

RESEARCH

Open Access



Gated spatio and temporal convolutional neural network for activity recognition: towards gated multimodal deep learning

Novanto Yudistira^{1*} and Takio Kurita²

Abstract

Human activity recognition requires both visual and temporal cues, making it challenging to integrate these important modalities. The usual schemes for integration are averaging and fixing the weights of both features for all samples. However, how much weight is needed for each sample and modality, is still an open question. A mixture of experts via a gating Convolutional Neural Network (CNN) is one promising architecture for adaptively weighting every sample within a dataset. In this paper, rather than just averaging or using fixed weights, we investigate how a natural associative cortex such as a network integrates expert networks to form a gating CNN scheme. Starting from Red Green Blue color model (RGB) values and optical flows, we show that with proper treatment, the gating CNN scheme works well, indicating future approaches to information integration in future activity recognition.

Keywords: Gated network, CNN, Deep learning, Action recognition

1 Introduction

The video classification task has become an interesting topic in computer vision and pattern recognition because of its dynamic scenes and objects, which vary either spatially or temporally, making it challenging to design suitable and robust handcrafted features. The evolution of convolutional neural networks (CNNs) has led to significant changes in the way features are being learned. For instance, convolutional filters process pixels considering many aspects such as neighboring pixels and the shapes they form. Therefore, deep CNNs produce many parameters, which is advantageous for the classification task, especially for the classification of video. However, a CNN still needs gating to determine which modality should have more weight than the others. For instance, the gating network should be able to a spatial stream's output more heavily than a temporal one if spatial cues are more salient than motion cues, and vice versa.

Video classification using CNN has achieved significant improvement since the use of a collection of still images and ImageNet weights to be fine tuned on two stream

network. In this paper, we implemented the two-stream CNN proposed by Simonyan [11] for human action recognition, which uses spatial and motion streams using the Chainer framework [13]. Space and motion basically complement each other in nature to characterize activity in videos. There is evidence that integrating RGB channels and optical flow as a representation of space and motion respectively overcomes severe overfitting while increasing testing accuracy [11, 18, 22]. However, how to weight each spatial and motion feature remains an open question. A feature weighting mechanism is required to find the optimal solution given a set of solutions. Using a gating scheme enables a network to be better trained to understand under what conditions the weights of the RGB part should be increased and under what conditions the optical flow should be weighted more heavily. Despite its advantages, there is one drawback of running gating scheme; it requires a large amount of CPU/GPU memory because of, in the case of bimodalities, a large architecture of three networks (two expert networks and one gating network). In this research, each expert network is trained independently and the gating network is then trained to weight each modality before integration.

*Correspondence: cbasemaster@gmail.com

¹Graduate School of Information Engineering, Hiroshima University, Higashi Hiroshima, Japan

Full list of author information is available at the end of the article

The gating network scheme is primarily the same as the mixture expert scheme. It is basically inspired by the associative cortex of the brain, which can handle information integration from many sources. Based on [14], it is evident that the presence of the associative cortex is needed to improve the perception of the environment by the brain. This conclusion is drawn from a study of cats with a deactivated corticocollicular system, where it was found that the ability to integrate target neurons in the superior colliculus is disrupted. Correspondingly, our gating CNN scheme follows the natural corticocolliculus to improve perceptions. Therefore, the main contribution of the gating CNN scheme is to select local patterns that best describes a decision. Because of the high number of degrees of freedom of scenes inside videos, spatial information alone is not enough to describe the target classification, which is sometimes disrupted from one scene to another. Information from one source might be not enough for a CNN to classify the video, regardless of millions of parameters, which tend to lead to overfitting. There are three possibilities to overcoming this problem: adding a larger variety of inputs, increasing training data, or gaining help from another source. When multi-source information is considered as input, normalization is required to make their spaces comparable. For instance, if all frames from one modality are at fixed scales, another source such as motion must be at a fixed scale of the same size to enable the network to perform better with respect to perception. Whenever the output of softmax cross-entropy is retained from each expert stream, the gating network's output weights both experts' output (the output dimensions of the gating network are two when only two expert networks are used).

The success of CNNs has led to a new trend in activity recognition research. Video activity recognition is basically formed by a set of images for which CNNs have demonstrated superior classification. Recently, large image datasets such as ImageNet have been used to enrich the network with the aim of improving the accuracy of image-based classification tasks. However, the incorporation of other sources of information is needed to further improve perceptual accuracy. Simonyan et al. [11] proposed a two-stream CNN that use spatial and temporal cues and performs simple fusion by averaging and using a support vector machine (SVM). Moreover, Wang et al. [12] improved the method of training the two streams by segmental sampling and used predefined fixed weights for the final feature fusion. Many fusion methods have been proposed, for instance, late fusion using a loss function [22] or feature amplification-multiplication [18]. However, we assume that independent streams and loss are more natural because each stream has more freedom to learn depending on its specific task. This motivates us to propose an independent gating CNN architecture.

To summarize, the main contributions of this work are as follows:

- 1) We propose a framework for a gating scheme that is more accurate than if we use only one expert network or merely predefine fixed weights for many expert network outputs.
- 2) We propose our method using two deep models: expert and gating networks with independent loss functions and adaptively weighted outputs of every sample.

2 Related work

Previous studies based on still images have significantly contributed to human activity recognition, such as the two-stream CNN approach used by Simonyan et al. [11], who proposed a very deep network for image recognition [10]. Their proposed method was extended to a temporal segment network [12], which segments the whole video sequence and trains each segment based on its respective network, achieving higher accuracy. However, how to fuse or integrate all streams is still an open question. Before deeply learned features became popular, there were many research approaches to video classification using various methods, especially handcrafted methods such as spatiotemporal features [1], dense trajectories [9], and local autocorrelation [19]. Three-dimensional (3D) CNN was the first attempt to train spatiotemporal features for video classification using deep CNNs. However, it had an overfitting problem due to the lack of available training videos [8]. Later, a YouTube video dataset was provided and late fusion and early fusion for 3D CNN were introduced. Slow features can be learned using deep learning, which is advantageous for action recognition [2]; however, the effectiveness of deep learning over handcrafted systems is still not evident. A breakthrough was proposed with a two-stream network that uses spatial and motion streams and fuses them by simple averaging and SVM fusion. Furthermore, it gains complementary information, which in turn improves accuracy. This approach adopts transfer learning from the large-scale ImageNet dataset and inherits the characteristics of image classification for video action recognition. Time series information was considered by [4, 7] in a long short-term memory network, which is basically a gated version of a recurrent neural network.

