# Motion-modulated Temporal Fragment Alignment Network For Few-Shot Action Recognition

Jiamin Wu[1], Tianzhu Zhang[1,*], Zhe Zhang[2], Feng Wu[1], Yongdong Zhang[1]
[1] University of Science and Technology of China
[2] Lunar Exploration and Space Engineering Center of CNSA
jiaminwu@mail.ustc.edu.cn, {tzzhang, fengwu, zhyd73}@ustc.edu.cn,cnclepzz@126.com

## Abstract

*While the majority of FSL models focus on image classification, the extension to action recognition is rather challenging due to the additional temporal dimension in videos. To address this issue, we propose an end-to-end Motion-modulated Temporal Fragment Alignment Network (MTFAN) by jointly exploring the task-specific motion modulation and the multi-level temporal fragment alignment for Few-Shot Action Recognition (FSAR). The proposed MTFAN model enjoys several merits. First, we design a motion modulator conditioned on the learned task-specific motion embeddings, which can activate the channels related to the task-shared motion patterns for each frame. Second, a segment attention mechanism is proposed to automatically discover the higher-level segments for multi-level temporal fragment alignment, which encompasses the frame-to-frame, segment-to-segment, and segment-to-frame alignments. To the best of our knowledge, this is the first work to exploit task-specific motion modulation for FSAR. Extensive experimental results on four standard benchmarks demonstrate that the proposed model performs favorably against the state-of-the-art FSAR methods.*

## 1. Introduction

Deep learning has achieved tremendous success in the field of action recognition [31,37,38,43]. However, modern deep learning approaches require large amounts of annotated data, and collecting these data is laboriously difficult and costly [1]. To reduce the need for human annotation, Few-Shot Learning (FSL) [9, 32, 41, 44] has been proposed and gained increasing interest, which aims at classifying unlabeled samples (query set) into new unseen classes with only a few labeled examples (support set).

While the majority of FSL models [9, 32, 41, 44] focus on image classification, its extension to video classification is rather challenging. This is because videos have a much more complicated structure than images with an additional temporal dimension [4]. To utilize the temporal information, some recent methods [3, 4, 25] perform temporal alignment to match the video frames or segments in the temporal dimension (see Figure 1), which helps to differ order-sensitive actions. In [4], Cao et al. use Dynamic Time Warping [24] to find the optimal alignment path between frames. Then the video distance is measured as the alignment cost of frame sequences. In [3] and [25], the attention mechanisms are utilized to achieve temporal alignment. In [3], Bishay et al. first uniformly sample segments from videos, with each of them containing the fixed length of frames. Then they feed these segments into 3D CNNs to extract motion features, and conduct segment-level attention for temporal alignment. Similarly, in [25], it randomly samples pairs/triplets of frames from videos to form video segments, and performs attention over these segments.

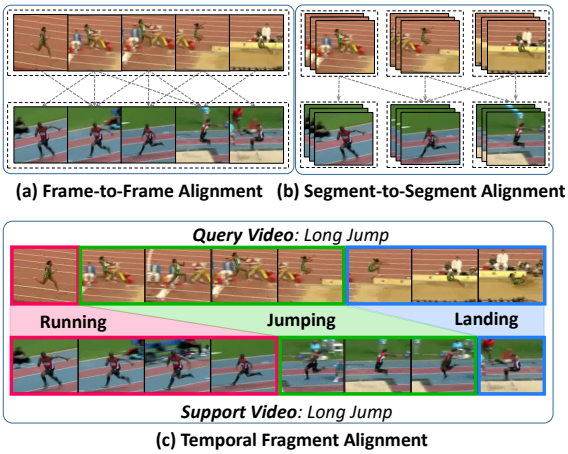


Figure 1. Different ways of alignment. (a), (b): The previous methods [3, 4, 25] match videos at the single level, i.e., "frame to frame" or "segment to segment". (c) The temporal fragment alignment is performed at multiple levels, i.e., "frame to frame", "segment to segment", and "frame to segment", which suits the videos with different speeds in the real-world setting.

By studying the previous Few-Shot Action Recognition (FSAR) methods that are based on temporal alignment [3, 4, 25], we sum up two aspects that are imperative for building a robust FSAR model. **(1) Task-specific Motion Pattern Mining.** Motion modeling has been proved essential for action recognition, as videos contain rich temporal structures [19]. Some methods [3, 16, 48] adopt 3D

*Corresponding Author

CNNs to extract motion features for FSAR. However, they have high computational costs and lack specific consideration in modeling temporal structure for the few-shot setting. Specially in FSAR, the tasks are composed of novel categories, which causes large inter-task variances. Thus, utilizing the same network to extract motion features for all tasks may not be suitable, as the motion patterns have substantial variances in different specific tasks. Therefore, how to effectively mine task-specific motion patterns is of vital importance for FSAR. **(2) Multi-level Temporal Fragment Alignment**. Most previous methods perform either the frame-level alignment [4] or the segment-level alignment [3,25] (see Figure 1 (a) and (b)), which may cause ambiguity and misalignment when matching videos at different speeds. Furthermore, the segments in [3, 25] are obtained by pre-defined sampling strategies, and thus suffer large randomness and may exacerbate the problem of misalignment. In fact, the temporal alignment in the real-world setting not only includes the "frame-to-frame" and "segment-to-segment" alignment, but also includes the "frame-to-segment" alignment, where segments are composed of several semantically-related frames (see Figure 1(c)). Here, we use the **temporal fragments** to uniformly represent both frames and segments. To achieve robust matching for videos at different speeds, the model should have abilities to automatically discover higher-level segments and perform multi-level temporal fragment alignment.

Inspired by the above insights, we propose an end-to-end **Motion-modulated Temporal Fragment Alignment Network (MTFAN)** by jointly exploring the task-specific motion modulation module and the multi-level temporal fragment alignment module for FSAR. In the **task-specific motion modulation module**, we first aggregate the temporal differences over the consecutive frames from the support videos to induce the **task-specific motion pattern**. Subsequently, a motion modulator is proposed to excite motion-relevant channels for frames according to the task-level temporal knowledge. In this way, the networks are forced to discover and enhance the task-shared informative motion information, which facilitates a better alignment between videos in the same task. In the **multi-level temporal fragment alignment module**, we propose a Transformer-inspired Segment Attention Layer to adaptively generate segments by aggregating the arbitrary number of related frames. Specifically, we introduce several learnable segment prototypes conditioned on the prior video context to serve as queries, and take frame features as keys and values. We can obtain higher-level segments by operating attention between the segment prototypes and frames. With the frames and the discovered segments, we can exploit more diverse alignments between the temporal fragments, including the "frame to frame", "segment to segment", and "frame to segment". By considering the **multi-level temporal fragment alignment**, the model could flexibly discover and align similar temporal patterns with different time durations. Finally, we reformulate the temporal fragment alignment process as an Optimal Transport problem [40] and use the Sinkhorn algorithm [8] to solve it.

