·Article·

Adaptive cross-fusion learning for multi-modal gesture recognition

Benjia ZHOU[1], Jun WAN[2*], Yanyan LIANG[1], Guodong GUO[3]

1. *Macau University of Science and Technology, Macau* 999078, *China*

2. *National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing* 100190, *China*

3. *Baidu Research, Beijing* 100193, *China, and National Engineering Laboratory for Deep Learning Technology and Application, Beijing* 100193, *China*

*** Corresponding author,** jun.wan@ia.ac.cn

Received: 10 October 2020 **Accepted:** 12 December 2020

Supported by the Chinese National Natural Science Foundation Projects (61961160704, 61876179); the Key Project of the General Logistics Department(ASW17C001); the Science and Technology Development Fund of Macau(0010/2019/AFJ, 0025/2019/AKP).

Citation: Benjia ZHOU, Jun WAN, Yanyan LIANG, Guodong GUO. Adaptive cross-fusion learning for multi-modal gesture recognition. Virtual Reality & Intelligent Hardware, 2021, 3(3): 235—247
DOI: 10.1016/j.vrih.2021.05.003

Abstract **Background** Gesture recognition has attracted significant attention because of its wide range of potential applications. Although multi-modal gesture recognition has made significant progress in recent years, a popular method still is simply fusing prediction scores at the end of each branch, which often ignores complementary features among different modalities in the early stage and does not fuse the complementary features into a more discriminative feature. **Methods** This paper proposes an Adaptive Cross-modal Weighting (ACmW) scheme to exploit complementarity features from RGB-D data in this study. The scheme learns relations among different modalities by combining the features of different data streams. The proposed ACmW module contains two key functions: (1) fusing complementary features from multiple streams through an adaptive one-dimensional convolution; and (2) modeling the correlation of multi-stream complementary features in the time dimension. Through the effective combination of these two functional modules, the proposed ACmW can automatically analyze the relationship between the complementary features from different streams, and can fuse them in the spatial and temporal dimensions. **Results** Extensive experiments validate the effectiveness of the proposed method, and show that our method outperforms state-of-the-art methods on IsoGD and NVGesture.

Keywords Gesture recognition; Multi-modal fusion; RGB-D

1 Introduction

Gesture recognition is attracting increasing attention in both research and industrial communities because of its vast applications[1−4], such as human-computer interaction[5] and video surveillance. Because complementary feature learning can benefit from different data modalities within different aspects, multi-

modal gesture recognition technology[6−9] has been proposed. For example, it can be easy to distinguish foregrounds (*i.e.,* face, hands, and arms) from backgrounds with the help of depth modality, whereas RGB data can provide a higher texture/color appearance modality. However, a problem of multi-modal gesture recognition is how to effectively fuse the feature representations extracted from different data modalities, which is not a trivial task for gesture recognition.

According to multi-modal gesture recognition, the effective fusion of complementary feature learning from such data benefit gesture recognition. Previous fusion methods have attempted to directly add, multiply, or average the final Softmax scores[6,8,10−12]. However, gesture recognition often focuses on information regarding subtle changes in hand/arm movements, which can often be captured in intermediate stages. These methods are equivalent to using the fusion machine for global representation, and do not consider the subtle changes in hand/arm movements in the intermediate stage. Thus, the results of different modalities of data can be fused in the network, rather than trained separately or combined as late fusion.

Motivated by the above observations, to exploit the spatio-temporal correlation among different modalities of data, we propose an adaptive cross-modal weighting (ACmW) module that is employed in different stages throughout the network. Unlike the one-off fusion strategies[10,13,14], inspired by the message passing mechanism of FishNet[15], we use the ACmW module to generate the "spatio-temporal correlation message" of different modalities, and combine it with the original data stream for successive feature learning. This strategy can avoid losing details of the single modality of data during the early stage. Our ACmW module can accept two inputs of different sizes, and the outputs have the same size as the original feature maps. With this design, it can be embedded in any network architecture. The network can be end-to-end trainable and focus on gesture-related features, even with multi-modal inputs. Our contributions can be summarized as follows:

(1) We propose an adaptive fusion module called ACmW. Unlike the previous offline multi-modal fusion scheme[6,16], ACmW enables the network to train different data modalities in an end-to-end manner. It also leverages the strength of different modalities of data by generating a "spatio-temporal correlation message" and combining them from different streams instead of simply late-fusing the score of different data.

(2) Extensive experiments prove that the integration of our designs can ultimately improve the performance of gesture recognition. The experiment results demonstrate that our method can strike a balance between a good performance and low computation burden, and that it outperforms the top techniques on two large-scale benchmark gesture datasets: IsoGD and NVGesture.

The rest of this paper is organized as follows. Related works are introduced in Section 2. Section 3 introduces the proposed ACmW module. Section 4 provides the details of our experiment, including a performance evaluation of the ACmW module on two benchmark gesture datasets, comparing the results with those of other state-of-the-art methods. At the end of this section, we visualize the neural activation. Finally, we provide some concluding remarks in Section 5.

2 Related work

In this section, we first introduce recent developments in multi-modal gesture recognition. Then, some recent progress in feature fusion strategies will be listed.

2.1 Gesture recognition based on multi-modal approach

For gesture recognition, the most challenging task is to allow the network to adaptively focus on the motion of the hands and arms without being affected by background noise. Previous studies utilized handcrafted features[17−20] and deep neural network extracted features[21−29] for gesture recognition.