A multiplicative gating scheme has been introduced by previous researchers for object detection, language modeling, people re-identification [3], or video classification. Gated object detection was introduced by Xingyu et al. [16] to make use of visual cues of different scales and resolutions. A gated CNN for language modeling was presented by Yann et al. [17], who proposed a gating mechanism that outperforms long short-term memory-based

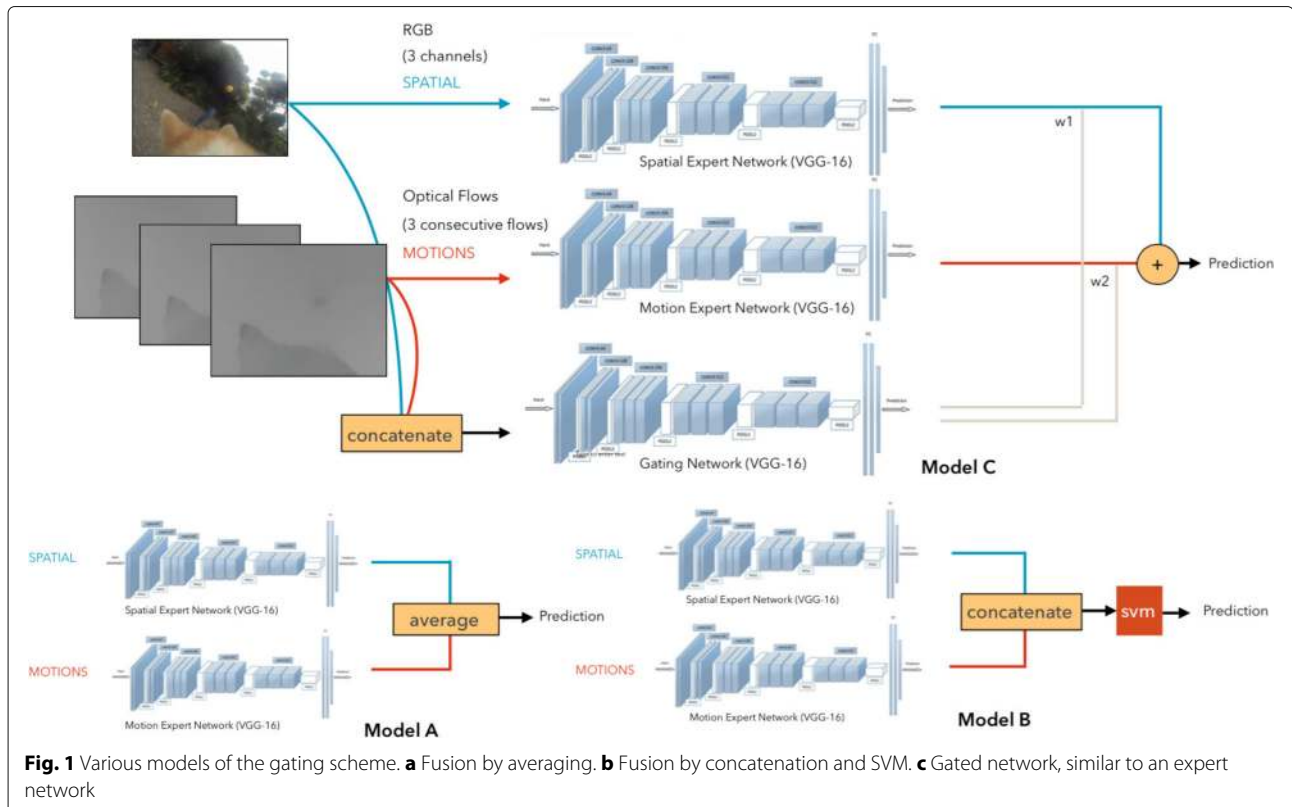
gating. There is one multiplicative gating scheme for video classification [18]. It introduces feature amplification to perform soft gating on intermediate feature maps, which is a different approach to our work, which uses an additional gating network instead. Recently, weighted image segmentation for scene geometry and semantics has been an issue in deep learning applications [15]. If we consider adding one gating network for weighting, it is necessary to calibrate measurements because the gating network itself is for predicting uncertainties. We consider how to manually define learning rate parameters to stabilize expert networks. How to provide an adaptive learning rate such as ESGD [17] remains an open issue. A natural gating network is able to learn non-linearities such as natural transformations [5] for weighting the expert streams. Hadsell et al. [22] proposed a fusion scheme for both RGB and optic flow streams in various layer position and trained it as a model using one loss function. Our approach is different from this in that we use a separate loss for the RGB, flow, and gating streams, which are independently trained in a sequential way. The gating output is trained to weigh both the last layer of the RGB and flow before fusion and classification.

3 Methods

A very deep gating network is introduced to handle the noise and occlusion in a scene for activity recognition.

The proposed gating architecture can be adapted to different contexts depending on the purpose, i.e., a gating network for integrating the audio, text, images, and objects of various spatial resolutions, or actions with various temporal segments. This enables the lower layers of the network to learn parameters with discriminative power. Furthermore, to the best of our knowledge, despite its simplicity, the proposed approach is the first natural gating CNN to be introduced in video classification. We use a gating network that is similar to or shallower than the expert networks. For example, if the gating network is VGG-16, it means that both expert networks are also VGG-16, ResNet-50, or a simple classifier for simplicity.

The use of deep neural networks does not necessarily mean that a specific model or size of CNN must be used; however, VGG-16 and Residual Net (ResNet) [6] have become popular and achieved state-of-the-art results on image classification [10, 24]. Thus, in addition to VGG-16, we use another popular network called ResNet-50. Figure 1 summarizes the models, which consist of fusion by averaging, fusion by SVM, and a gating network. The reason for using various models is to compare possible fusion schemes, including our gating CNN. Even though the gating network model is similar to that of the experts, it is different in terms of output dimensions. The dimensionality of the gating output is two, one for weighting



the spatial expert and the other for weighting the motion expert.

In VGG-16, while making the network deeper, the convolution filter size is smaller, which allows coarse to fine image patterns to be captured. For every output layer, the non-linear activation function of a rectified linear unit is used because it has shown better convergence properties and performance gains with little risk of overfitting. Other network models such as ResNet or Inception could be chosen and possibly achieve higher accuracy while saving memory. However, for the training gating scheme, VGG-16 and resNet-50 are suitable as a starting point.

4 Gated bi-modal CNN design

We briefly introduce the gated CNN in Section 4.1, describing the pipeline of a gated bi-model CNN. Section 4.2 presents the general framework of the gated CNN, and Section 4.3 explains the training and testing scheme. Section 4.4 considers various combination of gating architecture.