The contributions of our model could be summarized into three-fold: (1) We propose an end-to-end Motion-modulated Temporal Fragment Alignment Network (MTFAN) by jointly exploiting the task-specific motion modulation and the multi-level temporal fragment alignment. (2) We design a motion modulator to activate the channels related to task-specific motion patterns for the frames. Also, a segment attention layer is proposed to discover the higher-level segments for multi-level temporal fragment alignment. To our best knowledge, this is the first work to exploit task-specific motion modulation for FSAR. (3) Extensive experimental results on four challenging benchmarks demonstrate that our method performs favorably against the state-of-the-art FSAR methods.

## 2. Related Work

In this section, we introduce several lines of research in few-shot image classification, few-shot action recognition, and motion modeling in action recognition.

**Few-Shot Image Classification.** There are two main streams in the few-shot image classification literature. **(1) Optimization-based** methods [2, 9, 17, 22, 27, 34] utilize the meta-learner as the optimizer to adapt model parameters to new tasks. MAML [9] and many of its variants [2, 12, 34] attempt to meta-learn a good model initialization to make sure the model can rapidly adapt to unseen tasks with limited optimization steps. **(2) Metric-based** methods [20,30,32,35,36,41,44,47] learn an embedding space for their chosen distance metrics. Prototypical Network [32] computes Euclidean distances between mean class representations (*i.e.*, prototypes), and performs classification by nearest neighbour searching. Several methods [14, 29, 35, 46] directly learn a deep distance metric by using the CNNs or graph neural networks to infer the affinities. Our method falls into the metric-based type. However, due to the complex temporal structures in videos, directly extending the above methods into FSAR may be suboptimal. Thus, in this paper, we design a motion modulation module and a temporal fragment alignment module to effectively exploit the rich temporal cues for FSAR.

**Few-Shot Action Recognition.** Most of existing FSAR methods [3, 4, 10, 16, 48, 49] adopt the metric learning paradigm, and focus on exploring good metrics to compute the distances between the query and support videos for classification. Some of them [5, 16, 50] directly aggregate the frame features to obtain a single video representation for distance computation. However, these aggregation-based methods ignore the temporal relations that are essential for video classification. To utilize the temporal information,

another line of researches [3, 4, 25] focuses on temporal alignment between videos. OTAM [4] performs the explicit temporal alignment by the DTW algorithm [24], and measures the video distance as the alignment cost. Some methods [3,25] achieve temporal alignment by using an attention mechanism between the segment embeddings of the videos. Their segments are sampled from videos and thus suffer from a certain randomness. In a word, the above temporal alignment methods consider the frame-level alignment or the segment-level alignment solely, but ignore the frame-to-segment alignment that is also common in realistic video matching. Differently, our method automatically explores the higher-level segments for multi-level temporal fragment alignment, which includes the frame-to-frame, segment-to-segment, and frame-to-segment alignments.

**Motion Modeling in Action Recognition.** Motion modeling has been proved essential for action recognition [19]. Recent action recognition methods [7, 26, 37, 38, 45], including several FSAR methods [3, 16, 48], utilize 3D CNNs to model appearance and motion features simultaneously. However, the methods based on 3D CNNs have tremendous parameters to be optimized and thus may not be suitable for the few-shot setting. Another line of works is based on two-stream networks [31, 43] with an optical flow stream to incorporate the motion features. However, the computation of optical flow is also expensive. To avoid high computational cost, some recent methods [13, 19, 42] design temporal difference modules that can be inserted into 2D CNNs for motion extraction. In these methods, the temporal differences can be seen as an efficient substitute for optical flow as motion representation. Still, the above methods are not designed for FSAR, where the test tasks contain novel classes that are unseen in training. Therefore, they may not generalize to the unseen tasks well. Differently, in this paper, we design a task-specific motion modulation module, which can learn the task-related motion patterns and adapt the model to arbitrary tasks.

# 3. Our Method

In this section, we first formulate the task of few-shot action recognition. Then we describe each component of the proposed Motion-modulated Temporal Fragment Alignment Network (MTFAN) in detail. As shown in Figure 2, our MTFAN consists of two modules. (1) The task-specific motion modulation module aims at enhancing the frame features based on the task-specific motion pattern, which involves a channel-wise modulation mechanism. (2) The multi-level temporal alignment module is responsible for automatically discovering higher-level segments that can be combined with frames for temporal fragment alignment.

## 3.1. Problem Definition

The few-shot action recognition is conducted on a set of tasks $\mathcal{T}$ (also called episodes) during testing. The training set $\mathbb{D}_{train}$ is segmented into a set of tasks $\mathcal{T}_{train}$ to mimic the test setting, in the hope of acquiring the generalization ability across tasks. The testing set $\mathbb{D}_{test}$ comprises testing tasks $\mathcal{T}_{test}$ that contain action classes disjoint from the training set $\mathbb{D}_{train}$. Each few-shot task $\mathcal{T}$ consists of a support set $\mathcal{S}$ and a query set $\mathcal{Q}$. Specifically, the $N$-way $K$-shot task means that the task is composed of $N$ classes with $K$ support samples per class. *i.e.*, $\mathcal{S} = \{(V_i^s, y_i^s)\}_{i=1}^{NK}$, where $y_i^s \in \{1, 2, \cdots, N\}$. The query set is composed of $M$ samples per class, *i.e.*, $\mathcal{Q} = \{(V_i^q, y_i^q)\}_{i=1}^{MN}$. The ultimate goal is to classify a query video $V_i^q \in \mathcal{Q}$ into one of the $N$ support classes given a few labeled videos from $\mathcal{S}$.

## 3.2. Task-specific Motion Modulation Module

To extract motion features, some FSAR methods [3, 16, 48] adopt the off-the-shelf 3D CNNs [37]. However, they have high computational costs and do not make modifications for few-shot settings. Here, we propose an efficient motion modeling strategy by introducing the task-specific motion modulation, which can be readily embedded into 2D CNNs.