Meanwhile, with the release of RGB-D sensors in recent years, simultaneously captured RGB and depth data are easily available, which promotes the development of multi-modal gesture recognition technology. Miao et al. proposed a multi-modal gesture recognition method based on a ResC3D network to overcome the barriers of gesture-irrelevant factors[6]. Tran et al. proposed a multi-modal continuous gesture recognition method, which consists of two modules: segmentation and recognition[30]. In addition to the method based on a convolutional neural networks (CNN), Zhu et al. utilized the LSTM variants AttenConvLSTM[31] and PreRNN[32] for RGB-D gesture recognition. All of these methods first train different branches of the network on different data modalities, and then, combine the Softmax scores they predicted. An advantage of these methods is that the errors in the fusion scores come from different branches; thus, they do not affect each other and will not lead to further accumulation of errors. As mentioned by Roitberg et al.[8], the disadvantage of this method is that it ignores many intermediate representations that have a significant impact on the classification performance. Therefore, a fusion strategy of multi-modal features has attracted the attention of researchers.

2.2 Multi-modal fusion strategy

The leveraging of multi-modal data can be found in many previous studies. Decision- and feature-level fusion are two common strategies in multi-modal gesture recognition. Decision-level fusion[14,33-35] techniques are easily implemented but only concern the majority, and other types of data cannot help in the final recognition. Feature-level fusion[6,8-10,16,36,37] contains sufficient information of all features and avoids the complicated pre-processing of registration, owing to its uniform dimension. Among these methods, the Roitberg et al.[8] fusion strategy is the most similar to our own. However, instead of directly using a convolutional layer for fusion, we designed a more comprehensive fusion scheme, which can model the temporal correlation among different data modalities and fuse the extracted multi-stream features at an early stage, thereby enhancing the spatio-temporal representation. Specifically, the features of different modalities are first unfolded into 1-D vectors, and we then use different kernels of convolutional layers, which can learn a more adaptive fusion feature for each modality of data. The network can then exploit the complementary spatio-temporal information of the other modality of the data according to the property of the current data stream, rather than simply adding the blended features to it. At the same time, inspired by a study conducted by Hu et al.[38], which proposed a channel-based attention mechanism, we model the correlation of multi-stream features in a temporal series to enhance the temporal representation, to achieve a fusion of multi-stream features. See Section 3 for more details.

3 Methodology

In this section, we first formulate the structure of the proposed ACmW module in Section3.1. We then introduce the implementation of the ACmW module in Section 3.2. Finally, the details of the ACmW network architecture are provided in Section 3.3.

3.1 Adaptive cross-modal fusion scheme

As mentioned before, the different modalities of data can be complementarity to each other. To improve the recognizing accuracy, a comprehensive scheme should be sophisticatedly designed by exploiting and combining the advantages of different modalities of the data. As shown in Figure 1, the ACmW module takes the features from the RGB and depth branches as inputs and conducts adaptive convolution to derive two groups of weighted feature maps, rather than simply blending the inputs to generate one fused data. This guarantees that the fusion process can be used to learn complementary features of a multi-modality

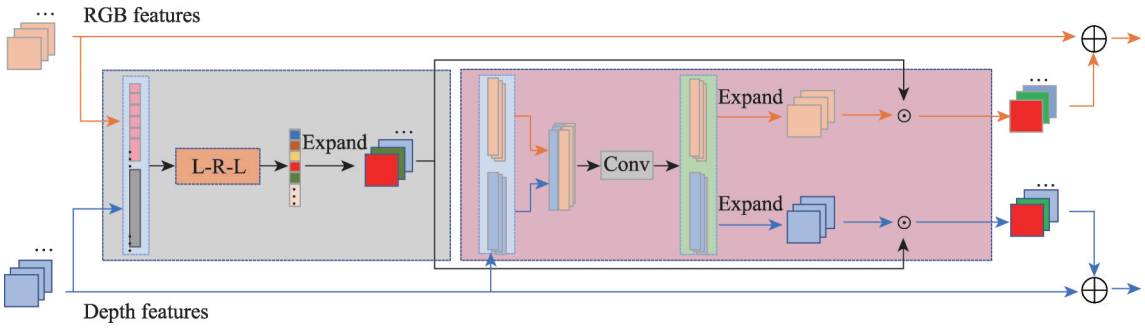

Figure 1 Structure of the ACmW module. Here, ⊙ indicates an element-wise product, ⊕ indicates an element-wise sum, and the different colors of feature maps indicate different weight values. We applied two fusion strategies for multiple feature streams. The first time is temporal-based fusion (the left part), which mainly produces a temporal descriptor by aggregating feature maps across their spatial dimensions ($C \times H \times W$) to learn the correlation of the multi-stream features on the temporal series through the linear layers. The second time is spatial-based fusion (the right part), which mainly produces more representative features using an adaptive convolution layer (a 3D convolution with a kernel and step size of 1). By combining these two fusion strategies, we can make full use of the semantic information of high-level features and fine-grained information of low-level features.

from low-level visual features to high-level semantic features. ACmW mainly contains two sub-structures: the spatial feature fusion mechanism and temporal series-based adaptive fusion mechanism.

For the spatial feature fusion mechanism, as described in Equation 1, both RGB and depth features are initially unfolded. The unfolding processing is achieved by stretching the features into a one-dimensional vector, which makes the fusion more efficient.

$$z_i = C\left(F\left(map\left(x_i, y_i\right)\right), W\right), \tag{1}$$

where x_i and y_i indicate the feature maps of a specific stage, i, of the RGB and depth branches, respectively. Function $map(\cdot)$ achieves the unfolding process, and C indicates the convolution operation.

