PAPER

Object-ABN: Learning to Generate Sharp Attention Maps for Action Recognition

Tomoya NITTA[†], **Tsubasa HIRAKAWA**[††], ***Nonmembers***, **Hironobu FUJIYOSHI**[††], ***and*** **Toru TAMAKI**[†], ***Members***

SUMMARY In this paper we propose an extension of the Attention Branch Network (ABN) by using instance segmentation for generating sharper attention maps for action recognition. Methods for visual explanation such as Grad-CAM usually generate blurry maps which are not intuitive for humans to understand, particularly in recognizing actions of people in videos. Our proposed method, Object-ABN, tackles this issue by introducing a new mask loss that makes the generated attention maps close to the instance segmentation result. Further the Prototype Conformity (PC) loss and multiple attention maps are introduced to enhance the sharpness of the maps and improve the performance of classification. Experimental results with UCF101 and SSv2 shows that the generated maps by the proposed method are much clearer qualitatively and quantitatively than those of the original ABN.

key words: *attention map, action recognition, object mask, PC loss, entropy*

1. Introduction

Action recognition [1]–[5] is one of long-standing topics in computer vision and still actively studied thanks to the emergence of deep learning techniques and large datasets. The task is to classify a trimmed video clip (typically several second-long) into pre-defined action categories [6], [7]. It is a basis of other video-related tasks such as temporal action localization [8] which detects the temporal extent of action events in untrimmed videos, and spatio-temporal action localization [9] in which actors are detected in each frame during the action event.

Deep models, not limited to action recognition but also other tasks, are difficult to investigate because of its black-box nature, hence visual explanations have been studied in the field of explainable AI [10]–[15]. These are attempts to generate saliency maps to visualize which of the parts in the scene are important for classification, and many methods have been utilized, including Grad-CAM [16], Score-CAM [17], LRP [18], and ABN [19]. A common problem of these approaches is that generated maps are often blurry and ambiguous, hence difficult to understand and interpret for human observers [10], [20]. Recent studies have therefore been attempting to improve the quality and sharpness of the maps by focusing on objects, for example, by combining Grad-CAM and LRP [21], applying LRP to Vision Transformers [22], improving ABN with Score-CAM [23], and even human intervention [24] or additional supervision [25].

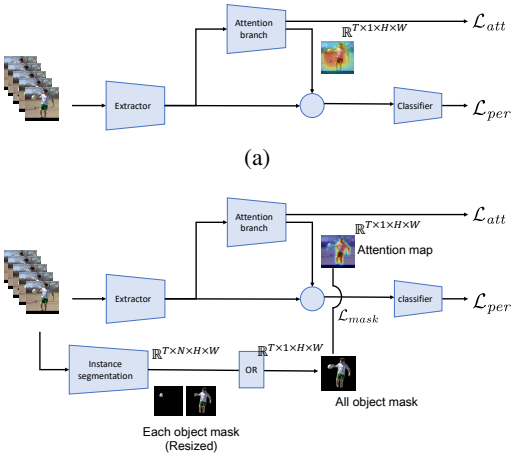


Fig. 1: Overview of models of (a) ABN for action recognition, and (b) the proposed Object-ABN.

For action recognition, this blurry map issue still remains while many visual explanation methods tailored for videos have been proposed. The challenge is to make the model focus on the regions of people who perform the actions in the scene. To this end, some works evaluate visualization results by checking the peak of the map being inside bounding boxes of humans (called pointing games) [26]–[30]. However, this is not a direct approach to the blurry map issue, and models still suffer from the representation bias; models may use clues of backgrounds of the scene instead of the foreground [1], [31], [32].

In this paper, we propose *Object-ABN*, a direct and simple approach to the blurry map issue of action recognition. The key idea is to combine an off-the-shelf instance segmentation model with Attention Branch Network (ABN) [19] (Fig.1(a)). ABN is a classification model with attention and perception branches; the attention branch generates an attention map and uses it as weights of the feature map fed to the perception branch. The attention branch has its own classifier to improve the predictive power of the attention map. In the proposed method (Fig.1(b)), a constraint is added so that the attention map is close to the instance segmentation result. This is expected to lead to a better explanability with a sharper map focusing on people and objects in the scene. However, a possible drawback is the performance-explainability trade-off when adding explana-