**Motion Encoder.** To exploit the **task-specific motion patterns**, a motion encoder $\mathbf{E}$ is proposed to transform the temporal differences into the motion features. The temporal differences between adjacent frames are related to the optical flow and can be stacked to approximate the motion features [42]. In specific, we first randomly sample $T$ frames that expand the whole video length for each video $V$ as in [43]. These frames are separately fed into a ResNet-based feature extractor $\mathbf{\Psi_f}$ to obtain features: $\mathbf{I} = \{I_t\}_{t=1}^{T}$, where $I_t \in \mathbb{R}^{H \times W \times C}$ is the feature of the $t$-th frame. Then a motion encoder $\mathbf{E}$ is proposed to extract the task-specific motion embedding $M_{\mathcal{T}}$ by gathering the motion features of all the support videos $\{V_1^s, V_2^s, \cdots, V_{NK}^s\}$ in the current task $\mathcal{T}$:

$$M_{\mathcal{T}} = \mathbf{E}(V_1^s, V_2^s, \cdots, V_{NK}^s). \tag{1}$$

We first explain how to extract the motion feature for each video in $\mathbf{E}$. Concretely, we compute the forward difference feature $D_{f,t}$ and backward difference feature $D_{b,t}$ between $I_t$ and its adjacent frames $I_{t+1}$ and $I_{t-1}$ by

$$\begin{aligned} D_{f,t} &= I_{t+1} - \phi_{\mathbf{smt}}(I_t), \\ D_{b,t} &= I_{t-1} - \phi_{\mathbf{smt}}(I_t), \end{aligned} \tag{2}$$

where $t = 1, \cdots, T$, and $\phi_{\mathbf{smt}}$ is a convolution layer for spatially smoothing, which can alleviate the spatial misalignment. Then, for forward difference features $\{D_{f,t}\}_{t=1}^{T}$, another convolution layer $\phi_{\mathbf{mot}}$ is applied to transform them into the compact motion features. The final forward motion feature $M_f \in \mathbb{R}^C$ is derived by compressing the temporal dimension. Formally,

$$M_f = \frac{1}{T}\sum_{t=1}^{T} GAP(\phi_{\mathbf{mot}}(D_{f,t})), t = 1, 2, \cdots, T, \tag{3}$$

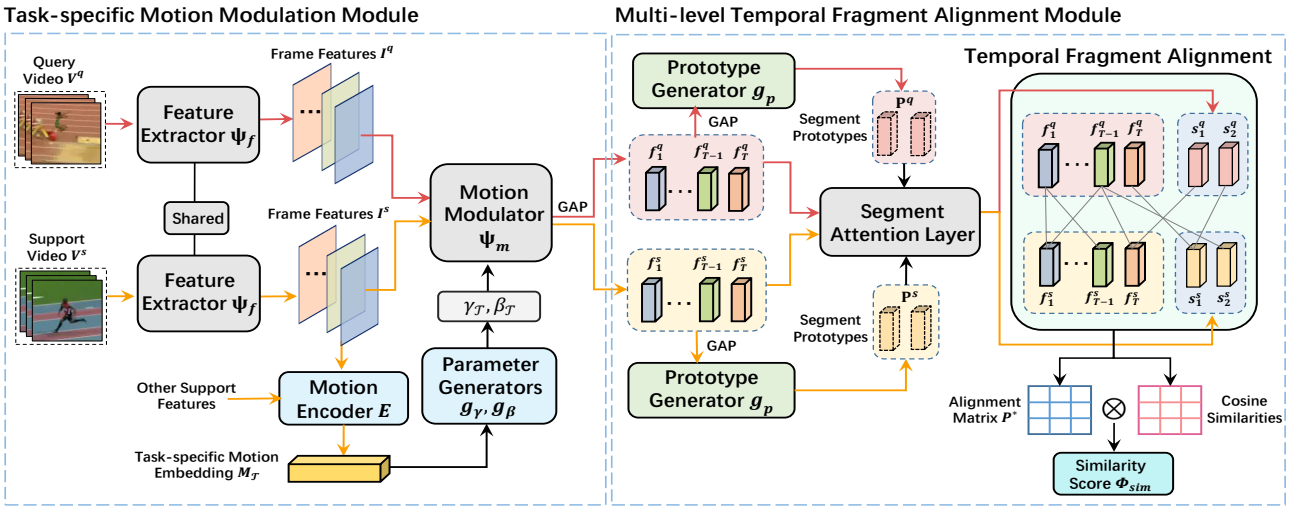


Figure 2. The architecture of our method (illustrated in the 1-shot setting): (1) In the task-specific motion modulation module, given the query and support videos $V^q$ and $V^s$, we first obtain the task-specific motion embedding $M_\mathcal{T}$ by a motion encoder $\mathbf{E}$, then use it to modulate the frame features (*i.e.*, $\mathbf{I}^q$ and $\mathbf{I}^s$) of query and support videos by the motion modulator $\mathbf{\Psi}_m$. (2) In the multi-level temporal fragment alignment module, we generate segment prototypes $\mathbf{P}^q, \mathbf{P}^s$ from the contextual frame embeddings for $V^q$ and $V^s$. The prototypes and the modulated frame embeddings $\{f_t^q\}_{t=1}^T, \{f_t^s\}_{t=1}^T$ are sent into the Segment Attention Layer to discover higher-level segments (*i.e.*, $s_1^q, s_2^q, s_1^s, s_2^s$), which are then combined with the frame embeddings for the multi-level temporal fragment alignment.

where $GAP$ denotes the Global Average Pooling to aggregate the spatial information. The backward motion features $M_b$ can be acquired in a similar way by Equation (3). For every support video $V_i^s$, we can extract such bi-directional motion features $M_f^i, M_b^i$, where $i = 1, 2, \cdots, NK$. Subsequently, the task-specific motion embedding $M_\mathcal{T}$ is obtained by aggregating the support motions:

$$M_\mathcal{T} = \frac{1}{NK} \sum_{i=1}^{NK} \frac{1}{2}(M_f^i + M_b^i). \tag{4}$$

In this way, $M_\mathcal{T}$ is aware of the contextualized global motion knowledge, thus can reveal the useful patterns that could be essential to distinguish different novel categories in the task $\mathcal{T}$.

