Decoupling and Recoupling Spatiotemporal Representation for RGB-D-based Motion Recognition

Benjia Zhou\textsuperscript{1,2*}, Pichao Wang\textsuperscript{2}, Jun Wan\textsuperscript{1,3,4†}, Yanyan Liang\textsuperscript{1}, Fan Wang\textsuperscript{2}, Du Zhang\textsuperscript{1}, Zhen Lei\textsuperscript{3,4}, Hao Li\textsuperscript{2}, Rong Jin\textsuperscript{2}

\textsuperscript{1}Macau University of Science and Technology \textsuperscript{2}Alibaba Group \textsuperscript{3}NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing, China \textsuperscript{4}School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China

Abstract

Decoupling spatiotemporal representation refers to decomposing the spatial and temporal features into dimension-independent factors. Although previous RGB-D-based motion recognition methods have achieved promising performance through the tightly coupled multi-modal spatiotemporal representation, they still suffer from (i) optimization difficulty under small data setting due to the tightly spatiotemporal-entangled modeling; (ii) information redundancy as it usually contains lots of marginal information that is weakly relevant to classification; and (iii) low interaction between multi-modal spatiotemporal information caused by insufficient late fusion. To alleviate these drawbacks, we propose to decouple and recouple spatiotemporal representation for RGB-D-based motion recognition. Specifically, we disentangle the task of learning spatiotemporal representation into 3 sub-tasks: (1) Learning high-quality and dimension independent features through a decoupled spatial and temporal modeling network. (2) Recoupling the decoupled representation to establish stronger space-time dependency. (3) Introducing a Cross-modal Adaptive Posterior Fusion (CAPF) mechanism to capture cross-modal spatiotemporal information from RGB-D data. Seamless combination of these novel designs forms a robust spatiotemporal representation and achieves better performance than state-of-the-art methods on four public motion datasets. Our code is available at https://github.com/damo-cv/MotionRGBD.

1. Introduction

The RGB-D-based motion recognition has attracted much attention in computer vision due to its broad application scenarios such as video surveillance and human-object interfaces. Recently, the CNN and RNN based methods greatly improve the performance of recognition on both gesture [1, 27, 43, 45] and action [7, 17, 38, 41] through fully exploring the color and depth cues. Meanwhile, inspired by the transformer scaling success in vision tasks, Transformer-based methods [10, 21] also achieve surprising results on RGB-D-based motion recognition by introducing the cross-attention module for multi-modality fusion.

Although these works make great progress, we find they are still problematic in the following three aspects. (i) Optimization difficulty exists in the case of limited RGB-D data due to the tightly spatiotemporal entangled modelling (e.g., C3D [33] and I3D [3]). (ii) Redundant information is hard to deal with in the entangled space-time space. To address the above two issues, some decoupled networks (i.e., 2D CNN+LSTM/Transformer [19, 34]) are proposed to learn the spatiotemporal independent representation. However, we argue that these methods are not conducive to compact representation as they somewhat weaken or even de-
stroys the original spatiotemporal coupling structure. Considering that a certain number of human action classes have strong correlations between time and space, the recoupling process after spatiotemporal decoupling is still necessary. (iii) **Insufficient interaction** occurs between multi-modal spatiotemporal information. Several works \[45, 47\] adopt independent branches for unimodal spatiotemporal representations learning followed by late fusion, resulting in insufficient cross-modal information communication. Thus, it is still a challenge to explore high quality multi-modal spatiotemporal features.