Manuscript received July 31, 2022.
Manuscript revised July 31, 2022.
[†]The author is with the Nagoya Institute of Technology, Gokisocho, Showa, Nagoya 466-8555, Japan.
[††]The author is with the Chubu University, 1200 Matsumotocho, Kasugai, Aichi 487-0027, Japan.
DOI: 10.1587/trans.E0.??.1

tion modules to a model [10], [11], [14], [15]. To mitigate this, we propose to use Prototype Conformity (PC) loss [33] that enforces features to be separated in several clusters. In the following sections, we summarize related works, and we briefly describe ABN, then explain the proposed Object-ABN in detail. Then we show experimental results with two datasets, UCF101 [34] and Something-Something v2 [35], with quantitative evaluation on the sharpness of attention maps.

2. Related works

Explainable AI (XAI) [10]–[15] has become an important topic particularly for deep learning models. Visual explanation (or visual attribution) is a topic of XAI, which is to generate a map (sometimes called saliency map, attention map, or attribution map) that visually indicates where in the image the model is focusing on for classification.

Methods are often categorized into post-hoc and intrinsic [10]–[15]. *Post-hoc* methods are used to analyse a single prediction of a trained model, and the name comes from the fact that the visual explanation is done after the model has been trained and fixed. This category includes well-known methods such as CAM [36], Grad-CAM [16], and LRP [18]. These were originally proposed for images, but can be used for videos as well, so they have been used as a baseline for comparison. In addition, some works proposed post-hoc methods tailored for videos by extending methods for images. For example, DevNet [37] used gradient-based Deep Inside CNN [38] and graph-cut for extracting important regions, EB-RNN [30] is based on Excitation Backprop (EB) [26], [27] for models wit CNN and RNN, and EP-3D and ST-EP [28], [29] extends Extremal Perturbation (EP) [39] for spatio-temporal 3D volumes. LRP/DTD [18] has also been applied to videos [40], [41]. Few methods have been proposed specific for video; saliency tubes [42] proposed an additional module for visualizing spatio-temporal tubes, and class feature pyramids [43] proposed a feature back-propagation of 3D CNN.

The post-hoc approach is useful for investigating the behavior of a given model, particularly sensitively can be visualized by showing maps for each category. However, post-hoc methods based on gradients (Grad-CAM [16]) and back-propagation (LRP [18]) are inherently difficult to generate sharp maps because the class-prediction information flows from the top to the bottom through the network, although these methods were also evaluated for object detection and localization tasks. Therefore some methods have been proposed to provide the sharper map and better localization ability; for example, Relevance-CAM [21] combines LRP and Grad-CAM. Perturbation-based methods [26], [39] suffer from the same problem, as well as a high computation cost for perturbing masks many times for video volumes.

Intrinsic methods has its own mechanism of visual explanation in the model itself. This approach has an advantage that the model is designed to have explanability in the first place [15], and that visual explanation during a training phase would be useful for practitioners to check the model performance qualitatively. Because the explanation mechanism of an intrinsic method is a part of the model, there are a great variety of model architectures. Sharma *et al.*[44], [45] used LSTM to predict the soft attention map of the next frame, which were later extended to video captioning with attention [46]. Attention pooling [47] decomposed a 3D attention map with 2nd order pooling and rank-1 approximation. Interpretable spatio-temporal attention [48] used spatial and temporal attention via ConvLSTM. Recent self-attention mechanisms are also introduced in STA-TSN [49] and GTA [50], as well as Transformer-based video models [3]. Although some of these methods do not aim to visual explanation, the blurry map issue still remains for videos because the ability of temporal modeling, which is useful for classification, may be harmful to capture sharp spatial attention maps. If a model has a good ability to model the temporal information and to generate sharp spatial attention maps, a better spatio-temporal action localization would be possible with the attention mechanism.

In this paper, we focus on ABN [19], an intrinsic method proposed for images. ABN first extracts features, then the attention branch computes an attention map which is multiplied to the feature map, then the perception branch classifies the weighted feature map (see Fig.1(a)). The attention map of ABN is useful as visual explanation because the attention map directly specifies the importance of the feature map that is used for classification. However, sometimes the attention map of ABN differs greatly from the human intuition. To alleviate this problem, a Human-in-the-loop (HITL) framework [51] was proposed to enable human operators to modify the attention map of ABN. This results in a sharper attention map that are easy to interpret by humans, leading to a better explanability through attention visualization. However, human intervention on videos that requires frame-by-frame annotations is costly and impractical. In contrast, our proposed Object-ABN scales to a large amount of video frames because it is trained in an end-to-end manner by introducing instance segmentation as an additional self-supervision.