**Motion Modulator.** To effectively utilize the task-specific motion embedding, we propose a motion modulator $\mathbf{\Psi_m}$ to inject the global motion patterns contained in $M_\mathcal{T}$ into the individual videos. The motion modulator $\mathbf{\Psi_m}$ is composed of several modulation layers, with each layer employing an affine transformation to adapt the frame features in the corresponding layer of the backbone $\mathbf{\Psi_f}$. For brevity of description, we use $\mathbf{\Psi_m}$ to generally demonstrate the modulation process for every layer. Specifically, given the frame features $\mathbf{I} = \{I_t\}_{t=1}^T$ for video $V$, we obtain the adapted frame embedding $f_t$ by the motion modulation:

$$\mathbf{\Psi_m}(I_t) = \gamma_\mathcal{T} I_t + \beta_\mathcal{T}, t = 1, 2, \cdots, T, \tag{5}$$

where $\gamma_\mathcal{T} \in \mathbb{R}^C$ and $\beta_\mathcal{T} \in \mathbb{R}^C$ are the task-shared channel-wise modulation parameters, which are produced by the parameter generators $\mathbf{g}_\gamma$ and $\mathbf{g}_\beta$ conditioned on $M_\mathcal{T}$:

$$\gamma_\mathcal{T} = \mathbf{g}_\gamma(M_\mathcal{T}), \beta_\mathcal{T} = \mathbf{g}_\beta(M_\mathcal{T}), \tag{6}$$

where each parameter generator consists of two linear layers, with the first one followed by a ReLU activation function. Under the guidance of the task-specific motion embedding, $\gamma_\mathcal{T}$ and $\beta_\mathcal{T}$ can strengthen the distinctive channels that are sensitive to the task-shared motion patterns, which helps to find the co-occurrences in the temporal structures for the query and support videos in the same task.

### 3.3. Multi-level Temporal Alignment Module

To accommodate the videos with large speed variations, we design a Segment Attention Layer to discover higher-level segments, which can be combined with the frame sequences to achieve the multi-level temporal alignment.

**Segment Attention Layer.** Inspired by the success of Transformer architecture in discovering local regions [6,18], we extend the cross-attention module in Transformer [39] to the FSAR for segment generation. Concretely, given the modulated features of the final layer of $\mathbf{\Psi_f}$: $\{f_t\}_{t=1}^T, f_t \in \mathbb{R}^C$ for video $V$, we introduce a set of learnable segment prototypes $\mathbf{P} = \{p_j\}_{j=1}^J, p_j \in \mathbb{R}^C$ to serve as queries $\mathbf{Q}$ for gathering the related frames. We design several prototype generators $\mathbf{g_p} = \{g_p^j\}_{j=1}^J$ to produce segment prototypes from the context of frame sequence:

$$p_j = g_p^j(\frac{1}{T}\sum_{t=1}^{T} f_t), j = 1, 2, \cdots, J, \tag{7}$$

where each prototype generator $g_p^j$ consists of a linear layer. Then, we take $\{f_t\}_{t=1}^T$ as keys $\mathbf{K}$ and values $\mathbf{V}$. Following

Transformers, the $\{\mathbf{Q}, \mathbf{K}, \mathbf{V}\}$ triplets are generated by the independent linear projection layers:

$$\mathbf{Q}_j = p_j \mathbf{W}_q, \mathbf{K}_t = f_t \mathbf{W}_k, \mathbf{V}_t = f_t \mathbf{W}_v, \tag{8}$$

where $t = 1, 2, \cdots, T$ and $j = 1, 2, \cdots, J$, and $\mathbf{W}_q \in \mathbb{R}^{C \times d_q}, \mathbf{W}_k \in \mathbb{R}^{C \times d_k}, \mathbf{W}_v \in \mathbb{R}^{C \times d_v}$ are linear projection layers. Then, we can obtain attention scores $\tilde{a}_{jt}$ between keys and queries by:

$$\tilde{a}_{jt} = \frac{\exp(a_{jt})}{\sum_{t'=1}^{T} \exp(a_{jt'})}, a_{jt} = \frac{\mathbf{Q}_j \mathbf{K}_t^T}{\sqrt{d_k}}, \tag{9}$$

where $\sqrt{d_k}$ is a scaling factor. The attention scores $\tilde{a}_{jt}$ can be regarded as the soft correspondences between segment prototypes and frames, which can be used to select and aggregate arbitrary number of semantically-related frames into higher-level segments. Concretely, the segment $\mathbf{s}_j$ is defined as the weighted sum over all values:

$$\mathbf{s}_j = \sum_{t=1}^{T} \tilde{a}_{jt} \mathbf{V}_t, j = 1, 2, \cdots, J. \tag{10}$$

**Temporal Fragment Alignment.** We combine the frame features and segments as the temporal fragment representation $\mathbf{h}$ for the given video $V$: $\mathbf{h} = \{f_1, \cdots, f_T, \mathbf{s}_1, \cdots, \mathbf{s}_J\}$. In order to achieve the multi-level alignment between the temporal fragments, we formulate the video matching task as an Optimal Transport (OT) problem [21, 23]. OT aims at finding the optimal transportation with the minimal cost transportation plan between two discrete distributions $\mu, \nu \in \mathbb{R}^d$. The optimal transportation plan $\mathbf{P}^*$ is obtained by minimizing the transportation cost:

$$\begin{aligned} \mathbf{P}^* &= \underset{\mathbf{P} \in \prod(\mu, \nu)}{\arg\min} \langle \mathbf{P}, \mathbf{C} \rangle, \\ s.t. \quad & \mathbf{P}\mathbf{1} = \mu, \mathbf{P}^T \mathbf{1} = \nu, \end{aligned} \tag{11}$$

where $\prod(\mu, \nu)$ is the joint distribution with marginals $\mu$ and $\nu$, $\langle \cdot, \cdot \rangle$ denotes the cosine similarity, and $\mathbf{C} \in \mathbb{R}^{d \times d}$ represents the cost matrix of transporting $\mu$ to $\nu$. The problem in Equation (11) can be efficiently solved by the Sinkhorn algorithm [8]. To apply OT into the video matching task, we assume the compared query and support videos are the uniform distributions over the temporal fragments. The cost matrix $\mathbf{C}$ is defined by the distances between the temporal fragments, *i.e.* $\mathbf{C} \in \mathbb{R}^{(T+J) \times (T+J)}$. By solving (11), we can obtain $\mathbf{P}^*$ to serve as the alignment matrix, and then define the final similarities between the query video $V^q$ and support video $V^s$ as:

$$\phi_{\mathbf{sim}}(V^q, V^s) = \sum_{m=1, n=1}^{T+J} \langle \mathbf{h}_m^q, \mathbf{h}_n^s \rangle \mathbf{P}_{mn}^*, \tag{12}$$

where $\phi_{\mathbf{sim}}(\cdot, \cdot)$ denotes the similarity function, and $\mathbf{h}^q, \mathbf{h}^s$ denotes the temporal fragments of query and support video $V^q$ and $V^s$. In Equation (12), the alignment scores in $\mathbf{P}^*$ can measure how the similarities between different temporal fragments contribute to the video-level similarities. Notably, for many-shot settings, we can average the temporal fragment representations of the support instances in the same class as the category representation, and then use Equation (12) to compute the similarity between the $V^q$ and the category $c$ as $\phi_{\mathbf{sim}}(V^q, c)$.