Given the aforementioned concerns, as illustrated in Figure 1, we introduce a new method of multi-modal spatiotemporal representation learning for RGB-D-based motion recognition. It mainly consists of a decoupled spatial representation learning network (DSN), a decoupled temporal representation learning network (DTN) and a cross-modal adaptive posterior fusion module (CAPF). For each unimodal branch, as shown in Figure 2, we propose a decoupling and recoupling spatiotemporal feature learning method, wherein a spatiotemporal recoupling module (RCM) is designed as a bonding of DSN and DTN. RCM acts as feature selection for DSN and knowledge integration for DTN. The entire framework can be decomposed into 3 steps: (1) **Spatiotemporal decoupling learning.** In the DSN, the video clips are first fed into stack of inception-based spatial multi-scale features learning (SMS) modules to extract hierarchical spatial features. Meanwhile, they are also input into a bypath network, called fast regional positioning module (FRP), to generate visual guidance map, which guides the network to focus on local important areas in the video frame. Then the integrated spatial features from SMS and FRP are fed into RCM for feature selection. After that, we sample several sub-sequences at different frame rates from the enhanced spatial features as input to the DTN. The DTN is configured as a multi-branch structure with an inception-based temporal multi-scale layer (TMS) and multiple Transformer blocks for hierarchical local fine-grained and global coarse-grained temporal feature learning. (2) **Spatiotemporal recoupling learning.** To rebuild the space-time interdependence, a self-distillation-based recoupling strategy is developed. As shown by the red dashed line in Figure 2, the recoupling method is designed as an inner loop optimization mechanism to distill the inter-frame correlations from time domain into the space domain, to enhance the quality of the spatial features via RCM. (3) **Cross-modal interactive learning.** For multi-modal representation learning from RGB-D data, as shown in Figure 1, we propose an interactive cross-modal spatiotemporal representation learning method. Specifically, the cross-modal spatial features derived from unimodal branches firstly interact at the spatial level and are mapped to a joint spatial representation. Then it is separately integrated with the two unimodal spatial feature streams through the residual structure. After that, the two spatial feature streams are input into their respective temporal modeling networks to capture temporal features. Similar to the spatial feature interaction, a joint temporal representation can also be obtained through interaction at the temporal level. Combined with the joint temporal representation, the two temporal feature streams are fed into the CAPF, which is based on a multi-loss joint optimization mechanism, to conduct deep multi-modal representation fusion.

Through the above design, our method not only effectively achieves the spatiotemporal information decoupling and recoupling learning within each modality, but also realizes the deep communication and fusion of multi-modal spatiotemporal information. The proposed method achieves state-of-the-art performance on four public RGB+D gesture/action datasets, namely NvGesture \[26\], Chalearn
2. Related Work

2.1. Motion Recognition based on RGB-D Data

Recently, with the availability of low-cost RGB-D sensors, RGB-D-based motion recognition has attracted extensive attention. To effectively encode the robust multi-modal spatiotemporal information for motion recognition, Zhu et al. [46] presents a pyramidal-like 3D convolutional network structure for spatiotemporal representation extraction and fusion. Kong et al. [20] propose to compress and project the RGB-D data into a shared space to learn cross-modal features for effective action recognition. Yu et al. [43] employ NAS to search for modal-related network structures and operators for spatiotemporal representation extraction and aggregation. Unlike previous methods that interact with the cross-modal information on coupled spatiotemporal information, we focus on the interaction of multi-modal features in two independent dimensions of space and time.

2.2. Decoupled Spatiotemporal Feature Learning

Considering the importance of decoupled spatiotemporal feature learning in sequence, Shi et al. [31] present a decoupled spatiotemporal attention network (DSTA-Net) for skeleton-based action recognition. Liu et al. [22] present a decoupled spatiotemporal Transformer (DSTT) architecture to improve video inpainting tasks. He et al. [14] propose an effective spatiotemporal network StNet, which employs separated channel-wise and temporal-wise convolution operations for decoupled local and global representation learning. Zhang et al. [44] present a hierarchically decoupled spatiotemporal contrastive learning method for self-supervised video representation learning. They capture the spatial and temporal features by decoupling the learning objective into two contrastive sub-tasks, and perform it hierarchically to encourage multi-scale understanding. In contrast, in this work, we target to learn recoupled features due to strong correlations between time and space for some human actions. Thus, a distillation-based recoupling process is introduced on the decoupled spatiotemporal features.