Later, as described in Equation 2, the unfolded features are concatenated in the unfolded dimension by function $F(\cdot)$ and sent to the adaptive convolution, which has two convolution kernels. Note that the number of kernels is in accord with that of the data types. Adaptive convolution is achieved by a $1 \times 1 \times 1$ convolution with the weights of W. Next, we can obtain two weighted features, having the same shape as the original, through feature unfolding:

$$x_i' = F\left(z_i\right), \quad y_i' = F\left(z_i\right), \tag{2}$$

where x_i' and y_i' indicate the fused feature maps.

For the temporal series-based adaptive fusion mechanism, inspired by a previous study[39], we model the temporal series, instead of the channel, to learn the correlation of the multi-modal features. Specifically, both RGB and depth feature maps are mapped into a one-dimensional vector in the temporal dimension using transformation function F_{tr}, which is formulated in Equation 3.

$$\tilde{x}_i = F_{tr}\left(z_i\right), \ \tilde{y}_i = F_{tr}\left(z_i\right). \tag{3}$$

They are then concatenated by the column, that is, $\tilde{z}_i = [[\tilde{x}_i], [\tilde{y}_i]]$. As described in Equation 4, after passing through the fully connected (FC) layers to obtain a weight vector with the same shape as the temporal dimension of the single branch input, we finally expand this weight vector to the same number of dimensions as the raw input features. Moreover, an element-wise product is employed to enhance the temporal representation of multi-modal features to achieve the deep fusion of multi-stream features within the time dimension. By enhancing the temporal representation and fusing the spatial information, the ACmW module can effectively aggregate spatio-temporal features.

$$L_i = E\left(\sigma\left(g\left(\tilde{z}_i, W\right)\right)\right) = E\left(\sigma\left(W_2\delta\left(W_1\tilde{z}_i\right)\right)\right), \tag{4}$$

where δ refers to the ReLU function, and σ refers to the sigmoid function. In addition, $W_1 \in \mathbb{R}^{(r\times C)\times C}$ and $W_2 \in \mathbb{R}^{C\times C}$, where r is the number of branches and $E(\cdot)$ indicates the expanded function.

To preserve the identical information of the original data modality, an element-wise sum that combines the weighted results and original features is used.

$$x_i^o = E\left(x_i'\right)\odot L_i + x_i,\ y_i^o = E\left(y_i'\right)\odot L_i + y_i, \tag{5}$$

where x_i^o and y_i^o indicate the outputs of the ACmW module, and $\odot$ indicates the element-wise product.

Because the parameters of the adaptive convolution are learned by the network itself, the fusion of the features can be adaptive. Meanwhile, because the ACmW module does not derive one fused output stream directly, each branch can still learn the identical features of the corresponding data, and the complementarity of the different data can be exploited from low-level to high-level features.

3.2 Implementation of ACmW

As shown in Figure 2, both branches adopt the same base backbone, such as C3D and 3D ResNet-50 (Res3D). We fuse multi-stream features at different stages of the network. Meanwhile, to avoid losing the original feature information, an element-wise sum operation is utilized to combine the original and fused features. In addition, for the final prediction, instead of discarding the prediction scores of individual branches, we completely combine them with the scores of the fusion layer, which significantly improves the performance of the multi-branch network. For the C3D network structure, we extract the two feature streams after the pooling layer of each branch and input them into the ACmW module for fusion.

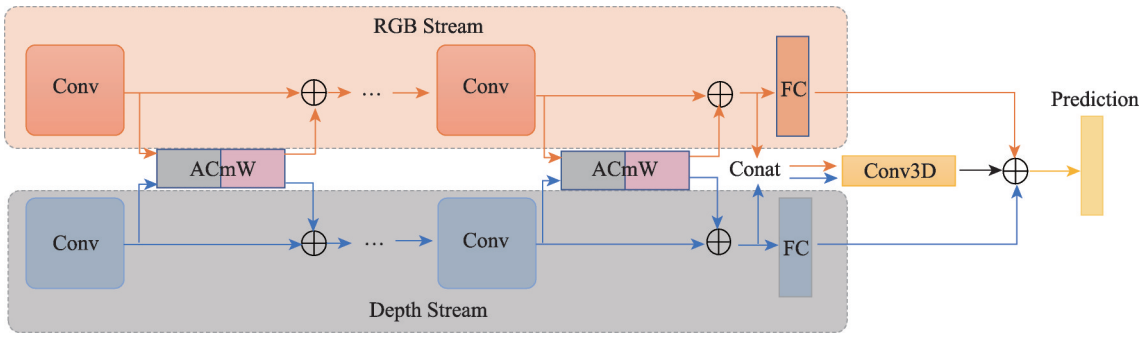

Figure 2 An overview of the multi-stream classification model. The ACmW module is embedded between the two network branches (RGB and depth) from the early to late stage for feature fusion, where ⊕ indicates the element-wise sum. The RGB branch carries visual information about the scenes and objects in the video, and the depth branch significantly eliminates background noise.

The fused features are then taken as the succeeding input of the next layer. Thus, a total of five ACmW modules are embedded in a cascade way in C3D. For the Res3D network, the ACmW modules are embedded behind each residual block for feature fusion, and a total of four ACmW modules are therefore embedded in a cascade manner in Res3D.