Based on the video similarities, the probabilities over class $c \in \{1, 2 \cdots, N\}$ for each query video $V^q$ in the current task can be inferred by a Softmax function: $p(y = c|V^q) = \frac{\exp(\phi_{\mathbf{sim}}(V^q, c))}{\sum_{c'=1}^{N} \exp(\phi_{\mathbf{sim}}(V^q, c'))}$. Then the classification loss $\mathcal{L}_c$ can be defined as the negative log-probability:

$$\mathcal{L}_c = -\frac{1}{|\mathcal{Q}|} \sum_{(V^q, y^q) \in \mathcal{Q}} \log p(y = y^q | V^q). \tag{13}$$

# 4. Experiments

In this section, we first introduce datasets and implementation details. Then, we show experimental results and some visualizations.

## 4.1. Dataset Descriptions

We evaluate our model on four challenging datasets including Something-Something V2 (SSv2) [11], Kinetics [7], UCF101 [33], and HMDB51 [15]. For SSv2 and Kinetics, we follow the same splits as in [4] and [49], which both randomly select 100 classes from the whole dataset with 64, 12, 24 classes used for train/val/test. UCF101 and HMDB51 contain 101 and 51 action classes, respectively. We use the few-shot splits following the practice in [48] for both datasets. The classes in UCF101 are split into 70, 10, 21 classes for train/val/test, respectively. In HMDB51, the 51 classes are split into 31 training classes, 10 validation classes, 10 testing classes.

## 4.2. Implementation Details

For each video, we sparsely and uniformly sample $T = 8$ frames and resize these frames to $256 \times 256$ as in [43]. We utilize TSN [43] to extract the 2D features for video frames. For a fair comparison with previous works [4, 25, 49], we choose ResNet-50 as the backbone for TSN. During training, the video clips are augmented with random horizontal flipping and are then randomly cropped to the size of $224 \times 224$. For testing, we use only the center crop to augment the video. Before the meta-training, we apply a pre-training strategy on the training set for the ResNet backbone to accelerate the training process following the practice in [28, 47]. Then the model is trained in an episodic way, with each episode comprised of the $N$-way $K$-shot task. We mainly experiment with 5-way 1-shot and 5-way 5-shot settings. We use the SGD optimizer with a learning

Table 1. Comparisons of our method with the state-of-the-art methods on **Kinetics**, **SSv2**, **UCF101** and **HMDB51**. The **red font** and **blue font** indicate the highest and the second highest results. The results from our re-implemented version are marked with †.

| Method | Backbone | Kinetics | | SSv2 | | UCF101 | | HMDB51 | |
|---|---|---|---|---|---|---|---|---|---|
| | | 1-shot | 5-shot | 1-shot | 5-shot | 1-shot | 5-shot | 1-shot | 5-shot |
| ProtoGAN [16] | C3D | - | - | - | - | 57.8 | 80.2 | 34.7 | 54.0 |
| TARN [3] | C3D | 66.6 | 80.7 | - | - | - | - | - | - |
| ARN [48] | C3D | 63.7 | 82.4 | - | - | 62.1 | 84.8 | 44.6 | 59.1 |
| Matching Net [49] | ResNet-50 | 53.3 | 74.6 | - | - | - | - | - | - |
| MAML [49] | ResNet-50 | 54.2 | 75.3 | - | - | - | - | - | - |
| CMN [49] | ResNet-50 | 60.5 | 78.9 | - | - | - | - | - | - |
| TARN [3] | ResNet-50 | 64.8 | 78.5 | - | - | - | - | - | - |
| OTAM [4] | ResNet-50 | 73.0 | 85.8 | 42.8 | 52.3 | - | - | - | - |
| TRX [25] | ResNet-50 | 63.6 | 85.9 | 42.0 | 64.6 | 78.8† | 96.1 | 52.2† | 75.6 |
| **MTFAN (ours)** | ResNet-50 | 74.6 | 87.4 | 45.7 | 60.4 | 84.8 | 95.1 | 59.0 | 74.6 |

Table 2. Ablation results on SSv2 and UCF101 in 5-way 1-shot and 5-way 5-shot settings.

| Method | SSv2 | | UCF101 | |
|---|---|---|---|---|
| | 1-shot | 5-shot | 1-shot | 5-shot |
| Baseline | 37.4 | 49.5 | 78.6 | 92.4 |
| Baseline+OT | 39.2 | 52.7 | 81.6 | 93.0 |
| Baseline+OT+segment | 41.6 | 53.9 | 82.3 | 93.1 |
| Baseline+OT+TMM | 42.4 | 55.5 | 83.3 | 94.4 |
| Baseline+OT+TFA | 43.3 | 57.5 | 83.8 | 94.2 |
| **MTFAN** | **45.7** | **60.4** | **84.8** | **95.1** |

rate of 0.0001. The number of the learned segments is set as 4 for SSv2 and 2 for other datasets, which are selected by the episodic cross-validation. During the testing stage, we report the average classification accuracy in 1000 randomly sampled tasks. The training requires one Tesla V100 GPU.