3. Proposed Method

In this paper, we assume the spatiotemporal representation can be decomposed into two sub-domains: the spatial domain that correlates with the visual information, and the temporal domain that describes the time-related concept. Based on this, for unimodal spatiotemporal representation learning, we first decouple the spatiotemporal modeling process to learn domain-independent representations (Sec. 3.1). Then a recoupling method is introduced based on the inner loop optimization mechanism during the training stage (Sec. 3.2), to strengthen the spatiotemporal connection. For multi-modal features interactive learning, we first separately integrate the cross-modal spatiotemporal information into the spatial and temporal domains, and then employ an adaptive posterior fusion mechanism to further fuse the multi-modal features (Sec. 3.3).

3.1. Decoupling Spatiotemporal Representation

3.1.1 Decoupled Spatial Feature Learning (DSN)

As shown in Figure 2, the DSN is composed of fast region positioning module (FRP) and stack of inception-based spatial multi-scale features learning (SMS) modules. Let \([I_1, I_2, \ldots, I_T]\) denote the input with length \(T\) sampled from the video. It is fed into the SMS and FRP modules in parallel to capture the hierarchical spatial features and generate visual guidance maps.

**SMS Module.** The SMS module is composed of a space-centric 3D Inception Module\(^1\) [3] and a Max Pooling layer. It extracts the multi-scale features of \(m\)-th frame at layer \(l\)-th by:

\[
 f^l_m = \begin{cases} 
 \text{Maxpool}(C_{S-Inc}(I_m, W)), & l = 1 \\
 \text{Maxpool}(C_{S-Inc}(f^{l-1}(I_m, W))), & l > 1 
\end{cases} 
\]

where \(f^{l-1}\) represents the output of the previous layer; \(C_{S-Inc}(\cdot, W)\) indicates the Inception Module with learnable parameter matrix \(W\); Maxpool is the Max Pooling layer. Meanwhile, to guide each SMS module to focus on local important areas in the image, a visual guidance map is embedded in parallel to it to further enhance its visual perception, which is the proposed FRP module.

---

\(^1\)The size of the convolution kernel in the temporal dimension is 1.
in an image for motion recognition. The visual guidance map is obtained directly by combining static and dynamic guidance maps. Specifically, as shown in Figure 3, within a sliding window from the \( n \)-th to the \( m \)-th frame, we first calculate the dynamic image \( DI(n, m) \) [45]:
\[
DI(n, m) = DI(n - 1, m - 1) + (m - n) \times (I_{n-1} + I_m) - 2 \sum_{i=n}^{m-1} I_i, \quad s.t. \quad m > n
\]
where \( DI(n - 1, m - 1) \) represents the dynamic image from the previous sliding window. Then the dynamic guidance map on \( m \)-th frame can be obtained by:
\[
D_m = \delta(DI(n, m)) \times \lambda
\]
where \( \delta \) and \( \lambda \) indicate activation function and signal amplification factor respectively, herein we use GELU [15] and \( \lambda = 2 \) in all of experiments. However, we find that \( D_m \) is sensitive to lighting as it only considers the motion information between multiple frames. To mitigate this problem, we introduce the static guidance map \( S_m \), which can be defined as:
\[
S_m = \text{dilate}(\text{erode}(D_m), (k \times k))
\]
\[
\hat{D}_m = \begin{cases} 
D_{m,(i,i)} & D_{m,(i,i)} \geq D_{m,\text{mean}} \\
0 & \text{otherwise}
\end{cases}
\]
where \( \text{dilate}(\text{erode}(\cdot, (k \times k))) \) indicates the dilation and erosion operations with a kernel size of \( k \times k \) in binary mathematical morphology. \( \hat{D}_m \) is an attention matrix, in which all elements except for these higher than the mean value are set to zero, as we empirically observe that the response value at the region affected more by lighting is generally below average. After that, combining the Eq.3 and 4, the visual guidance map of \( m \)-th frame at the \( l \)-th layer of the network can be derived by:
\[
G^l_m = \text{Maxpool}(\{D_m + S_m\} \times S_m)
\]
where the Max Pooling operation is used to scale the size of the current guidance map \( G^l_m \) to match the feature map \( f^l_m \). In addition, we also perform the normalization and alignment operations for the guidance map \( G^l_m \), which have been discussed in detail in the supplementary material. Finally, the spatial features \( O \in \mathbb{R}^{T \times d} \) with length \( T \) and dimension \( d \) output from the DSN can be simply formulated as:
\[
O = [O^l_1, O^l_2, \ldots, O^l_T], \quad \forall l = 1, 2, \ldots, M
\]
\[
s.t. \quad O^l_m = (f^l_m \odot G^l_m) \odot f^l + f^l_m
\]
where \( M \) represents the number of total network layers used in DSN; \( \odot \) and \( \odot \) represent element-wise product and 1D convolution operations respectively; And \( O^l_m \) means the result of fusing the raw feature stream \( f^l_m \) with the visual guidance map \( G^l_m \). It is then enhanced via RCM module for decoupled temporal representation learning.