4.1 Expert-gating pipeline

A training gating network can be implemented in two ways, by training in parallel both the experts and gating networks or sequential learning by training the experts first and then training the gating network. To train the gating network and expert networks at the same time, careful initial parameter setting is required. For example, we must ensure that during training, the spatial expert network and motion expert network do not exceed each other in terms of accuracy so that the gating network can learn from the true prediction data sufficiently. Specifically, we use a learning rate of 0.000001 for the spatial stream because it tends to converge significantly faster than the motion stream. This is due to the higher number of matches between RGB frames with the pre-trained data (ImageNet). For the motion stream, we use a learning rate of 0.0001 because this combination of parameters is sufficient to stabilize the procedure so that the gating stream is able to train enough data. However, to tune this type of learning for both the expert and gating streams is trivial, and the result is somewhat suboptimal. For instance, if the spatial expert network achieves a 10% increase in accuracy compared with that of the motion expert network, it means the learning rate of the spatial stream must be slowed to balance the gating scheme. Rather than perform this type of learning, we consider splitting the data to train expert network first followed by gating network and continue learning after gating is trained. This process can be summarized by the following pipeline:

- 1 Random video frames are selected; thus, every iteration is given a different input frame. RGB frames are inputted into the spatial network while flows are

inputted into the temporal network. The gating stream is inputted with a concatenation of RGB and flows for the sake of competitiveness between both modalities.

- 2 Given input modalities, each expert is trained independently until it converges.
- 3 The gating network is trained until the loss is stagnant.
- 4 On testing, the gating output weights each expert's output and fused both weighted outputs. Then, classification is performed by selecting the maximum value within a dimension as the predicted label.

4.2 Framework overview

The input of the gating network is concatenation of the spatial and motion information. Each stream has its own loss function that is updated independently, as shown in Fig. 2. The gating mechanisms such as the input gates and output gates follow this equation:

$$y_{final} = x_1y_1 + x_2y_2 \quad (1)$$

where:

$$y_1 + y_2 = 1 \quad (2)$$

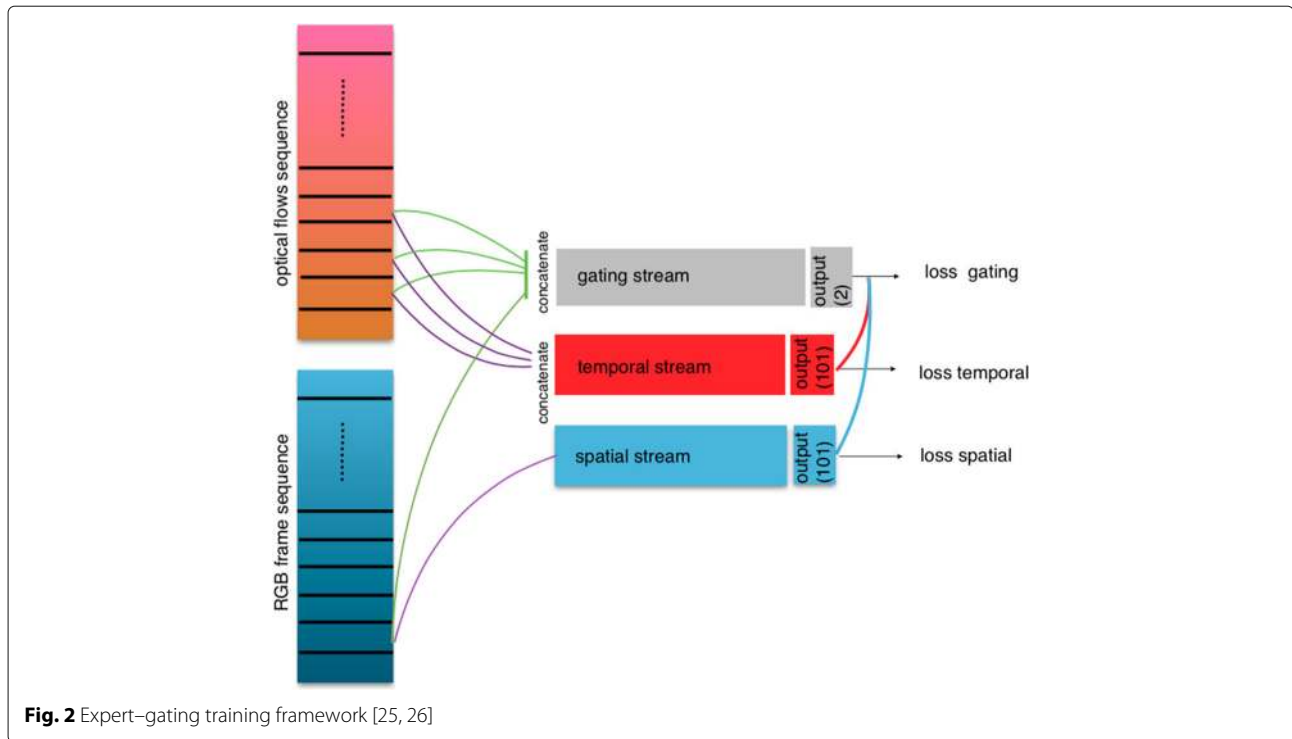
where x_1 , x_2 , $y = (y_1, y_2)$, and y_{final} are the outputs of the RGB stream, optical flow stream, gating stream, and final prediction, respectively. This fusion scheme is presented in Fig. 1 as model C and in Fig. 2 in detail. The output gate is an additional fully connected layer with 101 inputs and two output dimensions. This structure is considered because, in nature, VGG output is 101-dimensional for UCf-101 and 51 for the HMDB-51 dataset (trained on ImageNet with 1000 classes). The final fusion of the output of the expert streams is then normalized using a softmax cross entropy function. Furthermore, for the output of the gating stream, a softmax function is used to transform every feature vector's element as a float between 0 to 1 while the sum of a y_1 and y_2 is 1.

4.3 Input, training-testing scheme, and loss function

Learning consists of two parts: expert learning and gating learning. To train the gating network, experts must be trained and produce feature vectors so that the gating CNN can estimate the proportion of each network relative to the other.

4.3.1 Dataset

We split the training dataset into half: the first half is for training the expert networks and the other half is for training the gating network. However, the whole dataset is used to train the expert network once the gating networks have been trained.



4.3.2 Input and data augmentation

The frame selection for each iteration is randomized. Hence, for every iteration, the method selects a different frame for the same video, thus training on all the frames as it iterates. Three networks are used for this gating CNN scheme; hence, there are three inputs: RGB for the spatial expert network, optical flow for the motion expert network, and a concatenation of RGB and optical flow for the gating network. In this case, RGB contains three channels and the optical flow contains three consecutive flow fields over time with two flow field differences. Therefore, for the gating network input, there are six channels for the first layer of convolution. The optical flow representation is basically transformed into a gray-scale image; thus, three consecutive flows give the same amount of input as the RGB. To overcome overfitting, various pre-processing schemes such as cropping and flipping were performed. We used four-corner cropping and center cropping along with flipping. All the inputs were resized to a resolution of 250×250 with an arbitrary cropping of size 224×224 along with a horizontal flip. A mean image size of 250×250 was computed for the training set and used to subtract all the images.