### 4.3. Comparisons with Other Methods

We compare our MTFAN with various state-of-the-art methods on different datasets and few-shot settings. As shown in Table 1, our MTFAN sets the new state-of-the-art results on all datasets in the 5-way 1-shot setting, which strongly proves the effectiveness of our method. In the 5-way 5-shot setting, our method also performs comparably with the state-of-the-art methods. Based on the results, we have the following observations. **(1) Compared with the best C3D-based methods** (*i.e.*, ARN [48]), our method achieves a large improvement of **22.7%** and **10.3%** in 1-shot and 5-shot settings on UCF101. The 3D CNNs introduce a large number of optimization parameters, which may cause the over-fitting problem, especially in the few-shot learning. Also, directly using the general motion extractor may not accommodate the needs of different few-shot tasks. Differently, we utilize the temporal differences to extract task-specific motion patterns to augment the representation of each video in the task. **(2) Compared with the ResNet-based methods**, our method outperforms the state-of-the-art performance by a significant margin of **6.0%** and **6.8%** on UCF101 and HMDB51 in the 1-shot setting, which proves that our network enables effective adaptation to the novel task when the data is extremely scarce. The task-specific mechanism in the motion modulation helps our model to extract the helpful motion representations for each specific task, which further boosts the generalization ability. **(3) Compared with the methods based on temporal alignment** (*i.e.*, TARN [3], OTAM [4], and TRX [25]), our method acquires better performances in the majority of the results, which proves the superiority of our proposed multi-level alignment. We conduct more diverse and flexible alignments between the frames and the automatically-learned segments, which is more robust to the videos with different speeds. Notably, MTFAN performs slightly lower than TRX [25] in 5-shot setting. This is because we average the support features for each class before the comparison. While in TRX, all support features are compared to the query video by an attention mechanism, which causes higher computation cost and does not work for the 1-shot setting (with accuracy lower than MTFAN by 11% on Kinetics). In the future, we will try to improve 5-shot results by considering the importance of different support videos.

### 4.4. Ablation Study

In this section, we perform detailed ablation studies to demonstrate the effectiveness of our selections of the proposed method.

**Baseline.** We begin with the introduction of our baseline method, which is extended from the classical few-shot learning method ProtoNet [32]. The same pre-trained backbone as MTFAN is utilized. Specifically, we average the frame features in the temporal dimension to get a video-level representation. We take the mean representations of the support samples in the same class as the class prototypes. Then we calculate the distances between the given query and prototypes, and classify the query as the category of the nearest class prototype.

**Analysis of Model Components.** We perform a detailed analysis of model components of MTFAN on SSv2 and UCF101 (see Table 2). We denote the Task-specific Motion Modulation module as **TMM**, and the multi-level Temporal Fragment Alignment module as **TFA**. We also evaluate

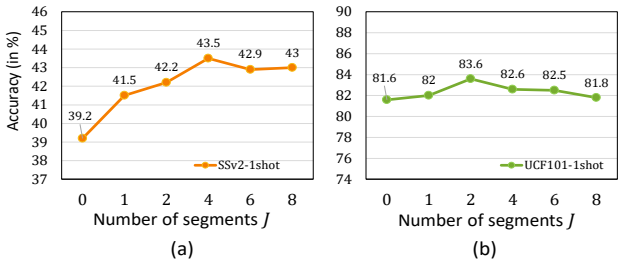


Figure 3. The effect of the number of discovered segments on (a) SSv2 and (b) UCF101 in the 5-way 1-shot setting.

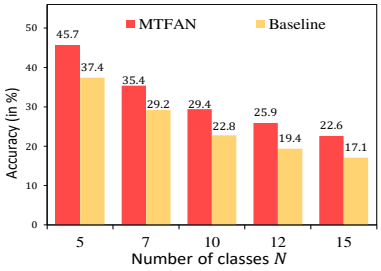


Figure 4. Comparisons of MTFAN with the baseline under different $N$ way 1 shot tasks on SSv2 in the 5-way 1-shot setting, where $N$ denotes the number of classes and varies in $[5, 7, 10, 12, 15]$.

the performance of performing the frame-to-frame alignment solely (denoted as "Baseline + OT") or the segment-to-segment alignment solely (denoted as "Baseline + OT + segment") by use of the Optimal Transport algorithm. The results are analyzed as follows: (1) Compared to the baseline, the utilization of OT for the frame-level alignment brings obvious improvements (*e.g.*, 3.2% in 5-shot setting on SSv2). Performing the segment-level alignment solely also substantially boosts the performance, which proves the efficacy of automatic segment generation by adaptively collecting several semantically-related frames. (2) With the utilization of TMM, a clear performance boost can be observed. The TMM can enhance the motion patterns for every sample by use of channel modulation based on the task-specific motion embeddings. In this way, the global task-shared information can further benefit the subsequent alignment within the task. (3) The introduction of TFA achieves remarkable accuracy gains compared with solely using the frame-level alignment or the segment-level alignment. The improvements can be mainly ascribed to the multi-level temporal fragment alignment that discovers diverse matching relations between any pair of frames and segments.

**Analysis of the Number of Segments.** We study the impact of the number of discovered segments (denoted as $J$) on SSv2 and UCF101 (see Figure 3). On SSv2, even using one segment brings a certain improvement compared to the baseline. With the growth of $J$, the accuracy presents an obvious ascending trend, as more segments can expand the diversity of the temporal fragment alignment and contribute to the more accurate video matching. However, the accuracy reaches the peak value at $J = 4$, and then falls when introducing more segments. The reason may be that increasing $J$ brings more parameters and may aggravate the risk of over-fitting. On UCF101, we can observe the same accuracy variation trend, but the best performance is achieved when using 2 segments. Overall speaking, the performance changes relatively smoothly under different $J$, which indicates the robustness of the proposed segment attention layer.

Table 3. Comparisons of our generated segments with the sampled segments on SSv2 and UCF101 in the 5-way 1-shot setting.

| Method | SSv2 | UCF101 |
|---|---|---|
| ***Sampled Segments*** | | |
| Triplets | 37.3 | 81.8 |
| Pairs & Triplets | 37.8 | 82.0 |
| 2 Segments in 8 Frames | 40.3 | 81.5 |
| 4 Segments in 8 Frames | 39.3 | 81.6 |
| 4 Segments in 12 Frames | 41.1 | 82.2 |
| **Our Generated Segments** | **42.6** | **82.6** |

Table 4. Comparisons of different modulation strategies on SSv2 in 5-way 1-shot and 5-way 5-shot settings.

| Method | 1-shot | 5-shot |
|---|---|---|
| Modulating with appearances | 39.3 | 52.6 |
| **Modulating with motions** | **42.4** | **55.5** |

**Different number of classes $N$.** We test the model performance in more challenging few-shot scenarios by increasing the number of classes in the task (denoted as $N$). From Figure 4 we can see that the accuracies of MTFAN and the baseline decrease when increasing $N$. This is not surprising as more involved novel classes make the few-shot classification more difficult. Notably, even in the extremely challenging few-shot setting (*e.g.*, $N = 15$), MTFAN still surpasses the baseline by a significant margin of 5.5%, which demonstrates the generalization ability of our method in handling the scarce data.