### 3.1.2 Decoupled Temporal Feature Learning (DTN)

The temporal representation learning network DTN takes the enhanced spatial features \( O \in \mathbb{R}^{T \times d} \) (formal definition in Sec.3.2) as the input. As shown in Figure 2, the DTN is configured as a multi-branch and two-stage structure to progressively learn the hierarchical temporal representation at local fine-grained level and global coarse-grained level. Specifically, for a single sub-branch \( k \), we first sample a sub-sequence of features \( \hat{O}_k \in \mathbb{R}^{T_n \times d/2} \) of length \( T_n \) from \( \hat{O} \) by a discrete sampling strategy:
\[
\hat{O}_k = \{\hat{O}_t\}_{t=1}^{T_n} \in \mathbb{R}^{T_n \times d/2}, t = 1, 2, \ldots, T_n \}
\]
where \( \mathbb{R}[a, b] \) represents randomly selecting an integer \( x, s.t. \ a \leq x \leq b; \hat{O}_k \) serves as the input of temporal multi-scale features learning module (TMS).

**TMS Module.** The TMS is composed of a time-centric 3D Inception Module\(^2\) \( C_{T-I\text{nc}}(\cdot, W_k) \) and Max Pooling layer. We only use one layer TMS to capture the local fine-grained spatial features \( \hat{O}_k^L \):
\[
\hat{O}_k^L = \text{Maxpool}(C_{T-I\text{nc}}(\hat{O}_k, W_k))
\]
After that, \( \hat{O}_k^L \) is fed into stack of Transformer blocks to learn the coarse-grained temporal representation.

**Transformer block.** To reduce the redundant marginal information in captured temporal features, we utilize a Transformer structure based on \( k\)-NN multi-head self-attention layer [42]. Thus the modeling process of each Transformer block can be formulated as:
\[
\hat{O}_{k}^G = \text{MLP}(\text{LN}(\text{MSA}_{k\text{NN}}(\hat{O}_k^L))) + \hat{O}_{k}^{G-1}
\]
where \( \hat{O}_{k}^{G-1} \) indicates the output feature of the previous layer; \( \text{MSA}_{k\text{NN}}(\cdot) \) indicates the \( k\)-NN multi-head self-attention layer and LN represents layer normalization. Furthermore, to avoid overfitting to one of the sub-branches, we introduce a temperature parameter \( \tau \) to control the sharpness of the output distribution of each sub-branch and impose a constraint loss on it. Therefore, the output of the DTN network can be formulated as:
\[
O_{CLS}^\tau = \sum_{k=1}^{K} \frac{\text{MLP}(O_{k}^{CLS})}{\tau}, \forall k = 1, 2, 3, \ldots, K
\]
where \( O_{CLS}^\tau \) is the class token vector embedded in the Transformer block and \( K \) indicates the number of sub-branches. \( \tau \) follows a cosine schedule from 0.04 to 0.07 during the training. We analytically demonstrate in Sec.5.1 that the tactics of \( k\)-NN Attention, sharpness and multi-loss can bring performance gains for the proposed network.