4.3.3 Training the expert CNNs

For the spatial stream, pre-trained ImageNet was used to reduce overfitting. This kind of transfer learning has improved accuracy by a large margin. For the motion

stream, network was trained from scratch because optical flow features are clear enough to define action, in contrast to spatial scene information. Whether the pre-trained ImageNet model or an untrained model is used initially, the effect on test accuracy and overfitting is still the same for the motion stream. For VGG-16 and ResNet-50, we used a learning rate of 0.001 for the spatial streams. It decreases to 9/10 of its value every 5000 iterations with a momentum of 0.9. The maximum number of iteration was set as 20,000. For the temporal streams, we set a smaller initial learning rate (0.0001) in our experiments. It decreases to 9/10 of its value for every 20,000 iterations and uses momentum of 0.9. The maximum number of iterations was set as 100,000. We also consider transferring the weight of trained expert streams for VGG-16 using the good practice approach from [23] and used [12] for the temporal segment network to be gated with our trained VGG-16 gating network. Note that our trained VGG-16 uses the Caffe framework.

4.3.4 Training the gating CNN

For the gating network, we initialized network weights with pre-trained models from ImageNet. Next, we trained using a learning rate of 0.001, which decreases to 1/10 of its value every 20,000 iterations. Based on experience, if we set the learning rate to a large value (e.g., 0.1), the network tended to choose one of the expert streams, which is not desirable. Training a very deep network such as

VGG-16 is computationally heavy and it takes a long time to converge. Training a simple classifier uses a learning rate of 0.001, reducing by 10% every 5000 iterations. The maximum number of iterations was set as 100,000.

4.3.5 Testing the gating CNN

For given video sequence, we sampled 25 frames equally spaced and fed every frame to its respective stream (three-channel RGB to the spatial stream and three consecutive flow fields to the motion stream) and paired RGB and optical flow into the gating stream. Each of 25 softmax output pairs were then weighted and averaged to predict class.

4.3.6 Testing the two good-practice streams

For a given video sequence, we sampled 25 equally spaced frames and fed every frame to its respective stream and paired RGB and optical flow into the gating stream. The gating output weighted all 750 softmax cross entropy outputs and then averaged to predict the classes. For every frame in the spatial sequence, there were five crops (four corners and one center) with horizontal flips; thus, 10 images were generated for every frame and $25 \times 10 = 250$ were generated for every sequence. Optical flow only formed the center of 10 stacks of three consecutive flow fields for 25 images in a sequence multiplied by two, thus generating $50 \times 10 = 500$ images in total.

4.3.7 Testing the temporal segment network

For given sequence of video, we sampled 25 equally spaced frames and fed every frame to its respective stream and paired RGB and optical flow into the gating stream. All 25 softmax outputs were then averaged to predict class. For every frame, for the spatial sequence, there were five crops (four corners and one center) with flipping; thus, the number of images for every frame was 10. Optical flow only formed the center of 10 stacks for 25 sequences, thus their total was $25 \times 10 = 250$.

4.3.8 Loss function

We used a separate loss function for the expert and gating networks. However, both have basically the same loss function, which minimizes the error of the predefined labels. For the gating network, backpropagation tried to minimize the loss of the gated feature vector using the following loss function:

$$E = - \sum_i t_i \log o_i \tag{3}$$

where o is the softmax cross entropy of output network v :

$$o = \text{softmax}(v) \tag{4}$$

The gradients with respect to the feature vectors at the last layer were computed from the contrastive loss function and backpropagated to the lower layers of the network. Once all the gradients were computed at all layers, we used minibatch stochastic gradient descent to update the parameters of the network.

4.4 Various expert-gating CNN combinations

The base of the expert network can be either two streams of VGG-16 or two streams of ResNet-50 with its gating. A gating CNN also has many possibilities; however, to keep pace with the expert networks, the gating stream should have the same capability as the expert stream. The architecture of the gating itself is still an open question; however, a combination of deep and shallow networks (a simple classifier) can reveal its drawbacks and strengths. Therefore, we prepared several scenarios for expert-gating combinations. VGG-16 has 16 layers while ResNet-50 has 50 layers. We assume that deeper network will increase the number of degrees of freedom, which distracts the network from reaching the optimum solution. As shown in Fig. 3a, VGG-16 streams can be attached using a ResNet-50 or VGG-16. Figure 3b shows that ResNet-50 streams are gated with ResNet-50 or VGG-16. Figure 3c shows that ResNet-50 streams are weighted by a simple classifier with

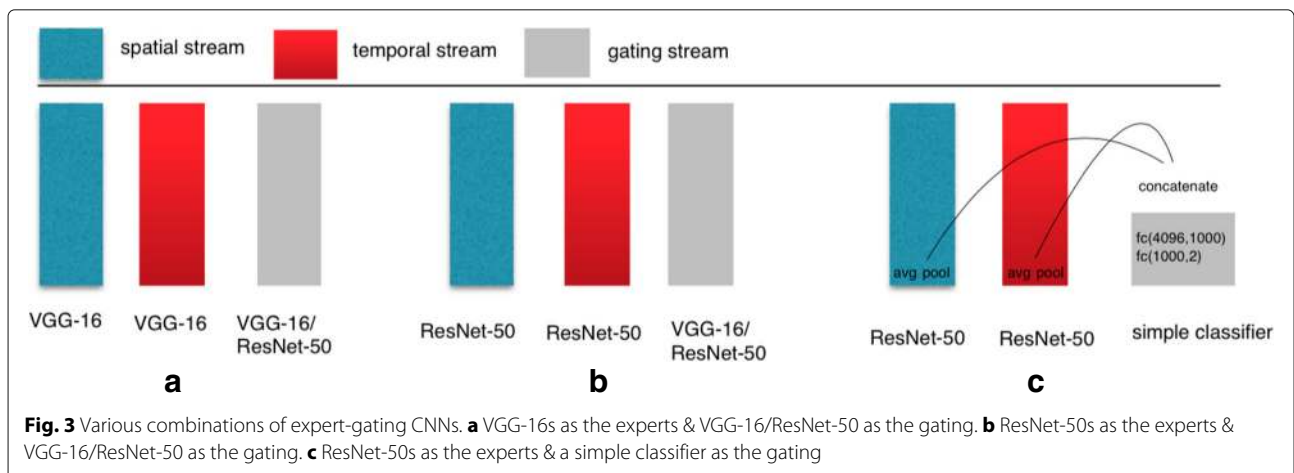


Fig. 3 Various combinations of expert-gating CNNs. **a** VGG-16s as the experts & VGG-16/ResNet-50 as the gating. **b** ResNet-50s as the experts & VGG-16/ResNet-50 as the gating. **c** ResNet-50s as the experts & a simple classifier as the gating

an input size of 4096 (the concatenation output of ResNet-50's last layer without the fully connected layer from both experts). The simple classifier consists of two layers with 4096 inputs and 1000 outputs followed by a layer of 1000 inputs and two outputs.