3.3 Details of the ACmW network architecture

In this section, we present the details of the proposed ACmW module. Taking the C3D as a backbone, as shown in Table 1, we give the size of the features, floating point operations per second (FLOPs), and parameters of each stage. Taking stage 1 as an example, we first conduct temporal-based fusion, which

Table 1 Details on the ACmW network architecture. Here, we utilize C3D as the backbone to analyze the details of the ACmW

	Feature Streams	Fusion	Output Size	FLOPs		#Parameters	
				C3D	ACmW	C3D	ACmW
Stage1	RGB Features + Depth Features	Spatial Temporal ⊙	$N \times 64 \times 32 \times 56 \times 56$ $N \times 64 \times 32 \times 56 \times 56$ $N \times 64 \times 32 \times 56 \times 56$	8.6 GB	154.2 MB	5.3 KB	3.1 KB
Stage2	RGB Features + Depth Features	Spatial Temporal ⊙	$N \times 128 \times 16 \times 28 \times 28$ $N \times 128 \times 16 \times 28 \times 28$ $N \times 128 \times 16 \times 28 \times 28$	88.9 GB	38.5 MB	221.6 KB	772.0 B
Stage3	RGB Features + Depth Features	Spatial Temporal ⊙	$N \times 256 \times 8 \times 14 \times 14$ $N \times 256 \times 8 \times 14 \times 14$ $N \times 256 \times 8 \times 14 \times 14$	133.3 GB	9.6 MB	2.7 MB	196.0 B
Stage4	RGB Features + Depth Features	Spatial Temporal ⊙	$N \times 512 \times 4 \times 7 \times 7$ $N \times 512 \times 4 \times 7 \times 7$ $N \times 512 \times 4 \times 7 \times 7$	66.6 GB	2.4 MB	10.6 MB	52.0 B
Stage5	RGB Features + Depth Features	Spatial Temporal ⊙	$N \times 512 \times 1 \times 4 \times 4$ $N \times 512 \times 1 \times 4 \times 4$ $N \times 512 \times 1 \times 4 \times 4$	11.1 GB	196.6 KB	14.2 MB	7.0 B
Total	—	—	—	308.7 GB	204.9 MB	79.0 MB	4.1 KB

drives the multi-stream features into a vector with the shape of $N \times 1 \times 32 \times 1 \times 1$, and then, expands it to a $N \times 64 \times 32 \times 56 \times 56$ shape to match the spatial-based features. For spatial-based fusion, we first stretch the features into a one-dimensional vector with a shape of $N \times 1 \times 6422528$ (in each channel, we stretch the features with a size of $32 \times 56 \times 56$ into a one-dimensional vector with a size of 1×100352, and the 64-channel feature is then concatenated together into a one-dimensional feature, the size of which is 1×64225284). This step makes the fusion more efficient. Then, the unfolded features are concatenated in the unfolded dimension, which has the shape of $N \times 2 \times 6422528$, and are then sent to the adaptive convolution layer, which has two convolution kernels. Next, we obtain two weighted features. These two features are reshaped to the same size ($N \times 64 \times 32 \times 56 \times 56$) as the original through the feature unfolding part. In addition, the total FLOPs and parameters of the ACmW are about 204.9 M and 4.1 K, whereas the single C3D network branch is about 308.7 G and 79.0 M, which clearly shows that it is an extremely lightweight network.

4 Experiments

In this section, we first present the details of the benchmark datasets in Section 4.1, which are used to evaluate our method. Then, the implementation details of the experimental setup are given in Section 4.2. Finally, we thoroughly evaluate the impacts of the ACmW module by embedding it into different backbones on two benchmark datasets in Section 4.3.

4.1 Datasets

We evaluate our method on two RGB-D gesture datasets: the Chalearn IsoGD dataset[40] and NVGesture dataset[7]. As shown in Figure 3, NVGesture comprises constrained driving gestures, whereas IsoGD contains multiple types of gestures, e.g., mudra and diving gestures, that are in an unconstrained setting.

Chalearn IsoGD dataset. The Chalearn IsoGD dataset was proposed by Wan et al.[40]. It contains 47, 933 RGB-D gesture videos divided into 249 types of gestures performed by 21 individuals. The dataset has

Figure 3 Some example images from different benchmark datasets: (a), (b) RGB frames and corresponding depth frames from the ChaLearn IsoGD dataset. (c), (d) RGB frames and corresponding depth frames from the NVGesture dataset.

three subsets, i. e., training, validation, and test sets, which contain 35878, 5784, and 6271 samples, respectively. The samples in the three subsets are excluded. It is also used as a benchmark for two rounds of the Chalearn LAP large-scale isolated gesture recognition challenge.

NVGesture dataset. NVGesture[7] focuses on touchless driver control. It contains 1532 dynamic gestures that are separated into 25 classes, which involve RGB and depth videos and a pair of stereo-IR streams. This dataset is divided into training and testing subsets with at a ratio of 7:3, namely, 1050 samples for training and 482 for testing. Unlike work of Molchanov et al.[7] that used all modalities to obtain the result, because we consider RGB-D gesture recognition, we experiment using only RGB-D data.

4.2 Experimental setup

Our experiments are all conducted using Pytorch[41] on the RTX 2080 Ti GPU. During the training stage, the input frames are spatially resized to 256 × 256, and then, cropped to 224 × 224 randomly, whereas during the inference stage, they are cropped at the center. We randomly sample 32 frames in the video, train the network with a mini-batch of 32 samples, and utilize the SGD optimizer with a weight decay of 0.0005 and momentum of 0.9. The initial learning rate was 0.01, which is reduced by ten-fold when the accuracy on the validation set does not improve every three epochs. The training stage is stopped after the learning rate becomes less than le-5.