\(^2\)The size of the convolution kernel in the spatial dimension is \( 1 \times 1 \).
3.2. Recoupling Spatiotemporal Representation

The recoupling strategy is developed to rebuild the space-time interdependence, which reversely applies the distilled inter-frame correlations from time domain into the space domain through a self-distillation-based inner loop optimization mechanism during the training. Specifically, the spatial feature stream $O \in \mathbb{R}^{T \times d}$ derived from the spatial features learning network DSN is first linearly mapped into a low-dimensional space. Then the mapped features $\hat{O} \in \mathbb{R}^{T \times d/2}$ is transposed and fed into the recoupling module RCM.

RCM module. The RCM module is configured as dual pathway to enhance the spatial features from the X (feature dimension) and Y (sequence dimension) directions, as shown in Figure 4. For X direction (intra-frame), inspired by self-attention [8], we utilize a set of learnable matrices: $W_Q \in \mathbb{R}^{T \times T}$ and $W_K \in \mathbb{R}^{T \times T}$ to calculate the attention map $A_X$ (Note that if multiple heads are configured, they will be concatenated and mapped to a fixed dimension).

$$Q = \hat{O}^T W_Q, \quad K = \hat{O}^T W_K, \quad A_X = \delta(\text{GAP}(\frac{Q K^T}{\sqrt{d}})), \quad A_X \in \mathbb{R}^{1 \times T/2}$$ (11)

where $Q$ and $K$ denote the queries and keys, $\delta$ is the Sigmoid activation function, and GAP indicates the adaptive global average pooling operation. The attention map $A_X$ describes the correlation of the intra-frame in the sequence. For Y direction (inter-frame), the $\hat{O}$ is first compressed along the channel dimension through GAP operation to obtain a feature vector with a dimension of $1 \times T$. Then it passes through an MLP block with two hidden layers to obtain a weight embedding $W_{\text{embed}}$.

$$W_{\text{embed}} = \text{MLP}(\text{GAP}(\hat{O}^T)), \quad W_{\text{embed}} \in \mathbb{R}^{1 \times T}$$ (12)

Finally, it is combined with a Sigmoid activation function to describe the correlation of the inter-frame in the sequence.

$$A_Y = \delta(W_{\text{embed}}), \quad A_Y \in \mathbb{R}^{1 \times T}$$ (13)

However, the MLP layer cannot effectively learn the inter-frame correlation from the captured spatial features $\hat{O}$. Therefore, we employ a self-distillation loss function $C_d$ to introduce additional supervision for the MLP block to distill inter-sequence correlation knowledge from temporal domain into $W_{\text{embed}}$. The distillation process can be formulated as:

$$C_d = L_d(CLS, \mathcal{F}(W_{\text{embed}})), \quad CLS = \sum_{k=1}^{K} O_{k}^{CLS}$$ (14)

s.t. $L_d(x, y) = \frac{1}{N_B} \sum_{i=1}^{N_B} KL(x_i / T - y_i / T)$

where $T$ is the distillation temperature parameter, $N_B$ is the batch size, $KL$ indicates Kullback-Leibler divergence [16] and $\mathcal{F}$ indicates linear mapping function. Combining Eq.11 and 13, the attention map for spatial feature enhancement can be derived as:

$$A_{XY} = \sum_{i=1}^{I} \sum_{j=1}^{J} A_{XY,i,j} \times A_{Y,j}, A_{XY} \in \mathbb{R}^{T \times d/2}$$ (15)

where $I$ and $J$ indicate the element index in $A_X$ and $A_Y$ respectively. It is then applied to the raw spatial feature stream $\hat{O}$ by the element-wise product operation:

$$\hat{O} = O \odot A_{XY}$$ (16)

where $\hat{O}$ represents the enhanced spatial feature, which is used for temporal representation learning.