5 Results and discussion

5.1 Datasets and experiment details

Two challenging datasets were used in the evaluation setup: UCF-101 (Fig. 4) and HMDB-51. These are

challenging datasets because they are small in size for deep learning. UCF-101 consists of 13K videos with 180 frames per video on average and 101 classes. HMDB-51 consists of 6.8K videos and 51 classes. For training the gating network, we used UCF-101 dataset split 1 and use that trained model for the entire experiment, which suitably increased accuracy for all cases. The training and testing split scheme is based on the THUMOS13 challenge [21]. For the entire experiment, we only used split 1 for the analysis of our gating network. We used stochastic

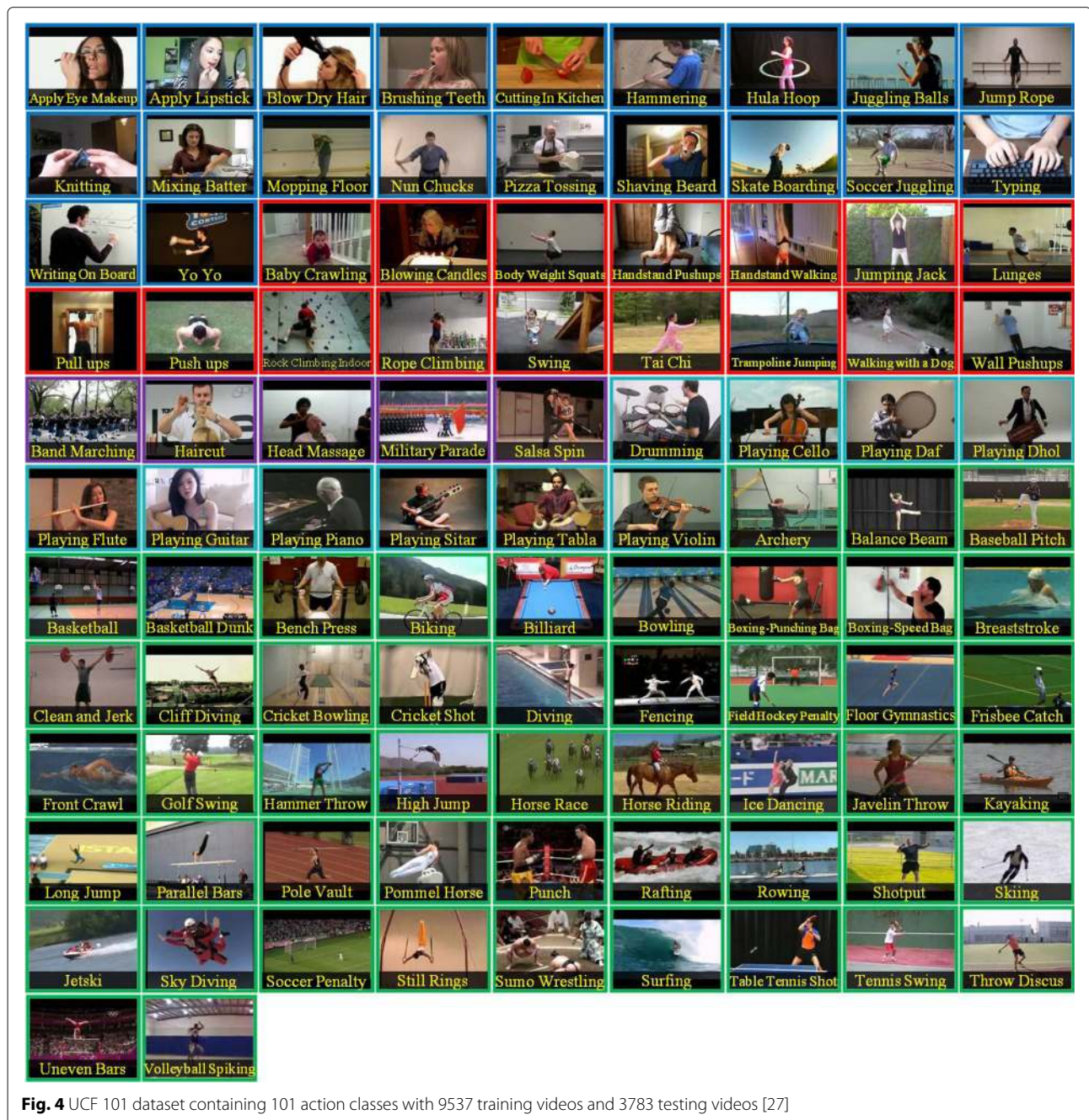


Fig. 4 UCF 101 dataset containing 101 action classes with 9537 training videos and 3783 testing videos [27]

gradient descent as an optimizer for both the experts and gating networks. Due to the limited memory resources of our system, we used minibatch sizes of 12 with a momentum of 0.9. The learning rates were set to 0.0001 and 0.001 for the RGB and flow networks, respectively. To extract the optical flow, we chose the TVL1 optical flow algorithm implemented in OpenCV with CUDA. The whole training time on UCF101 was around 2 h for the spatial network, 18 h for the temporal network, and 6 h for the gating network with a TITAN GPU.

During the training of the VGG-16 experts, after 40 epochs, training was stopped and the gating CNN was trained using the other half of the training dataset and evaluation. After that, training was continued until 80 epochs and an evaluation was run. Next, both experts were trained using the entire training dataset until convergence. We also used an initial parameter copied from the two streams trained using the good practice proposed by [23] (called VGG-16 good practice) and the temporal segment network of [12] for gating with previously trained gating.