4.3 Ablation study

The selected backbone networks are C3D[37] and 3D ResNet-50[42]. All backbone networks are pre-trained on the 20BN Jester V1 dataset[1].

Impact of ACmW module on C3D network. In these experiments, we use C3D as the backbone to study the impact of the ACmW module. C3D can simultaneously model the appearance and motion information, and is more suitable for spatio-temporal feature learning than 2D ConvNets. The training process is divided into two stages: (1) training two C3D network branches on the RGB and depth datasets, and (2) training as shown in Figure 2. First, the ACmW module is embedded between the RGB and depth

[1] https://20bn.com/datasets/jester

network branches in a cascading manner. Then, the RGB and depth branches are fine-tuned using the weight of the first stage of training. Finally, we used a small learning rate (0.001 in this experiment) and the Adam optimizer to train the entire network. After a few (about 7 to 10) epochs, the network converged. In addition, to reduce the number of parameters based on a previous study[43], we finally use a $1 \times 1 \times 1$ convolution layer, instead of a fully connected layer, to predict the final classification probability. Figure 4 shows the accuracy of different fusion results on the IsoGD and NVGesture datasets. Notably, compared with the performance of ACmW and other common fusion schemes, i.e., the score fusion and element-wise multiplication fusion, our fusion strategy significantly improves the recognition accuracy.

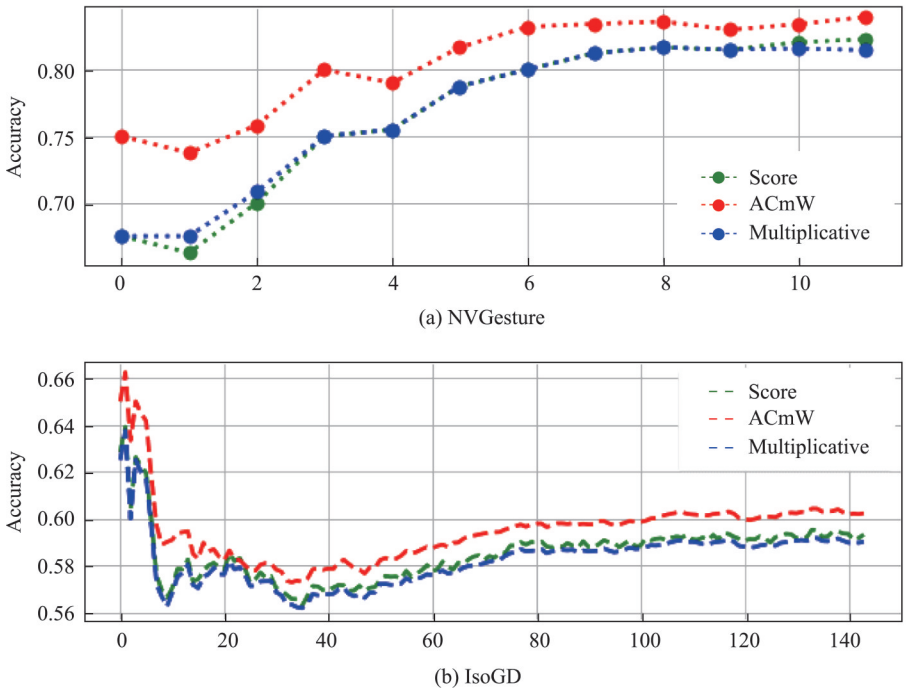

Figure 4 Impact of ACmW module on 3D ResNet-50. (a) Fusion results on the NVGesture test set; (b) Fusion results on the IsoGD validation set.

The accuracies of the different fusion results are shown in Table 2. The score fusion method is used to conduct the fusion in the latest stage, which makes the decision based only on the maximum probability of the predictions with different modalities. Element-wise multiplication fusion involves multiplying the predicted probability values in different modalities to obtain a new probability distribution. These two fusion methods cannot sufficiently exploit the advantages of different data modalities, and do not consider the correlation of the temporal series in video-based classification tasks. Therefore, they clearly cannot achieve a high score on the performance of any single modality of data. By comparison, the ACmW can expand the complementarity spatially and temporally throughout the network, which helps the features of different modalities focus on the gesture. Consequently, it can achieve the best performance. Specifically, the performance on these two benchmark datasets can be about 1% higher than the score fusion, and about

Table 2 Comparison of different fusion results on the C3D network. For the IsoGD dataset, because the results of most methods are from the validation subset, we also conducted our experiments using this subset for a fair comparison

Dataset	RGB	Depth	Fusion Strategy		
			Score Fusion	Multiplicative Fusion	ACmW
IsoGD	53.18%	54.22%	58.10%	57.42%	59.97%
NVGesture	78.54%	80.83%	82.16%	81.33%	83.96%

2% higher than the multiplicative fusion.

Impact of ACmW module on 3D ResNet-50 network. In this experiment, we use 3D ResNet-50 as the backbone to study the impact of the ACmW module. Similar to C3D, 3D ResNet-50 also uses a 3D convolution kernel to extract the spatio-temporal representations. However, Res3D outperforms networks such as C3D on large datasets. The training process is the same as that of C3D, whereas in 3D ResNet-50; the ACmW module is embedded after each residual block, and the features fused after the last residual block are input into the $1 \times 1 \times 1$ convolution to conduct the final fusion score prediction. Figure 5 shows the performance improvement of the ACmW module embedded in the dual-branch 3D ResNet-50 network. Compared with the other fusion strategies, ACmW has a significant improvement on these two datasets. Table 3 demonstrates this more clearly, where the performance on both the IsoGD and NVGesture datasets is about 1% higher than the score fusion and about 2% higher than multiplicative fusion.