**Comparisons of the generated and sampled segments.** To quantitatively analyse the effect of the segment generation, we make comparisons with other segment sampling strategies adopted in [3, 25]. Following [25], we sample eight frames for each video and then exhaustively sample pairs and triplets of frames as segments. We also uniformly sample segments from videos with each segment consisting of the fixed length of frames, as in [3]. We experiment with 4 or 2 segments using 8 frames or 12 frames in total, and average the frame features to form the segment representations. We use the OT to conduct segment-to-segment alignment for all the methods for a fair comparison. As we can see from Table 3, our generated segments achieve the best results. Even compared with the sampled segments that use much more frames (see the results in the fifth row), our method can still lead by 1.5% on SSv2. The results strongly prove the advantage of the automatical segment generation, where the related frames are selected and aggregated to form more reasonable segments.

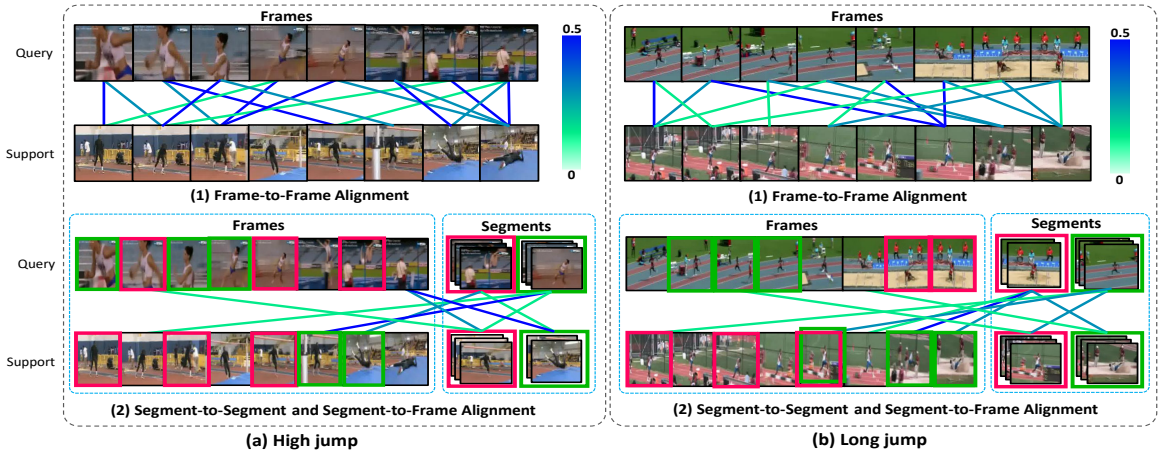


Figure 5. **Visualization of the learned multi-level temporal fragment alignment** between the query and support videos in category (a) high jump and (b) long jump on UCF101. We sample eight frames and learn two segments for each video. The colors of connecting edges indicate the values of alignment scores in the alignment matrix $\mathbf{P}^*$. (a) (1) and (b) (1) present the **Frame-to-Frame Alignment**, while (a) (2) and (b) (2) illustrate **the learned segments** and the **Segment-to-Segment, Segment-to-Frame Alignment**. The frames with high attention scores in the segment attention layer and their corresponding 2 segments are illustrated with boxes in the same colors (red and green). We can see that the frames and the segments that contain similar sub-actions have high alignment scores.

**Comparisons of different modulation strategies.** To evaluate the impact of motion modeling, we replace the task-specific motion embedding $M_{\mathcal{T}}$ with the 2D appearance features in the modulation. Specifically, we average the frame features of the support samples, and use them to generate modulation parameters. The comparison results are shown in Table 4. The replacement of the motion features leads to a clear decline in the accuracies in both 1-shot and 5-shot settings, which reveals the necessity and usefulness of the motion modeling for FSAR. The appearance representations alone may not express the key information in temporal structures.

### 4.5. Visualizations

In this section, we present several visualizations to vividly illustrate the procedure and effectiveness of the multi-level temporal fragment alignment.

**Visualization of Frame-to-Frame Alignment.** In Figure 5 (a) (1) and (b) (1), we display the learned alignment matrix $\mathbf{P}^*$ in the Frame-to-Frame Alignment between the query and the support videos of two categories ("high jump" and "long jump"). As can be observed, a certain number of frames can be aligned well. However, due to the randomness of the sampling strategy and the noise of the background regions, some of the isolated frames are semantically ambiguous when the context frames are not considered (*e.g.*, the seventh support frame in Figure 5 (b) (1)), which causes misalignments between frames.

**Visualization of Discovered Segments.** We visualize the segments that are automatically generated by the segment attention layer. For each video, we learn two segments from the sampled eight frames. As presented in Figure 5 (a) (2) and (b) (2), the frames with high attention scores and their corresponding two segments are illustrated with the boxes of the same colors (red and green). We can observe that the frames that make up the same segments generally have semantically similar action patterns, such as "running", "jumping" and "falling", which proves the effectiveness of the segment attention layer in matching the related frames with the segment prototypes.

**Visualization of Segment-to-Segment and Segment-to-Frame Alignment.** As shown in Figure 5 (a) (2) and (b) (2), the high alignment scores can be observed between the segments that have large similarities. The segments also have strong connections with the semantically related frames. These phenomena justify the reliability of the multi-level temporal fragment alignment, which considers more comprehensive ways of alignments and enables the flexible exploit of the higher-order temporal relations. Therefore, the learned temporal alignment can contribute to a more accurate similarity measure for videos.

## 5. Conclusion

In this paper, we propose a Motion-modulated Temporal Fragment Alignment Network for FSAR. We design a motion modulator to enhance the frame features based on the learned task-specific motion embedding. Also, a segment attention mechanism is proposed to automatically discover higher-level segments for the multi-level temporal fragment alignment. Experiments show the effectiveness.

## 6. Acknowledgement

This work was partially supported by the National Nature Science Foundation of China (Grant 62022078, 62121002, 62071122), and National Defense Basic Scientific Research Program (JCKY2020903B002).