5.2 Results

Our gating experiment clearly outperforms the fixed weight scheme. Table 1 shows test accuracy after 40-epoch training. The gating VGG-16 and gating classifier give the best accuracy along with gating classifier in this state at 71.8%. Gating ResNet-50 does not result in the best solution even when the loss starts to converge. The gating network is only trained on this epoch when expert networks training is resumed. Table 2 shows that the gating classifier still outperforms the fixed weights method even after training for 80 epochs. However, in Table 3, after

Table 1 VGG-16, 40 epochs, on UCF-101 (split 1)

RGB weight	Flow weight	Test accuracy
1.0	0.0	46.34%
0.0	1.0	66.80%
0.9	0.1	53.27%
0.8	0.2	58.72%
0.7	0.3	63.60%
0.6	0.4	67.54%
0.5	0.5	70.05%
0.4	0.6	71.48%
0.3	0.7	70.69%
0.2	0.8	69.18%
0.1	0.9	69.18%
Gating VGG-16		71.82%
Gating ResNet-50		67.54%
Gating classifier		71.82%

Table 2 VGG-16, 80 epochs, on UCF-101 (split 1)

RGB weight	Flow weight	Test accuracy
1.0	0.0	65.47%
0.0	1.0	69.66%
0.9	0.1	71.21%
0.8	0.2	72.04%
0.7	0.3	73.55%
0.6	0.4	74.23%
0.5	0.5	76.34%
0.4	0.6	77.01%
0.3	0.7	76.22%
0.2	0.8	73.45%
0.1	0.9	72.32%
Gating VGG-16		75.5%
Gating ResNet-50		74%
Gating classifier		76%

the expert networks converge, only the gating VGG-16 performs better than the fixed weights, while the simple classifier overfits. Meanwhile, ResNet-50 has a high number of degrees of freedom, which stops the gating network from approaching the optimum solution. After the expert networks converge, training achieves an accuracy of nearly 90% for both the spatial and temporal networks while testing yields 72 and 76%, which indicates overfitting. In this situation, the gating network cannot be trained because the training dataset has already become nearly saturated, yielding a large margin between the training and testing accuracy.

Table 3 VGG-16, 300 epochs, (already overfit) on UCF-101 (split 1)

RGB weight	Flow weight	Test accuracy
1.0	0.0	72.45%
0.0	1.0	76.33%
0.9	0.1	79.21%
0.8	0.2	80.23%
0.7	0.3	81.54%
0.6	0.4	82.77%
0.5	0.5	82.81%
0.4	0.6	83.5%
0.3	0.7	82.74%
0.2	0.8	81.22%
0.1	0.9	79.61%
Gating VGG-16		83.5%
Gating ResNet-50		81.24%
Gating classifier		82.10%

Table 4 shows the result obtained by ResNet-50 expert networks with a VGG-16 gating network. The gating classifier also outperforms the fixed weights at 40 epochs. With 80-epoch training, as shown in Table 5, gating the VGG-16 also gives best results for weighting. The gating classifier obtains 77.80% which also exceeds the fixed weights performance. However, after the training is finished and the difference between training and testing accuracy margin becomes greater than 99 to 78.08% for the spatial stream and 90 to 74.89% for the motion stream, a shallower network (the classifier network) overfits the testing data, as shown in Table 6.

We also evaluated gating using a two-stream CNN with its weight transferred from the good practice of [23]. Our gating VGG-16 shows the best accuracy while also approaching the optimum solution if compared with all defined fixed weights on UCF-101 (split 1), as shown in Table 7. For the fixed-weight case, combined weights of 0.4 and 0.6 for the spatial and temporal streams respectively gives the best accuracy. However, the gating CNN still performs better than those with the pre-defined fixed weights.

When weighting the temporal segment network using our trained gating CNN, it obtains the best results and approaches the optimum result when compared with the results of fixed weights, as shown in Table 8 for UCF-101 split 1. The fixed weight of the temporal segment network tends to choose the weight combination of 0.5 and 0.5 for the spatial and temporal streams (average) because it gives the most accurate result. However, our gating network still outperforms the fixed weight method by a margin of 0.24%, confirming the validity of our approach, which weights each sample rather than using fixed weights for

Table 5 ResNet-50, 80 epochs, on UCF-101 (split 1)

RGB weight	Flow weight	Test accuracy
1.0	0.0	70.47%
0.0	1.0	64.82%
0.9	0.1	72.21%
0.8	0.2	74.06%
0.7	0.3	75.49%
0.6	0.4	76.78%
0.5	0.5	77.50%
0.4	0.6	77.11%
0.3	0.7	76.15%
0.2	0.8	73.40%
0.1	0.9	70.55%
Gating VGG-16		78.11%
Gating ResNet-50		75.50%
Gating classifier		77.80%

all samples. We believe that this margin can be better improved with a better gating CNN training protocol in future work. Table 9 shows the result for HMDB-51 on split 1, which shows an improvement compared with the best results of the fixed weight method (0.5 and 0.5 for the spatial and temporal streams, respectively) with a margin of 0.07%. HMDB-51 has fewer training data than those of UCF-101, which is a challenge for training the gating network. As a result, we observe minor improvements in the HMDB-51 results. There is room for improvement using multitask learning.

The results for the UCF-101 and HMDB-51 datasets are given in Tables 10 and 11, respectively. For the expert

Table 4 ResNet-50, 40 epochs, on UCF-101 (split 1)

RGB weight	Flow weight	Test accuracy
1.0	0.0	69.83%
0.0	1.0	63.84%
0.9	0.1	71.60%
0.8	0.2	72.98%
0.7	0.3	74.36%
0.6	0.4	75.76%
0.5	0.5	76.39%
0.4	0.6	76.39%
0.3	0.7	74.94%
0.2	0.8	72.24%
0.1	0.9	69.49%
Gating VGG-16		77.21%
Gating ResNet-50		74.36%
Gating classifier		77.21%

Table 6 ResNet-50, 300 epochs, on UCF-101 (split 1)

RGB weight	Flow weight	Test accuracy
1.0	0.0	78.08%
0.0	1.0	74.89%
0.9	0.1	80.10%
0.8	0.2	82.08%
0.7	0.3	83.64%
0.6	0.4	84.72%
0.5	0.5	86.22%
0.4	0.6	86.25%
0.3	0.7	85.30%
0.2	0.8	82.97%
0.1	0.9	79.61%
Gating VGG-16		86.25%
Gating ResNet-50		83.64%
Gating classifier		85.30%

Table 7 Gated good practice for two streams trained by [12] on UCF-101 (split 1)

RGB weight	Flow weight	Test accuracy
1.0	0.0	79.34%
0.0	1.0	83.60%
0.9	0.1	82.10%
0.8	0.2	84.35%
0.7	0.3	86.47%
0.6	0.4	88.16%
0.5	0.5	89.32%
0.4	0.6	90.02%
0.3	0.7	89.67%
0.2	0.8	88.65%
0.1	0.9	86.73%
Gating		91%

networks that we trained using the Chainer framework [13], the proposed baseline gating scheme outperforms all other models. Note that in this test, we used center cropping to augment the data for both the spatial and temporal streams in this experiment to save computation time. Comparing our proposed models, which are shown in Fig. 1 (model A, model B, and model C), we find that the gated CNN obtains an improvement of 0.3% over averaging fusion (model B) and 1.5% compared with SVM fusion (model C). It also improves both the results for RGB and optical flow alone by 10.2 and 6.5%, respectively. The ResNet-50 expert network (ResNet-50 for the expert network and VGG-16 for the gating network) gives better results in our experiment compared with the VGG-16 expert network with a large margin of 6.1%. These results confirm the value of the mutual information provided by the spatial and motion modalities. It also demonstrates the