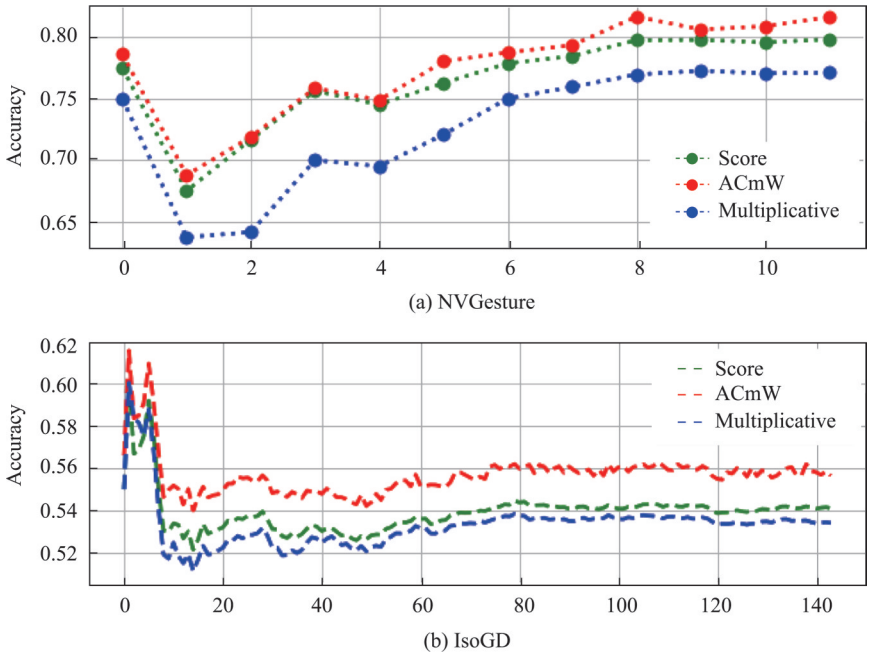

Figure 5 Impact of ACmW module on 3D ResNet-50. (a) Fusion results on NVGesture testset; (b) Fusion results on IsoGD validation set.

Table 3 Comparison of different fusion results on 3D ResNet-50 network

Dataset	RGB	Depth	Fusion Strategy		
			Score Fusion	Multiplicative Fusion	ACmW
IsoGD	47.12%	49.39%	54.12%	53.45%	56.23%
NVGesture	75.83%	77.29%	79.67%	76.97%	81.32%

4.4 Comparison with state-of-the-art methods

After studying the components described in Section 4.3, we evaluate the performance of the ACmW module on two benchmark datasets. Our method is compared with recent state-of-the-art methods on IsoGD and NVGesture datasets.

For the IsoGD dataset, because most of the methods release their result on the validation subset, we also conduct our experiments for a fair comparison. As shown in Table 4 and Table 5, existing video-based classification tasks adopt[34] 3DCNNs to first learn RGB- and depth-based network branches, respectively, and then, give the final classification result by combining the prediction results from them. Although this

average fusion method can improve the performance of the final classification, it is still challenging to fuse multi-stream features to obtain a more semantic representation. Our ACmW module is an intermediate fusion-based strategy, which mainly converts different modal data into a high-dimensional feature representation, and then, fuses them. As the main advantage of this method, the fusion location can be flexibly selected. Through this fusion strategy, the performance can be improved by about 5% on the IsoGD dataset and about 0.1% on the NVGesture dataset.

Table 4 Comparisons with state-of-the-art methods on IsoGD dataset

Method	Modality	Accuracy (%)
c-ConvNet[44]	RGB-D	44.80
Pyramidal C3D[45]	RGB-D	45.02
2SCVN+3DDSN[46]	RGB-D	49.17
32-frame C3D[47]	RGB-D	49.20
AHL[48]	RGB-D	54.14
3DCNN+LSTM[49]	RGB-D	55.29
ACmW (Ours)	RGB-D	59.97

Table 5 Comparisons with state-of-the-art methods on NVGesture dataset

Method	Modality	Accuracy (%)
HOG + HOG^2 [50]	RGB-D	36.90
I3D[51]	RGB-D	83.82
ACmW (Ours)	RGB-D	83.96

4.5 Feature visualization

The neural activation is shown in Figure 6. It can be seen from Figure 6 that the proposed ACmW module can effectively fuse spatio-temporal representations to drive the model to focus more on the movement of the arms and hands. Clearly, we can see that ACmW has a significant effect on the appearance of the feature maps. Integrating the advantages of the RGB and depth modalities indicates the contextual information of the movement path. The ACmW module not only marks the regions related to the gesture, such as the arm of the performer, but it also distinguishes the ranges of motion at different positions of the video sequence. It effectively avoids the impact of noise on the feature, which is presented without an attention mechanism, particularly when a drastic movement, such as the raising or dropping of an arm, occurs. Therefore, our ACmW module can better guide the network to focus on the hand and arm, and provide a more accurate prediction.

Figure 6 Feature visualization from dual-stream C3D network embedded with ACmW module on the IsoGD validation set.

5 Conclusion

In this study, we developed an ACmW scheme to exploit the complementarity features from RGB-D data through the network. The main functions of the ACmW module are exploring the correlation of multi-stream characteristics in the temporal dimension, and fusing the spatial representations of multi-stream features. Through an effective combination of these two functions, multi-stream features from different data modalities are deeply fused in the temporal and spatial dimensions. Extensive experiments show the

effectiveness of our approach. Future directions include exploring the fusion performance of the ACmW module on more than two feature streams, and proving the applicability of the ACmW module in 2D convolutional networks.