3. Method

In this section, we describe ABN [19] for action recognition. Although the original ABN was proposed for classifying images, notations are aligned with the the proposed models described below.

3.1 ABN

ABN consists of feature extractor E, attention branch A, and perception branch P. Let input video clip be $x \in \mathbb{R}^{T_{\text{in}} \times 3 \times H_{\text{in}} \times W_{\text{in}}}$, where T_{in} is the number of video frames, $H_{\text{in}}, W_{\text{in}}$ are height and width of the frame. The corresponding ground-truth action label is denoted by $y \in \{0, 1\}^L$, where L is the number of categories.

First, the extractor takes a video clip and output feature maps $h_1 = E(x) \in \mathbb{R}^{T \times C \times H \times W}$, where C is the channel

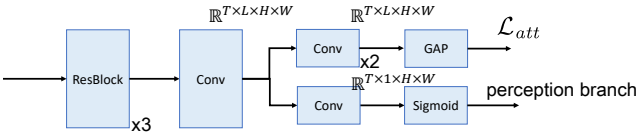


Fig. 2: Attention branch of ABN, which takes feature maps h_1 as input (from the left) then performs convolutions to change the channels to L. There are two internal branches; one generates class prediction y_a for computing loss $\mathcal{L}_{att}$ after Global Average Pooling (GAP), and the other generates attention maps M_u for the perception branch.

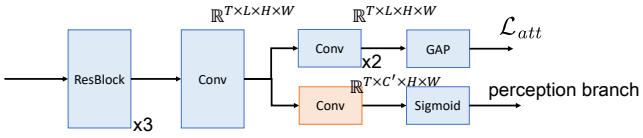


Fig. 3: Attention branch for multiple attention maps, which takes feature maps h_1 and generates class prediction y_a and attention maps $M_{c'}$ in a similar way with the attention branch of the original ABN (Fig. 2). The difference is the number of channels of the output of the convolution layers in the internal branch that generates the multiple attention maps.

size. Then the attention branch takes it, and generates frame-wise (unconstrained) attention maps $M_u \in [0,1]^{T\times 1\times H\times W}$ and class prediction $y_a \in [0,1]^L$ as well (see Figure 2). The maps M_u are applied to feature maps h_1 of each frame separately to generate $h_2 \in \mathbb{R}^{T\times C\times H\times W}$ as

$$h_{2,t,c} = h_{1,t,c} M_{u,t}, \tag{1}$$

for $t = 1,\ldots,T$ and $c = 1,\ldots,C$. The loss attached to the attention branch is $\mathcal{L}_{att} = \mathcal{L}_{CE}(y_a, y)$, where $\mathcal{L}_{CE}$ is a cross entropy loss.

The perception branch P takes weighted feature maps h_2 and outputs prediction $y_p = P(h_2) \in [0,1]^L$. The loss of this branch is $\mathcal{L}_{per} = \mathcal{L}_{CE}(y_p, y)$, and the total loss is

$$\mathcal{L}_{abn} = \mathcal{L}_{per} + \lambda \mathcal{L}_{att}, \tag{2}$$

where λ is a weight.

3.2 Object-ABN

As mentioned before, the attention maps M_u generated by ABN is blurry and often different from areas where people consider important. In this study, we assume that regions of people and objects in the scene are important for identifying action categories, and we enforce on the shape of the attention map being closer to the scene objects. We call this Object-ABN.

To this end, we propose to apply a pre-trained instance segmentation model to each frame of the video clip, which generates the ground-truth object masks $M_{gt} \in \{0,1\}^{T\times N(t)\times H\times W}$ for instances that appear in the video clip. $N(t)$ is the number of instances detected at frame t, so it differs at different frames. We aggregate the object masks to a single channel mask $M'_{gt} \in \{0,1\}^{T\times 1\times H\times W}$ by using logical OR as follows;

$$M'_{gt,t} = \bigcup_{c=1}^{N(t)} M_{gt,t,c}. \tag{3}$$

This is used to compute the following mask loss

$$\mathcal{L}_{mask} = \mathcal{L}_{\mathrm{MSE}}(M_o, M'_{gt}), \tag{4}$$

which is a mean squared error (MSE) loss between the ground-truth object masks and the attention maps M_o. Here we use M_o as the ABN attention maps (instead of M_u) to show that it is constrained by the object masks M'_{gt}.

