li, X. Gao, Language Knowledge-Assisted Representation Learning for Skeleton-Based Action Recognition, IEEE Transactions on Multimedia 27(2025) 5784-5799. https://doi.org/10.1109/TMM.2025.3543034

[23] J. Li, D. Li, C. Xiong, S. Hoi, BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation, in: Proc. 39th International Conference on Machine Learning (ICML), 2022. https://proceedings.mlr.press/v162/li22n.html

[24] J. Li, D. Li, S. Savarese, S. Hoi, BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models, in: Proc. 40th International Conference on Machine Learning (ICML), 2023. https://proceedings.mlr.press/v202/li23q.html

[25] D. Zhu, J. Chen, X. Shen, X. Li, M. Elhoseiny, MiniGPT-4: Enhancing Vision-Language Understanding with Advanced Large Language Models, in: Proc. International Conference on Learning Representations (ICLR), 2024. https://proceedings.iclr.cc/paper_files/paper/2024/hash/50623630a2372839c078474efa6c0cb8-Abstract-Conference.html

[26] P. Gao, J. Han, R. Zhang, Z. Lin, S. Geng, A. Zhou, W. Zhang, P. Lu, C. He, X. Yue, H. Li, Y. Qiao, LLaMA-Adapter V2: Parameter-Efficient Visual Instruction Model. <https://arxiv.org/abs/2304.15010>, 2023 (accessed 16.05.2025). https://doi.org/10.48550/arXiv.2304.15010

[27] L. Zhu, B. Liao, Q. Zhang, X. Wang, W. Liu, X. Wang, Vision Mamba: Efficient Visual Representation Learning with Bidirectional State Space Model, in: Proc. 41st International Conference on Machine Learning (ICML), 2024. https://proceedings.mlr.press/v235/zhu24f.html

[28] Y. Liu, Y. Tian, Y. Zhao, H. Yu, L. Xie, Y. Wang, Q. Ye, J. Jiao, Y. Liu, VMamba: Visual State Space Model, in: Proc. Conference on Neural Information Processing Systems (NeurIPS), 2024. https://doi.org/10.52202/079017-3273

[29] K. He, X. Zhang, S. Ren, J. Sun, Deep Residual Learning for Image Recognition, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016. https://doi.org/10.1109/CVPR.2016.90

[30] X. Pei, T. Huang, C. Xu, EfficientVMamba: Atrous Selective Scan for Light Weight Visual Mamba, in: Proc. AAAI Conference on Artificial Intelligence, 2025. https://doi.org/10.1609/aaai.v39i6.32690

[31] K. He, H. Fan, Y. Wu, S. Xie, R. Girshick, Momentum Contrast for Unsupervised Visual Representation Learning, in: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020. https://doi.org/10.1109/CVPR42600.2020.00975
[32] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A Simple Framework for Contrastive Learning of Visual Representations, in: Proc. 37th International Conference on Machine Learning (ICML), 2020. https://proceedings.mlr.press/v119/chen20j.html
[33] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, D. Krishnan, Supervised Contrastive Learning, in: Proc. Conference on Neural Information Processing Systems (NeurIPS), 2020. https://proceedings.neurips.cc/paper/2020/hash/d89a66c7c80a29b1bdbab0f2a1a94af8-Abstract.html
[34] S. Gidaris, A. Bursuc, N. Komodakis, P. Pérez, M. Cord, Boosting Few-Shot Visual Learning With Self-Supervision, in: Proc. IEEE/CVF International Conference on Computer Vision (ICCV), 2019. https://doi.org/10.1109/ICCV.2019.00815
[35] S. Zheng, S. Chen, Q. Jin, Few-Shot Action Recognition with Hierarchical Matching and Contrastive Learning, in: Proc. European Conference on Computer Vision (ECCV), 2022. https://doi.org/10.1007/978-3-031-19772-7_18
[36] J.-C. Su, S. Maji, B. Hariharan, When Does Self-supervision Improve Few-Shot Learning?, in: Proc. European Conference on Computer Vision (ECCV), 2020. https://doi.org/10.1007/978-3-030-58571-6_38
[37] A. Gu, K. Goel, C. Ré, Efficiently Modeling Long Sequences with Structured State Spaces, in: Proc. International Conference on Learning Representations (ICLR), 2022. https://openreview.net/forum?id=uYLFoz1vlAC
[38] R. Goyal, S.E. Kahou, V. Michalski, J. Materzyńska, S. Westphal, H. Kim, V. Haenel, I. Freund, P. Yianilos, M. Mueller-Freitag, F. Hoppe, C. Thurau, I. Bax, R. Memisevic, The “Something Something” Video Database for Learning and Evaluating Visual Common Sense, in: Proc. IEEE International Conference on Computer Vision (ICCV), 2017. https://doi.org/10.1109/ICCV.2017.622
[39] J. Carreira, A. Zisserman, Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017. https://doi.org/10.1109/CVPR.2017.502
[40] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, T. Serre, HMDB: A large video database for human motion recognition, in: Proc. International Conference on Computer Vision (ICCV), 2011. https://doi.org/10.1109/ICCV.2011.6126543
[41] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, L. Van Gool, Temporal Segment Networks: Towards Good Practices for Deep Action Recognition, in: Proc. European Conference on Computer Vision (ECCV), 2016. https://doi.org/10.1007/978-3-319-46484-8_2
[42] J. Xing, M. Wang, Y. Liu, B. Mu, Revisiting the Spatial and Temporal Modeling for Few-Shot Action Recognition, in: Proc. AAAI Conference on Artificial Intelligence, 2023. https://doi.org/10.1609/aaai.v37i3.25403
[43] Y. Zhang, Y. Fu, X. Ma, L. Qi, J. Chen, Z. Wu, Y.-G. Jiang, On the Importance of Spatial Relations for Few-shot Action Recognition, in: Proc. 31st ACM International Conference on Multimedia, 2023. https://doi.org/10.1145/3581783.3612192
[44] T. Yu, P. Chen, Y. Dang, R. Huan, R. Liang, Multi-Speed Global Contextual Subspace Matching for Few-Shot Action Recognition, in: Proc. 31st ACM International Conference on Multimedia, 2023. https://doi.org/10.1145/3581783.3612380
[45] J. Xing, M. Wang, Y. Ruan, B. Chen, Y. Guo, B. Mu, G. Dai, J. Wang, Y. Liu, Boosting Few-shot Action Recognition with Graph-guided Hybrid Matching, in: Proc. IEEE/CVF International Conference on Computer Vision (ICCV), 2023. https://doi.org/10.1109/ICCV51070.2023.00167
[46] X. Wang, S. Zhang, Z. Qing, M. Tang, Z. Zuo, C. Gao, R. Jin, N. Sang, Hybrid Relation Guided Set Matching for Few-shot Action Recognition, in: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022. https://doi.org/10.1109/CVPR52688.2022.01932