3.3. Cross-modal Interactive Learning

For the RGB-D multi-modal spatiotemporal features separately extracted from the two network branches, we
propose to interact them at the spatial and temporal dimensions respectively at first, as shown in Figure 1. Take the RGB modality as an example, the two spatial features from the RGB-D modalities are first interacted through the MLP layer to generate a joint spatial representation. It is then integrated with raw spatial features of the RGB modality by a residual structure:

$$
\hat{O}_R^2 = \text{LN}(\text{MLP}([O_R^2 || \hat{O}_D^2])) + O_R^2
$$

where $\hat{O}_R^2$ and $\hat{O}_D^2$ denote the enhanced spatial features of RGB-D modalities respectively; And $\parallel$ represents concatenation. Similar to the interactive way of spatial features, after obtaining a joint temporal representation $\hat{O}^T \in \mathbb{R}^{1 \times d/4}$, it is fed into the cross-modal adaptive posterior fusion (CAPF) module for deep multi-modal features fusion.

**CAPF module.** As shown in Figure 5, the CAPF module contains an Encoder and Decoder, which are composed of MLP blocks with multiple hidden layers. The $\hat{O}^T$ is first fed into the Encoder to generate the encoded embedding $\hat{O}^E \in \mathbb{R}^{1 \times d/4}$ which can be used for classification. Then $\hat{O}^E$ passes through the Decoder to obtain a decoded embedding $\hat{O}^D \in \mathbb{R}^{1 \times d/4}$ which can be used to supervise the Encoder. Therefore, the final classification score of the entire network can be obtained by:

$$
C = \text{Argmax}(f^1(\hat{O}_R^{CLS}) + f^2(O_D^{CLS}) + f^3(\hat{O}^E))
$$

where $f^*$ denotes the linear classifier and Argmax is used to take the index of the maximum value in score vector. Meanwhile, considering the challenge of optimizing the MLP layer, the entire network is trained by a multi-loss collaborative optimization strategy as shown in Figure 5.

4. Experiments

4.1. Implementation Details

The proposed method is implemented with Pytorch. The input sequences are randomly/center cropped into $224 \times 224$ during training/inference. We employ SGD as the optimizer with the weight decay of 0.0003 and momentum of 0.9. The learning rate is linearly ramped up to 0.01 during the first 3 epochs, and then decayed with a cosine schedule [23]. The training lasts for 100 epochs. The data augmentation only includes random clipping and rotation. Similar to [43], all of our experiments except NTU-RGBD are pre-trained on 20BN Jester V1 dataset [24]. Moreover, three sub-branches are configured in DTN. The number of spatial and temporal feature learning blocks in DSN and DTN are $M = 6$ and $N = 6$, respectively. We refer to this setting as the basic configuration of our network, unless otherwise specified.

4.2. Comparison with State-of-the-art Methods

The proposed method achieves state-of-the-art (SOTA) performance on four gesture and action datasets. It is noted that we only list some of the SOTA methods for comparison, and more comparison results can be found in the supplementary material.

<table>
<thead>
<tr>
<th>Method</th>
<th>Modality</th>
<th>Accuracy(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transformer [11]</td>
<td>RGB</td>
<td>76.50</td>
</tr>
<tr>
<td>MTUT [1]</td>
<td>RGB</td>
<td>81.33</td>
</tr>
<tr>
<td>NAS [43]</td>
<td>RGB</td>
<td>83.61</td>
</tr>
<tr>
<td>Ours (CAPF)</td>
<td>RGB</td>
<td>89.58</td>
</tr>
<tr>
<td>Transformer [11]</td>
<td>Depth</td>
<td>83.00</td>
</tr>
<tr>
<td>MTUT [1]</td>
<td>Depth</td>
<td>84.85</td>
</tr>
<tr>
<td>NAS [43]</td>
<td>Depth</td>
<td>86.10</td>
</tr>
<tr>
<td>Ours (CAPF)</td>
<td>Depth</td>
<td>90.62</td>
</tr>
<tr>
<td>Ours (Multiplication)</td>
<td>RGB+Depth</td>
<td>90.89</td>
</tr>
<tr>
<td>Ours (Addition)</td>
<td>RGB+Depth</td>
<td>91.10</td>
</tr>
<tr>
<td>Ours (CAPF)</td>
<td>RGB+Depth</td>
<td>91.70</td>
</tr>
</tbody>
</table>

Table 1. Comparison of the SOTA methods on the NvGesture. Similar to [21], wherein Multiplication and Addition are the score-level feature fusion methods.