Table 8 Gated temporal segment network on UCF-101 (split 1)

RGB weight	Flow weight	Test accuracy
1.0	0.0	85.87%
0.0	1.0	87.89%
0.9	0.1	89.63%
0.8	0.2	91.63%
0.7	0.3	92.98%
0.6	0.4	93.62%
0.5	0.5	93.86%
0.4	0.6	93.66%
0.3	0.7	93.14%
0.2	0.8	91.8%
0.1	0.9	89.98%
Gating		94.10%

Table 9 Gated temporal segment network on HMDB-51 (split 1)

RGB weight	Flow weight	Test accuracy
1.0	0.0	54.31%
0.0	1.0	62.35%
0.9	0.1	59.15%
0.8	0.2	63.46%
0.7	0.3	66.73%
0.6	0.4	68.95%
0.5	0.5	69.93%
0.4	0.6	69.93%
0.3	0.7	68.63%
0.2	0.8	67.45%
0.1	0.9	65.36%
Gating		70%

integration capability of the gating CNN. For HMDB-51, it is found that that the gated CNN is 0.5% better than averaging fusion. It also improves RGB or optical flow results alone by 5 and 12%, respectively. Note that for the temporal stream, we used three consecutive stacked flow fields with two displacements from one flow field to the next.

Table 12 compares our results with those of other fusion methods. Feichtenhofer's fusion method uses late fusion with VGG-16 and VGG-16 with one loss function. With the same VGG-16, RGB itself achieves 82.61% and flow achieves 86.25%, while their fusion achieves 90.62%. Our experiment on the same two streams achieves 91% with RGB results of 79.34% and flow results of 83.60%, which means that, while the two expert networks are actually weaker, our gating network achieves comparable performance. Another fusion method is feature amplification with multiplication. Even without any information about the RGB and flow alone, it achieves 89.1%; our result is slightly better, with a margin of 1.9%.

A comparison with state-of-the-art methods show that gating CNN improves all the expert types, either two-stream VGG-16 or temporal segment networks, as shown in Table 13 for UCF-101 and HMDB-51. We use the weight of trained networks for two-stream networks [23],

Table 10 UCF-101 (split 1)

Methods	Accuracy
Spatial streams (three-channel RGB)	72.7%
Motion streams (three flow fields)	76.5%
SVM Fusion (model B)	81.5%
Averaging (model A)	82.7%
Gating network (model C) VGG-16	83%
Gating network (model C) ResNet-50	88.5%

Table 11 HMDB-51 (split 1)

Methods	Accuracy
Spatial streams (three-channel RGB)	36%
Motion streams (three flow fields)	43%
Averaging (model A)	47.5%
Gating network (model C)	48%
Temporal segment network (averaging) [23]	69.93%
Our gating network (model C) + expert network of temporal segment network [23]	70%

which gives the highest accuracy according to their experiments. The main concern is the comparison with averaging fusion alone and SVM fusion, where the gated two streams achieved better accuracy with a difference of 4.8 and 0.54% for the averaging and SVM fusion, respectively, on UCF-101. When compared with two-stream good practice, as shown in Table 7, the proposed method has better accuracy with a margin of 0.8%.

5.3 Discussion

We have evaluated several gating schemes that basically use deep CNN for weighting. These experiments show that VGG-16 gives the closest-to-optimum solution compared with the deeper network of ResNet-50 and shallower networks. In the middle of training, the simple classifier (two layers with 4096 inputs and 1000 outputs) is robust for approaching the optimum solution; however, as the training converges, there is a shift of variance between the training and testing that the simple classifier does not handle. A deeper network tends to have a high number of degrees of freedom because the number of layers is high. As in ResNet-50, even though the number of parameters is less than those of VGG-16, with deeper layers (50), it fails to approach the optimum solution. Even though residual learning using ResNet-50 tends to benefit from lower number of parameters, they are found to be beneficial for classification instead of for gating. Further work is needed to investigate the ideal model for optimally weighting expert networks.

Table 12 Comparison with another fusion method

Methods	RGB	Flow	Fusion
Feichtenhofer of late fusion - VGG-M-2048 [22]	74.22%	82.34%	85.94
Feichtenhofer of late fusion - VGG-16 [22]	82.61%	86.25%	90.62
Feature amplification + multiplicative [18]	– %	– %	89.1%
Our gating VGG-16 + expert streams of [12]	79.34%	83.60%	91%

Table 13 Comparison with state-of-the-art methods (split 1)

Methods	UCF-101	HMDB-51
Slow fusion spatiotemporal [8]	36%	36%
Improved dense trajectories (IDT) [20]	85.9%	57.2%
Two stream (averaging fusion) [10]	86.2%	–
Two stream (SVM fusion) [10]	87.0%	–
Two stream of good practice [12]	90.2%	–
Our gating stream + good practice of [12] (VGG-16 gating)	91%	–
Temporal segment network [23]	93.86%	69.93%
Our gating stream + temporal segment network of [23] (VGG-16 gating)	94.1%	70%

6 Conclusions

We proposed a baseline gating scheme that is able to weight expert streams for video activity recognition. In this research, a gating CNN was trained to adaptively determine which network stream is more salient compared with the other. To this end, an independent loss function and backpropagation were applied for each expert and gating stream. The outputs from the expert streams are then weighted adaptively by the gating CNN for each sample.

We conducted experiments on the UCF-101 and HMDB-51 datasets using VGG-16 and ResNet-50 to evaluate the ability of deep networks to select the expert for each sample rather than using fixed weights. The results show that state-of-the-art performance is achieved when compared with other fusion methods. However, the gating CNN is burdened by its high number of parameters and degrees of freedom while a simple classifier tends to overfit using the training data. Therefore, further investigation is required to find the ideal structure for the gating CNN and a possible regularization method for overcoming these problems. The gating CNN is potentially useful for the integration of various expert networks such as multimodal, multiresolution, source, or multisegment networks along spatiotemporal space. Thus, rather than dealing with two modalities, an even greater challenge is to determine whether the gating CNN can optimally weight multiple modalities while considering the diversity of the sources.