3.3 Using multiple attention maps

Object-ABN generates the object-constrained attention maps M_o for each frame t, that is, $M_{o,t}$ for $t = 1,\ldots,T$. Each map has a single channel, which means that the same attention map is applied to all C channels of features h_1. However, different channels of h_1 may capture different concepts of the scene, and it might be desirable to use different attention maps for different channels.

Therefore, in this study, we propose to use multiple attention maps by using multi-head attention (MHA). Specifically, we use C' heads to output maps in the attention branch, and each head c' generates attention maps $M_{c'} \in [0,1]^{T\times 1\times H\times W}$ (see Figure 3).

The number of heads C' can be different from the number of channels C, and we align the dimensions as follows. First, we apply the attention map $M_{c'}$ to each channel c of h_1;

$$h^{c'}_{1,t,c} = h_{1,t,c} M_{c',t} \in \mathbb{R}^{T\times C\times H\times W}, \tag{5}$$

then concatenate them in the channel direction;

$$h'_1 = \mathrm{cat}(h^1_1, h^2_1, \ldots, h^{C'}_1) \in \mathbb{R}^{T\times (CC')\times H\times W}, \tag{6}$$

and use a 1×1 convolution

$$h_2 = \mathrm{conv}(h'_1) \in \mathbb{R}^{T\times C\times H\times W}, \tag{7}$$

with the kernel size of $1\times(CC')\times 1\times 1$ to generate h_2 with the appropriate dimension.

In this study, we set $C' = 3$. This means there are three attention maps M_1, M_2, and M_3, and we denote them as M_u, M_o, and M_b, respectively. M_u is the unconstrained attention maps as in the original ABN, and M_o is the object-constrained map M_o of Object-ABN. M_b is the attention maps of the background. In action recognition, it is known that the background can be a clue for classification [1], [31] because of the representation bias [32]. We use two maps for foreground and background by explicitly separating them.

We introduce the following loss for three attention maps;

$$\mathcal{L}_{mha} = \mathcal{L}_{\mathrm{MSE}}(M_o, M'_{gt}) + \lambda_b \mathcal{L}_{\mathrm{MSE}}(M_b, 1 - M'_{gt}), \tag{8}$$

where λ_b is a weight. The first term is the same with the mask loss (4). The second term is for the background attention maps M_b and uses the inverse of the ground-truth object masks. Note that we don't use any losses for M_u, and let the network to obtain the map by itself because the unconstrained maps might be useful like as in the original ABN.

3.4 PC Loss

When creating attention maps, it would be desirable to have features well separated in the middle of the network, particularly in the attention branch, because the attention branch can generated maps suitable for each action categories. To this end, we introduce the Prototype Conformity (PC) loss [33], which encourages cluster to be generated in the latent space and facilitates feature separation. The use of clustered features would be advantageous for generating sharp attention maps while preserving accuracy.

We use the PC loss for features $f \in R^d$ in the attention branch. The loss is represented by

$$\mathcal{L}_{PC} = \lambda_{PC_1} \|f - w_y^c\|_2 - \frac{\lambda_{PC_2}}{K-1} \sum_{j \neq y} \left(\|f - w_j^c\|_2 + \|w_y^c - w_j^c\|_2 \right), \quad (9)$$

where y is the label, K is the number of clusters, and w_j^c is the j-th trainable cluster center. The first term pulls the feature f toward the center w_y^c of the true category y, while the second term pushes the feature away from centers of the all other categories $j \neq y$, as well as separates centers from each other.

The total loss is one of the following;

$$\mathcal{L} = \mathcal{L}_{abn} + \lambda_{mask}\mathcal{L}_{mask} + \lambda_{PC}\mathcal{L}_{PC} \quad (10)$$

$$\mathcal{L} = \mathcal{L}_{abn} + \lambda_{mha}\mathcal{L}_{mha} + \lambda_{PC}\mathcal{L}_{PC}, \quad (11)$$

where λ_{mask}, λ_{mha}, and λ_{PC} are weights.