Declaration of competing interest

We declare that we have no conflict of interest.

References

1 Liu X, Shi H L, Hong X P, Chen H Y, Tao D C, Zhao G Y. 3D skeletal gesture recognition via hidden states exploration. IEEE Transactions on Image Processing, 2020, 29: 4583−4597
DOI:10.1109/tip.2020.2974061

2 Liu X, Zhao G. 3D skeletal gesture recognition via discriminative coding on time-warping invariant riemannian trajectories. IEEE Transactions on Multimedia, 2020, 99: 1
DOI:10.1109/TMM.2020.3003783

3 Rautaray S S, Agrawal A. Vision based hand gesture recognition for human computer interaction: a survey. Artificial Intelligence Review, 2015, 43(1): 1−54
DOI:10.1007/s10462-012-9356-9

4 Weissmann J, Salomon R. Gesture recognition for virtual reality applications using data gloves and neural networks. In: IJCNN. IEEE, 1999

5 Sun Y, Xu C, Li G F, Xu W F, Kong J Y, Jiang D, Tao B, Chen D S. Intelligent human computer interaction based on non redundant EMG signal. Alexandria Engineering Journal, 2020, 59(3): 1149−1157
DOI:10.1016/j.aej.2020.01.015

6 Miao Q, Li Y, Ouyang W, Ma Z, Cao X. Multimodal gesture recognition based on the ResC3D network. In: 2017 IEEE International Conference on Computer Vision Workshop (ICCVW). IEEE, 2017

7 Molchanov P, Yang X, Gupta S, Kim K, Kautz J. Online detection and classification of dynamic hand gestures with recurrent 3D convolutional neural networks. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2016

8 Roitberg A, Pollert T, Haurilet M, Martin M, Stiefelhagen R. Analysis of deep fusion strategies for multi-modal gesture recognition. In: CVPR Workshops. 2019

9 Wang P C, Li W Q, Ogunbona P, Wan J, Escalera S. RGB-D-based human motion recognition with deep learning: a survey. Computer Vision and Image Understanding, 2018, 171: 118−139
DOI:10.1016/j.cviu.2018.04.007

10 Li Y, Miao Q, Tian K, Fan Y, Xu X, Li R, Song J. Large-scale gesture recognition with a fusion of RGB-D data based on the C3D model. In: ICPR. IEEE, 2016

11 Neverova N, Wolf C, Taylor G, Nebout F. ModDrop: adaptive multi-modal gesture recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 38(8): 1692−1706
DOI:10.1109/tpami.2015.2461544

12 Pitsikalis V, Katsamanis A, Theodorakis S, Maragos P. Multimodal gesture recognition via multiple hypotheses rescoring. In: Gesture Recognition. Springer International Publishing, 2017, 467−496
DOI:10.1007/978-3-319-57021-1_16

13 Wang P, Li W, Liu S, Gao Z, Tang C, Ogunbona P. Large-scale isolated gesture recognition using convolutional neural networks. In: ICPR. IEEE, 2016

14 Zhu G M, Zhang L, Shen P Y, Song J. Multimodal gesture recognition using 3D convolution and convolutional LSTM. IEEE Access, 2017, 5: 4517−4524
DOI:10.1109/access.2017.2684186

15 Sun S, Pang J, Shi J, Yi S, Ouyang W. Fishnet: A versatile backbone for image, region, and pixel level prediction. In: NIPS. 2018

16 Narayana P, Beveridge J R, Draper B A. Gesture recognition: focus on the hands. In: 2018 IEEE/CVF Conference on

Computer Vision and Pattern Recognition. IEEE, 2018

17 Malgireddy M R, Inwogu I, Govindaraju V. A temporal Bayesian model for classifying, detecting and localizing activities in video sequences. In: Computer Vision & Pattern Recognition Workshops. IEEE, 2012

18 Wan J, Guo G D, Li S Z. Explore efficient local features from RGB-D data for one-shot learning gesture recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 38(8): 1626–1639
DOI:10.1109/tpami.2015.2513479

19 Wan J, Ruan Q Q, Li W, An G Y, Zhao R Z. 3D SMoSIFT: three-dimensional sparse motion scale invariant feature transform for activity recognition from RGB-D videos. Journal of Electronic Imaging, 2014, 23(2): 023017
DOI:10.1117/1.jei.23.2.023017

20 Wan J, Ruan Q Q, Li W, Deng S. One-shot learning gesture recognition from RGB-D data using bag of features. In: Gesture Recognition. Springer International Publishing, 2017, 329–364
DOI:10.1007/978-3-319-57021-1_11

21 Ji X P, Cheng J, Tao D P, Wu X Y, Feng W. The spatial Laplacian and temporal energy pyramid representation for human action recognition using depth sequences. Knowledge-Based Systems, 2017, 122: 64–74
DOI:10.1016/j.knosys.2017.01.035

22 Zhuang L, Liu Z, Chai X, Chen X. Continuous gesture recognition with hand-oriented spatiotemporal feature. In: IEEE International Conference on Computer Vision Workshop. IEEE Computer Society, 2017

23 Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition invideos. In: NIPS. 2014

24 Wang P, Li W, Wan J, Ogunbona P, Liu X. Cooperative training of deep aggregation networks for RGB-D action recognition. In: AAAI. 2018

25 Zhang L, Zhu G, Mei L, Shen P, Shah S A A, Bennamoun M. Attention in convolutional lstm for gesture recognition. In: NIPS. 2018