Acknowledgements

The research is supported by KAKENHI no. 16K00239. We thank Kim Moravec, PhD, from Edanz Group (www.edanzediting.com/ac) for editing a draft of this manuscript.

Funding

The authors declare that KAKENHI no. 16K00239 funded this research and publication.

Availability of data and materials

Not applicable.

Authors' contributions

NY contributed to proposing the gating model, writing code, implementation, evaluation, and writing of the paper. TK contributed by giving the foundation of the gating scheme, supervising, evaluation and grant support. Both authors read and approved the final manuscript.

Authors' information

Novanto Yudistira is currently a PhD student at Hiroshima University. He received his BS in informatics engineering from the Institut Teknologi Sepuluh in November in 2007 and his MS in computer science from Univeristi Teknologi Malaysia in 2011. His current research interests include multimodal feature extraction, deep learning, and computer vision and its applications to machine vision.

Takio Kurita received his B.Eng. degree from the Nagoya Institute of Technology and his Dr.Eng. degree from the University of Tsukuba in 1981 and in 1993, respectively. He joined the Electrotechnical Laboratory, AIST, MITI in 1981. From 1990 to 1991, he was a Visiting Research Scientist at the Institute for Information Technology, National Research Council, Canada. From 2001 to 2009, he was a Deputy Director of the Neuroscience Research Institute, National Institute of Advanced Industrial Science and Technology (AIST). In addition, he was a Professor at the Graduate School of Systems and Information Engineering, University of Tsukuba, from 2002 to 2009. He is currently a Professor at Hiroshima University. His current research interests include statistical pattern recognition and its applications to image recognition. He is a member of the IEEE, IPSJ, IEICE of Japan, Japanese Neural Network Society, and Japanese Society of Artificial Intelligence.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

¹Graduate School of Information Engineering, Hiroshima University, Higashi Hiroshima, Japan. ²Department of Information Engineering, Hiroshima University, Higashi Hiroshima, Japan.

Received: 21 June 2017 Accepted: 27 November 2017

Published online: 15 December 2017

References

- G Somasundaram, et al, Action recognition using global spatio-temporal features derived from sparse representations. *Comput. Vis. Image Underst.* **123**, 1–13 (2014)
- L Sun, K Jia, TH Chan, Y Fang, G Wang, S Yan, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. DL-SFA: deeply-learned slow feature analysis for action recognition, (2014), pp. 2625–2632
- E Ahmed, M Jones, TK Marks, *An improved deep learning architecture for person re-identification*, (2015), pp. 3908–3916
- FA Gers, J Schmidhuber, F Cummins, *Learning to forget: Continual prediction with LSTM*. (IET, 1999)
- R Hadsell, S Chopra, Y LeCun, in *Computer vision and pattern recognition, 2006 IEEE computer society conference on*. Dimensionality reduction by learning an invariant mapping, vol. 2 (IEEE, 2006), pp. 1735–1742
- K He, et al, Deep residual learning for image recognition. *CoRR*. **abs/1512.03385** (2015). <http://arxiv.org/abs/1512.03385>
- S Hochreiter, J Schmidhuber, Long short-term memory. *Neural Comput.* **9**(8), 1735–80 (1997)
- A Karpathy, G Toderici, S Shetty, T Leung, R Sukthankar, L Fei-Fei, in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. Large-scale video classification with convolutional neural networks, (2014), pp. 1725–1732
- H Wang, et al, Dense trajectories and motion boundary descriptors for action recognition. *Int. J. Comput. Vis.* **103**.1, 60–79 (2013)
- K Simonyan, et al, Very deep convolutional networks for large-scale image recognition. *CoRR*. **abs/1409.1556** (2014). <http://arxiv.org/abs/1409.1556>
- K Simonyan, A Zisserman, in *Advances in neural information processing systems*. Two-stream convolutional networks for action recognition in videos, (2014), pp. 568–576
- L Wang, Y Xiong, Z Wang, Y Qiao, D Lin, X Tang, L Van Gool, in *European Conference on Computer Vision*. Temporal segment networks: Towards good practices for deep action recognition (Springer International Publishing, 2016), pp. 20–36
- Chainer, A Flexible Framework for Deep Learning (2016). <http://chainer.org>. Accessed 07 Dec 2017
- BE Stein, TR Stanford, BA Rowland, The neural basis of multisensory integration in the midbrain: its organization and maturation. *Hear. Res.* **258**.1, 4–15 (2009)
- A Kendall, Y Gal, R Cipolla, Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics (2017). *arXiv preprint arXiv:1705.07115*
- X Zeng, W Ouyang, B Yang, J Yan, X Wang, in *European Conference on Computer Vision*. Gated bi-directional cnn for object detection (Springer International Publishing, 2016), pp. 354–369
- YN Dauphin, A Fan, M Auli, D Grangier, Language modeling with gated convolutional networks (2016). *arXiv preprint arXiv:1612.08083*
- E Park, X Han, TL Berg, AC Berg, in *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*. Combining multiple sources of knowledge in deep cnns for action recognition (IEEE, 2016), pp. 1–8
- N Yudistira, T Kurita, in *Systems, Man, and Cybernetics (SMC), 2015 IEEE International Conference on*. Multiresolution Local Autocorrelation of Optical Flows over time for Action Recognition (IEEE, 2015), pp. 1930–1935
- H Wang, C Schmid, in *Proceedings of the IEEE international conference on computer vision*. Action recognition with improved trajectories, (2013), pp. 3551–3558
- A Gorban, H Idrees, YG Jiang, AR Zamir, I Laptev, M Shah, R Sukthankar, in *CVPR workshop*. THUMOS challenge: Action recognition with a large number of classes, (2015)
- C Feichtenhofer, A Pinz, A Zisserman, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Convolutional two-stream network fusion for video action recognition, (2016), pp. 1933–1941
- L Wang, Y Xiong, Z Wang, Y Qiao, Towards good practices for very deep two-stream convnets (2015). *arXiv preprint arXiv:1507.02159*
- K He, X Zhang, S Ren, J Sun, in *Proceedings of the IEEE conference on computer vision and pattern recognition*. Deep residual learning for image recognition, (2016), pp. 770–778
- H El Khiyari, H Wechsler, Face recognition across time lapse using convolutional neural networks. *J. Inf. Secur.* **7**(03), 141 (2016)
- Y Iwashita, A Takamine, R Kurazume, MS Ryoo, in *Pattern Recognition (ICPR), 2014 22nd International Conference on*. First-person animal activity recognition from egocentric videos (IEEE, 2014), pp. 4310–4315
- K Soomro, Amir RZ, Mubarak S, UCF 101: A dataset of 101 human actions classes from videos in the wild (2012). *arXiv preprint arXiv:1212.0402*

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com