4. Experiment

4.1 Datasets

We used two datasets in the experiments; UCF101 [34], and something-something v2 (SSv2) [35].

UCF101 [34] has 101 classes of human actions, consisting of a training set of about 9500 videos and a validation set of about 3500 videos. Each video was collected from Youtube, with an average length of 7.21 seconds. We report the performance of the first split.

SSv2 [35] consists of a training set of 168913 videos, a validation set of 24777 videos. Each video is about 2 to 6 second-long (average 4.03 seconds), filmed by a crowd worker. The video contains 173 different templates as action categories, such as "Dropping [something] into [something]" that represents the action performed on objects.

4.2 Experimental setting

Training. From a video in the training set, we sampled 16 frames with a stride of four frames (starting at randomly decided frame) to make an input clip. We resized the shorter side of the frame randomly in the range of 256 to 320 pixels while keeping the aspect ratio, randomly cropped a square of size 224×224 pixels, and then performed the horizontal flip with a probability of 50%. The optimizer used for training was Adam [52] with the learning rate of 10^{-4}, and the number of training epochs was set to 50.

Validation. For UCF101, we used the multi-view test [53] as it is common for evaluating performance of this dataset. From a validation video, one clip was sampled as in training, and this was repeated 10 times to sample 10 clips. Each clip was resized to 224 pixels on its short side while maintaining the aspect ratio, and cropped to 224×224 at the right, center, and left. The results of these 30 clips (views) were averaged to compute a single prediction score. For SSv2, we don't use the multi-view test, instead we sampled one clip is sampled from a video in the validation set as in the training. Then we resized the short side to 256 pixels while maintaining the aspect ratio, then cropped the square patch of size 224×224 pixels in the center of the frame.

Evaluating sharpness. For a quantitative evaluation of the sharpness of attention maps, we propose to use entropy of the maps. If the attention maps are blurry across the entire frames, the distribution of values of the maps becomes broad and the entropy increases. If the attention maps are sharp, the distribution should be polarized toward 0 and 1, entropy should decrease, and the boundary between people and background is expected to be sharper. Therefore, we use the entropy as an indicator of the sharpness the attention maps. However, the entropy decreases when the attention maps are flat and values falls within a certain range. To mitigate this, we normalize the attention map at each frame so that the minimum and maximum values of maps are 0 and 1, respectively.

To compute the entropy, we create a histogram of attention maps with $N = 10$ bins from 0 to 1. The frequency $hist[i]$ of each bin i of the histogram is the normalized discrete probability p_i, which is used to compute the entropy as follows;

$$\text{entropy} = \sum_{i=1}^{N} -p_i \log_2 p_i, \quad p_i = \frac{hist[i]}{\sum_{j=1}^{N} hist[j]}. \quad (12)$$

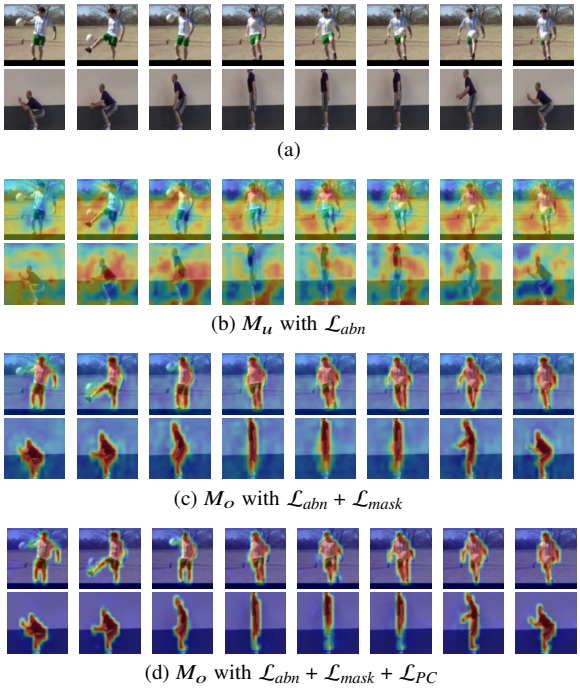
The entropy is calculated for each frame of the video clip, and the entropy of the video is calculated by averaging the entropy of all frames.