4.2.1 NvGesture Dataset

The NvGesture [26] dataset focuses on human-car interaction. It in total contains 1532 dynamic gesture videos (1050 for training and 482 for testing) in 25 classes. As can be seen in Table 1, the proposed method significantly boosts performance (RGB: ↑ 5.97%, depth: ↑ 4.52% and RGB-D: ↑ 3.3%) on this dataset for both single- and multi-modal configuration, which demonstrates its generalization ability in the field of human-computer interaction and small dataset. This might be because spatiotemporal decoupled modeling method prevents overfitting of the network to some extent.

4.2.2 THU-READ Dataset

The THU-READ [32] dataset consists 1920 videos with 40 different actions performed by 8 subjects. This dataset is challenging due to small-scale and background noise. For a fair comparison with other SOTA methods, we follow the released leave-one-split-out cross validation protocol utilized in [21]. As illustrated in Table 2, our method exceeds the SOTA results and achieves the best average accuracy (87.40%) in this protocol, which further demonstrates that
our method is also robust to complicated background. We conjecture that this is mainly attributed to the proposed FRP module.

<table>
<thead>
<tr>
<th>Method</th>
<th>Modality</th>
<th>Accuracy(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SlowFast [13]</td>
<td>RGB</td>
<td>69.58</td>
</tr>
<tr>
<td>NAS [43]</td>
<td>RGB</td>
<td>71.25</td>
</tr>
<tr>
<td>Trear [21]</td>
<td>RGB</td>
<td>80.42</td>
</tr>
<tr>
<td>Ours</td>
<td>RGB</td>
<td><strong>81.25</strong></td>
</tr>
<tr>
<td>SlowFast [13]</td>
<td>Depth</td>
<td>68.75</td>
</tr>
<tr>
<td>NAS [43]</td>
<td>Depth</td>
<td>69.58</td>
</tr>
<tr>
<td>Trear [21]</td>
<td>Depth</td>
<td>76.04</td>
</tr>
<tr>
<td>Ours</td>
<td>Depth</td>
<td><strong>77.92</strong></td>
</tr>
<tr>
<td>Ours(Multiplication)</td>
<td>RGB+Depth</td>
<td>86.10</td>
</tr>
<tr>
<td>Ours(Addition)</td>
<td>RGB+Depth</td>
<td>86.25</td>
</tr>
<tr>
<td>Ours(CAPF)</td>
<td>RGB+Depth</td>
<td><strong>87.04</strong></td>
</tr>
</tbody>
</table>

Table 2. Comparison of the SOTA methods on the THU-READ.

4.2.3 IsoGD Dataset

The Chalearn IsoGD [35,36] dataset contains 47,933 RGB-D gesture videos divided into 249 kinds of gestures and is performed by 21 individuals. It is a much harder dataset because (1) it covers gestures in multiple fields and different motion scales from subtle fingertip movements to large arm swings, and (2) many gestures have a high similarity. However, as shown in Table 3, our method also performs well on this dataset, possibly due to the hierarchical and compact features learned from the multi-scale network with the recoupling structure that can capture subtle differences from similar gestures.

4.2.4 NTU-RGBD Dataset

The NTU RGB-D [28] is a large-scale human action dataset, which contains more than 56,000 multi-view videos of 60 actions performed by 40 subjects. This dataset is challenging due to large intra-class and viewpoint variations. As the skeleton information is available, many recent works tend to perform 2D/3D skeleton-based action recognition on this dataset since skeleton inherently highlights the key information of human body, whilst being robust to various illuminations and complex backgrounds. However, the generalization ability and robustness of skeleton-based methods are limited. In this paper, we only use the modalities of color and depth for action recognition. As shown in Table 4, we achieve the SOTA performance on both released protocols: Cross-view (CV) and Cross-subject (CS). Comparing with the skeleton-based SOTA method CTR-GCN [4], the proposed method achieves about 2% improvement on CS protocol and 0.5% on CV protocol. The performance of depth modality can be on par or even better, which further demonstrates the robustness of our method to noisy background and its strong motion perception abilities.