## References


[1] Maria-Luiza Antonie, Osmar R Zaiane, and Alexandru Coman. Application of data mining techniques for medical image classification. In *Proceedings of the Second International Conference on Multimedia Data Mining*, pages 94–101, 2001. 1

[2] Antreas Antoniou, Harrison Edwards, and Amos Storkey. How to train your maml. In *International Conference on Learning Representations*, 2018. 2

[3] Mina Bishay, Georgios Zoumpourlis, and Ioannis Patras. Tarn: Temporal attentive relation network for few-shot and zero-shot action recognition. *arXiv preprint arXiv:1907.09021*, 2019. 1, 2, 3, 6, 7

[4] Kaidi Cao, Jingwei Ji, Zhangjie Cao, Chien-Yi Chang, and Juan Carlos Niebles. Few-shot video classification via temporal alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10618–10627, 2020. 1, 2, 3, 5, 6

[5] Chris Careaga, Brian Hutchinson, Nathan Oken Hodas, and Lawrence Phillips. Metric-based few-shot learning for video action recognition. *arXiv preprint arXiv:1909.09602*, 2019. 2

[6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020. 4

[7] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 3, 5

[8] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in Neural Information Processing Systems*, 26:2292–2300, 2013. 2, 5

[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135, 2017. 1, 2

[10] Yuqian Fu, Li Zhang, Junke Wang, Yanwei Fu, and Yugang Jiang. Depth guided adaptive meta-fusion network for few-shot video recognition. In *ACM International Conference on Multimedia*, 2020. 2

[11] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The "something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5842–5850, 2017. 5

[12] Muhammad Abdullah Jamal and Guo-Jun Qi. Task agnostic meta-learning for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2

[13] Boyuan Jiang, MengMeng Wang, Weihao Gan, Wei Wu, and Junjie Yan. Stm: Spatiotemporal and motion encoding for action recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, October 2019. 3

[14] Jongmin Kim, Taesup Kim, Sungwoong Kim, and Chang D Yoo. Edge-labeling graph neural network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11–20, 2019. 2

[15] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2556–2563, 2011. 5

[16] Sai Kumar Dwivedi, Vikram Gupta, Rahul Mitra, Shuaib Ahmed, and Arjun Jain. Protogan: Towards few shot learning for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 1, 2, 3, 6

[17] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10657–10665, 2019. 2

[18] Yulin Li, Jianfeng He, Tianzhu Zhang, Xiang Liu, Yongdong Zhang, and Feng Wu. Diverse part discovery: Occluded person re-identification with part-aware transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2898–2907, 2021. 4

[19] Yan Li, Bin Ji, Xintian Shi, Jianguo Zhang, Bin Kang, and Limin Wang. Tea: Temporal excitation and aggregation for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2020. 1, 3

[20] Bin Liu, Yue Cao, Yutong Lin, Qi Li, Zheng Zhang, Mingsheng Long, and Han Hu. Negative margin matters: Understanding margin in few-shot classification. *arXiv preprint arXiv:2003.12060*, 2020. 2

[21] Weizhe Liu, Bugra Tekin, Huseyin Coskun, Vibhav Vineet, Pascal Fua, and Marc Pollefeys. Learning to align sequential actions in the wild. *arXiv preprint arXiv:2111.09301*, 2021. 5

[22] Yaoyao Liu, Bernt Schiele, and Qianru Sun. An ensemble of epoch-wise empirical bayes for few-shot learning. In *European Conference on Computer Vision*, pages 404–421, 2020. 2

[23] Su Lu, Han-Jia Ye, and De-Chuan Zhan. Few-shot action recognition with compromised metric via optimal transport. *arXiv preprint arXiv:2104.03737*, 2021. 5

[24] Meinard Müller. Dynamic time warping. *Information Retrieval for Music and Motion*, pages 69–84, 2007. 1, 3

[25] Toby Perrett, Alessandro Masullo, Tilo Burghardt, Majid Mirmehdi, and Dima Damen. Temporal-relational crosstransformers for few-shot action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 475–484, 2021. 1, 2, 3, 5, 6, 7

[26] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5533–5541, 2017. 3

[27] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *International Conference on Learning Representations*, 2017. 2

[28] Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. In *International Conference on Learning Representations*, 2019. 5

[29] Victor Garcia Satorras and Joan Bruna Estrach. Few-shot learning with graph neural networks. In *International Conference on Learning Representations*, 2018. 2

[30] Christian Simon, Piotr Koniusz, Richard Nock, and Mehrtash Harandi. Adaptive subspaces for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4136–4145, 2020. 2

[31] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*, pages 568–576, 2014. 1, 3

[32] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pages 4077–4087, 2017. 1, 2, 6

[33] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 5

[34] Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele. Meta-transfer learning for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 403–412, 2019. 2

[35] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2

[36] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: a good embedding is all you need? *arXiv preprint arXiv:2003.11539*, 2020. 2

[37] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4489–4497, 2015. 1, 3

[38] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018. 1, 3

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017. 4

[40] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer, 2009. 2

[41] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, pages 3630–3638, 2016. 1, 2

[42] Limin Wang, Zhan Tong, Bin Ji, and Gangshan Wu. Tdn: Temporal difference networks for efficient action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1895–1904, June 2021. 3

[43] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European Conference on Computer Vision*, pages 20–36. Springer, 2016. 1, 3, 5

[44] Jiamin Wu, Tianzhu Zhang, Yongdong Zhang, and Feng Wu. Task-aware part mining network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8433–8442, 2021. 1, 2

[45] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *European Conference on Computer Vision*, 2018. 3

[46] Ling Yang, Liangliang Li, Zilun Zhang, Xinyu Zhou, Erjin Zhou, and Yu Liu. Dpgn: Distribution propagation graph network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 13390–13399, 2020. 2

[47] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. Few-shot learning via embedding adaptation with set-to-set functions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8808–8817, 2020. 2, 5

[48] Hongguang Zhang, Li Zhang, Xiaojuan Qi, Hongdong Li, Philip HS Torr, and Piotr Koniusz. Few-shot action recognition with permutation-invariant attention. In *European Conference on Computer Vision*, pages 525–542. Springer, 2020. 1, 2, 3, 5, 6

[49] Linchao Zhu and Yi Yang. Compound memory networks for few-shot video classification. In *European Conference on Computer Vision*, pages 751–766, 2018. 2, 5, 6

[50] Xiatian Zhu, Antoine Toisoul, Juan-Manuel Pérez-Rúa, Li Zhang, Brais Martínez, and Tao Xiang. Few-shot action recognition with prototype-centered attentive learning. *arXiv preprint arXiv:2101.08085*, 2021. 2