As a reference, the maximum of the entropy is achieved when values are $p_i = 1/N$. Since $N = 10$ in our case, $\log_2 10 \approx 3.332$ is the maximum value.

Model. As a backbone of ABN, we used X3D-M [6] pretrained on Kinetics400 [54]. We divided the X3D-M model in two between the third and fourth ResBlocks, using the first half as the feature extractor and the second half as the perception branch. We added an attention branch comprising of three ResBlocks and two conv layers, and the features immediately after the three ResBlocks were used to compute

Table 1: The performances and entropy values with different configurations for the validation set of UCF101.

$\mathcal{L}_{abn}$	$\mathcal{L}_{mask}$	$\mathcal{L}_{mha}$	$\mathcal{L}_{PC}$	top-1	top-5	entropy M_u	entropy M_o	entropy M_b
✓				95.43	99.29	3.065		
✓			✓	96.27	99.68	3.042		
✓	✓			95.29	99.44		2.004	
✓	✓		✓	95.14	99.07	1.453		
✓		✓		90.91	98.44	2.838	1.342	1.339
✓		✓	✓	89.98	97.99	2.793	1.381	1.406
interpretable attention [48]				87.11				
STA-TSN [49] RGB				83.4				
STA-TSN [49] RGB+flow				92.8				



(a)

(b) M_u with $\mathcal{L}_{abn}$

(c) M_o with $\mathcal{L}_{abn}$ + $\mathcal{L}_{mask}$

(d) M_o with $\mathcal{L}_{abn}$ + $\mathcal{L}_{mask}$ + $\mathcal{L}_{PC}$

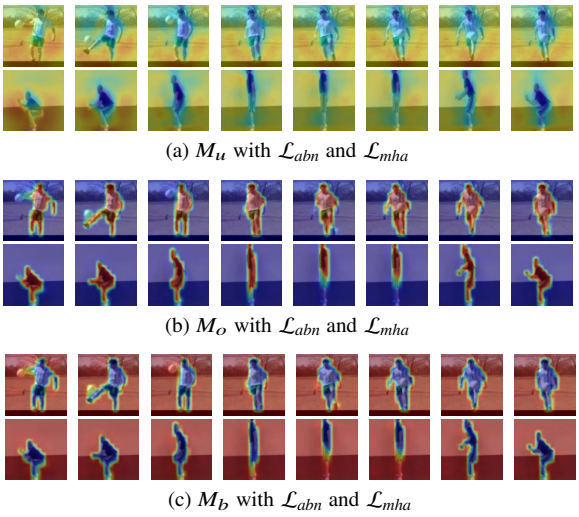
Fig. 4: Visualization of results for the validation set of UCF101. (a) Input videos (every two frames out of 16 frames of a clip are shown). (b) Unconstrained maps M_u with $\mathcal{L}_{abn}$ (original ABN). (c) Object-constrained maps M_o with $\mathcal{L}_{abn}$, and $\mathcal{L}_{mask}$, as well as (d) $\mathcal{L}_{PC}$.

the PC loss.

The resulting model takes an input video clip of size $T_{\text{in}} \times H_{\text{in}} \times W_{\text{in}} = 16 \times 244 \times 244$, and generates an attention map of size $T \times H \times W = 16 \times 14 \times 14$ which is the spatial resolution of the third ResBlock of the X3D-M.

For instance segmentation to obtain object masks M_{gt}, we used Mask R-CNN [55] implemented in Detectron2 [56] trained on the instance segmentation task of the COCO dataset [57] with 80 classes.

Parameters. The parameters used in the experiment were set as follows; $\lambda = 1, \lambda_b = 1, \lambda_{mask} = 10, \lambda_{mha} = 10, \lambda_{PC} = 10^{-4}, \lambda_{PC_1} = 1$, and $\lambda_{PC_2} = 10^{-3}$. The number of clusters K was set to L, the number of categories.



(a) M_u with $\mathcal{L}_{abn}$ and $\mathcal{L}_{mha}$

(b) M_o with $\mathcal{L}_{abn}$ and $\mathcal{L}_{mha}$

(c) M_b with $\mathcal{L}_{abn}$ and $\mathcal{L}_{mha}$

Fig. 5: Visualization of results for the validation set of UCF101 with $\mathcal{L}_{abn}$ and $\mathcal{L}_{mha}$. Three types of maps (a) M_u, (b) M_o, and (c) M_b.