5. Ablation Study

NvGesture and THU-READ(CS2) are employed for the ablation study. All of experiments are conducted on RGB data modality except Sec.5.3. We refer the reader to supplementary material for more ablation studies.
Table 5. Impacts of some introduced components. The Multi-loss means that we impose constraint loss on each sub-branch in DTN.

5.1. Impacts of Embedded Components

In Table 5, we show the impacts by introducing different components on the proposed network. First, we observe that in the absence of FRP module, the network does not work well either on NvGeture or THU-READ datasets, which shows that the early guidance of the network to focus on some local significant regions is beneficial to prevent the model from being trapped into the local optimum. We also demonstrate its robustness to lighting by visualization in supplementary material. In addition, imposing a constraint to each branch of the network can prevent the model from overfitting to one of the branches. Sharpening of the output distribution can encourage each sub-branch to learn more discriminative features. Finally, removing some of redundant information through \( k \)-NN Attention in the temporal representation can also bring certain performance gains.

5.2. Effect of Recoupling Representation Learning

As shown in Figure 6 (a), recoupling learning can boost the performance (Nv: ↑ 3%, THU: ↑ 2%). This is because distilling the knowledge from the temporal domain to enhance the spatial representation can help the network focus more on the informative features during training. The idea behind it may be that the inner loop optimization mechanism based on self-distillation can help the network deviate from the local optima as soon as possible and move toward the global optimal solution. Figure 6 (b) shows the influence of distillation temperature \( T \) on network performance. We observe that different data domains enjoy different temperatures \( T \) during training: 0.4 for gesture dataset and 0.5 for action dataset.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Baseline (Add)</th>
<th>CmSI</th>
<th>CmTI</th>
<th>CmSTI</th>
</tr>
</thead>
<tbody>
<tr>
<td>NvGesture</td>
<td>91.10</td>
<td>91.32</td>
<td>91.53</td>
<td>91.70</td>
</tr>
<tr>
<td>THU-READ</td>
<td>86.35</td>
<td>87.34</td>
<td>88.75</td>
<td>90.00</td>
</tr>
</tbody>
</table>

Table 6. Effects of cross-modal spatiotemporal information interaction. CmSI: Cross-modal Spatial information Interaction only. CmTI: Cross-modal Temporal information Interaction only. CmSTI: Cross-modal spatiotemporal information Interaction.

5.3. Cross-modal Information Interaction

As shown in Table 6, the communication of spatial or temporal information can boost the performance, wherein the information exchange at the spatial or temporal level alone can also bring performance gains, which reflects that the interaction of cross-domain knowledge can help learn discriminative features. The last experiment shows that the joint effective cross-domain information interaction at the spatial and temporal levels can bring the largest performance gains (Nv: ↑ 0.6%, THU: ↑ 3.7%), verifying that multi-modal feature learning can benefit from spatiotemporal independent information transformation.

6. Conclusion

We propose a method for unimodal decoupling and recoupling learning as well as a cross-modal interactive learning and fusion. Firstly, we observe that spatiotemporal recoupling learning is very effective as it can lower the optimization difficulty, especially under small dataset settings. Secondly, guiding the network to focus on the local important areas helps boost the performance. Finally, we prove that interacting with the cross-modal spatiotemporal information in two independent dimensions of space and time, respectively, can encourage the network to extract and fuse multi-modal spatiotemporal features.

7. Acknowledgment

This work was supported by the Alibaba Group through Alibaba Research Intern Program, the External cooperation key project of Chinese Academy Sciences 173211KYSB20200002, the Chinese National Natural Science Foundation Projects 61876179 and 61961160704, the Science and Technology Development Fund of Macau (0025/2019/AM4, 0008/2019/A1, 0010/2019/AJF, 0004/2020/A1 0070/2020/AM4), Guangdong Provincial Key R&D Programme: 2019B010148001, the InnoHK program, and the open Research Projects of Zhejiang Lab No. 2021PF0AB01.
References