26 Duan H J, Sun Y, Cheng W T, Jiang D, Yun J T, Liu Y, Liu Y B, Zhou D L. Gesture recognition based on multi-modal feature weight. Concurrency and Computation: Practice and Experience, 2020
DOI:10.1002/cpe.5991

27 He Y, Li G F, Liao Y J, Sun Y, Kong J Y, Jiang G Z, Jiang D, Tao B, Xu S, Liu H H. Gesture recognition based on an improved local sparse representation classification algorithm. Cluster Computing, 2019, 22(S5): 10935–10946
DOI:10.1007/s10586-017-1237-1

28 Jiang D, Zheng Z J, Li G F, Sun Y, Kong J Y, Jiang G Z, Xiong H G, Tao B, Xu S, Yu H, Liu H H, Ju Z J. Gesture recognition based on binocular vision. Cluster Computing, 2019, 22(S6): 13261–13271
DOI:10.1007/s10586-018-1844-5

29 Jiang D, Li G F, Sun Y, Kong J Y, Tao B. Gesture recognition based on skeletonization algorithm and CNN with ASL database. Multimedia Tools and Applications, 2019, 78(21): 29953–29970
DOI:10.1007/s11042-018-6748-0

30 Tran D, Bourdev L, Fergus R, Torresani L, Paluri M. Learning spatiotemporal features with 3D convolutional networks. In: ICCV. 2015

31 Zhu G M, Zhang L, Lin Y. Redundancy and attention in convolutional LSTM for gesture recognition. IEEE Transactions on Neural Networks and Learning Systems, 2019

32 Yang X, Molchanov P, Kautz J. Making convolutional networks recurrent for visual sequence learning. In: CVPR. 2018

33 Wang H, Wang P, Song Z, Li W. Large-scale multimodal gesture recognition using heterogeneous networks. In: ICCV Workshops. 2017

34 Zhang L, Zhu G, Shen P, Song J, Shah S A, Bennamoun M. Learning spatiotemporal features using 3DCNN and convolutional lstm for gesture recognition. In: ICCV. 2017

35 Zhu G, Zhang L, Mei L, Shao J, Song J, Shen P. Large-scale isolated gesture recognition using pyramidal 3D convolutional networks. In: ICPR. IEEE, 2016

36 Li Y, Miao Q, Tian K, Fan Y, Xu X, Li R, Song J. Large-scale gesture recognition with a fusion of RGB-D data based on saliency theory and C3D model. TCSVT, 2018, 28(10): 2956–2964

37 Kopuklu O, Kose N, Rigoll G. Motion fused frames: data level fusion strategy for hand gesture recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2018

38 Hu J, Shen L, Sun G. Squeeze-and-excitation networks. In: CVPR. IEEE, 2018
39 Hu T K, Lin Y Y, Hsiu P C. Learning adaptive hidden layers for mobile gesture recognition. In: AAAI. 2018
40 Wan J, Zhao Y, Zhou S, Guyon I, Escalera S, Li S Z. Chalearn looking at people RGB-D isolated and continuous datasets for gesture recognition. In: CVPR Workshops. 2016
41 Paszke A, Gross S, Chintala S, Chanan G, Yang E, DeVito Z, Lin Z, Desmaison A, Antiga L, Lerer A. Automatic differentiation in pytorch. In: NIPS 2017 Workshop Autodiff Submission. 2017
42 Li Y N, Miao Q G, Tian K, Fan Y Y, Xu X, Ma Z X, Song J F. Large-scale gesture recognition with a fusion of RGB-D data based on optical flow and the C3D model. Pattern Recognition Letters, 2019, 119: 187−194
DOI:10.1016/j.patrec.2017.12.003
43 Carreira J, Zisserman A. Quo vadis, action recognition? a new model and the kinetics dataset. In: CVPR. IEEE, 2017
44 Wang P, Li W, Wan J, Ogunbona P, Liu X. Cooperative training of deep aggregation networks for RGB-D action recognition. In: AAAI. 2018
45 Zhu G, Zhang L, Mei L, Shao J, Song J, Shen P. Large-scale isolated gesture recognition using pyramidal 3D convolutional networks. In Pattern Recognition (ICPR) 2016 23rd International Conference. IEEE, 2016
46 Duan J L, Wan J, Zhou S, Guo X Y, Li S Z. A unified framework for multi-modal isolated gesture recognition. ACM Transactions on Multimedia Computing, Communications, and Applications, 2018, 14(1s): 1−16
DOI:10.1145/3131343
47 Li Y, Miao Q, Tian K, Fan Y, Xu X, Li R, Song J. Large-scale gesture recognition with a fusion of RGB-D data based on the C3D model. In 2016 23rd International Conference on Pattern Recognition (ICPR). 2016
48 Hu T K, Lin Y Y, Hsiu P C. Learning adaptive hidden layers for mobile gesture recognition. In: AAAI. 2018
49 Zhang L, Zhu G, Shen P, Song J, Shah S A. Bennamoun M. Learning spatiotemporal features using 3DCNN and convolutional lstm for gesture recognition. In: ICCV. 2017
50 Ohn-Bar E, Trivedi M M. Hand gesture recognition in real time for automotive interfaces: a multimodal vision-based approach and evaluations. IEEE Transactions on Intelligent Transportation Systems, 2014, 15(6): 2368−2377
DOI:10.1109/tits.2014.2337331
51 Carreira J, Zisserman A. Quo vadis, action recognition? a new model and the kinetics dataset. In: CVPR. IEEE, 2017