4.3 Experimental results for UCF101

We obtained main results with UCF101, which are shown in Tab.1. Each row shows the performance and entropy with different configurations of losses. The first row with $\mathcal{L}_{abn}$ only is equivalent to the original ABN, and the second rows is the original ABN with the PC loss $\mathcal{L}_{PC}$. In the following rows, results are of Object-ABN with either of $\mathcal{L}_{mask}$ and $\mathcal{L}_{mha}$ is used, and with or without the PC loss.

4.3.1 Mask loss

First, we compare the original ABN with the proposed Object-ABN to verify the effect of the mask loss. As can be seen from the first row ($\mathcal{L}_{abn}$ only) and third row ($\mathcal{L}_{abn}$ and $\mathcal{L}_{mask}$) of Tab.1, the difference in the top-1 performance is 0.14 points and not so large. However, the entropy decreased by more than 1 when the mask loss is used, which means that quantitatively the sharpness of the attention map was drastically improved.

Also, the generated attention maps are completely different. In case of using the mask loss (Fig.4(c)), generated maps M_o are sharp so that objects and people are clearly visible, while the case without the mask loss (Fig.4(b)) produced maps M_u that are blurry and speckled, and action-related foreground and background doesn't appear.

4.3.2 PC Loss

Next, we see how the PC loss affect the performance and the attention maps. As shown in Tab.1, using the PC loss reduces the entropy values and improve the performances for the cases with the original ABN (the first two rows). However, when either of $\mathcal{L}_{mask}$ or $\mathcal{L}_{mha}$ is used, the PC loss seems not to contribute the improvements in this case.

Figures 4(d) shows maps with the PC loss. Compared with Fig.4(c) without the PC loss, the separation of the background and foreground is clearer, and the fluctuations in the background disappear.

4.3.3 Multiple attention maps

Here, we shows the effect of multi-head attention maps. Corresponding results are the last two rows of Tab.1, where the performance dropped by about 5% compared to the cases without $\mathcal{L}_{mha}$, even with the PC loss. Hence, in terms of performance, using a single attention map would be better.

However, the entropy values became smaller and the quality of the maps were further improved by using $\mathcal{L}_{mha}$, as shown in Fig.5. The object-constrained mask M_o (Fig.5(b)) are far more sharper than those with $\mathcal{L}_{mask}$ (Fig.4(c)), and even with $\mathcal{L}_{PC}$ (Fig.4(d)). Note that we can observe the representation bias, stated in the introduction, in unconstrained masks M_u that have higher values in the background than the foreground. In contrast, object-constrained masks M_o clearly capture the figure of the actor.

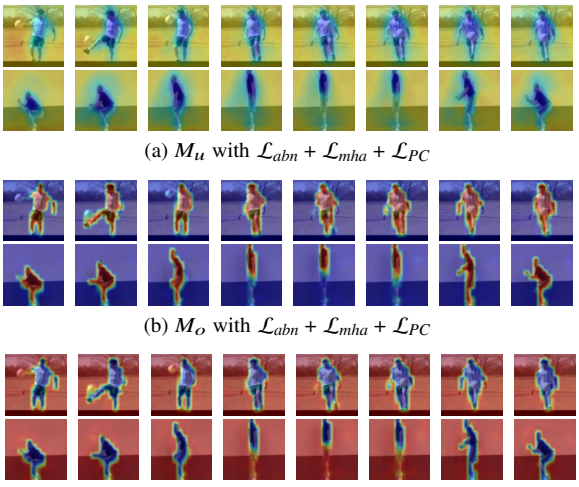
4.3.4 Comparisons with other methods

Table 1 also shows performances of other methods that are intrinsic models for action recognition for the purpose of visualizing attention maps. Of course none of them have published entropy values, however their results are visually much worse than our results in terms of the sharpens of the attention maps. The performance of our method is better when not using $\mathcal{L}_{mha}$, which might be caused by the nature of this dataset. As stated in the introduction, UCF101 [34] and similar datasets (such as Kinetics [54]) have the representation bias [1], [31], [32]. This effect may cause a visually-random attention map (such as 4(b)) to show a better performance. To investigate this issue, in the following, we show another experiment with SSv2 that doesn't have the effect.

4.4 Experimental result for SSv2

Here we show the experimental results for SSv2. The training settings was the same with UCF101, except for the frame sampling stride (two frames instead of four), no horizontal flip, and 25 epochs for training. Performances are shown in Tab.2, and visualization results with the mask loss are shown in Fig.7, and with the multiple attention maps in Fig.8.

Unlike results of UCF101 shown in previous sections, the performance was improved from the original ABN by adding the mask loss, and the entropy of M_o for the Object-ABN is smaller than that of M_u for ABN. Furthermore, using the PC loss and adding the multiple attention maps improve the performance. The object-constrained maps M_o shown in Fig.7(c)(d) and Fig.8(b) look almost the same, supported by the similar entropy values in Tab.2. This suggests that for this dataset the mask loss has the largest impact on the sharpness of the maps, while the multiple attention maps and the PC loss also contribute to the performance.



(a) M_u with $\mathcal{L}_{abn}$ + $\mathcal{L}_{mha}$ + $\mathcal{L}_{PC}$

(b) M_o with $\mathcal{L}_{abn}$ + $\mathcal{L}_{mha}$ + $\mathcal{L}_{PC}$

(c) M_b with $\mathcal{L}_{abn}$ + $\mathcal{L}_{mha}$ + $\mathcal{L}_{PC}$

Fig. 6: Visualization of results for the validation set of UCF101 with $\mathcal{L}_{abn}$, $\mathcal{L}_{mha}$, and $\mathcal{L}_{PC}$. Three types of maps (a) M_u, (b) M_o, and (c) M_b.

Table 2: The performances and entropy values with different configurations for the validation set of SSv2.

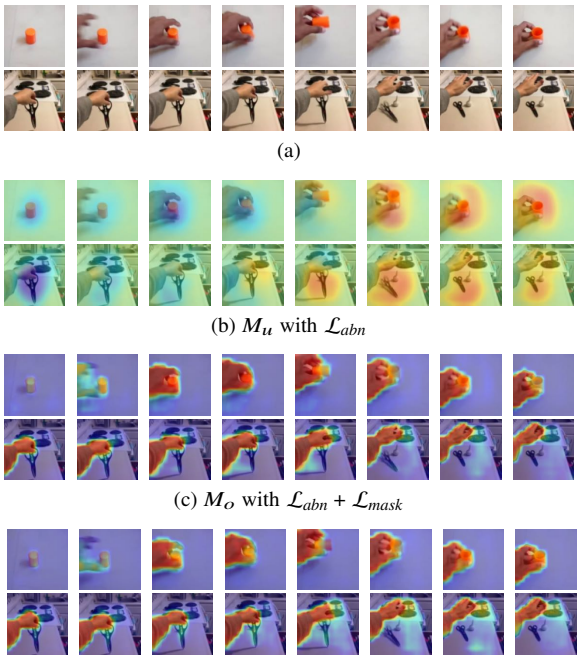
$\mathcal{L}_{abn}$	$\mathcal{L}_{mask}$	$\mathcal{L}_{mha}$	$\mathcal{L}_{PC}$	top-1	top-5	entropy M_u	entropy M_o	entropy M_b
✓				54.63	83.23	2.887		
✓	✓			54.83	82.73		2.331	
✓	✓		✓	54.97	83.15		2.427	
✓		✓	✓	55.05	83.60	2.921	2.236	2.237

The unconstrained maps M_u are shown in Fig.7(b) for ABN and Fig.8(a) for the proposed method. For the maps of ABN, the attentions to the object and hands are weaker (in blue) than to the background at the beginning. After the action starts, the attention is getting focused on the object, then becomes strong at the end. In contrast, the maps of the proposed method are flat at first, then the attention is continuously focused on the object during the action until the end.

Therefore, the maps M_u are expected to better represent the temporal information of the action. In contrast, the maps M_o clearly show the subject of the action (hands, in this case), which represents the spatial information of the action in each frame. Therefore, the use of multiple attention maps is expected to lead to a better understanding of spatio-temporal information of actions.

4.5 Effect of instance segmentation

The proposed method relies heavily on the results of instance segmentation as it is used to compute the mask loss, and restrict the object-constraint maps M_o (and M_b). Figure 9 shows the object masks M_{gt} (before mask aggregation by Eq.(3)) for each sequence shown in the experiments. Figs.9(a)–(c) show segmentation results of each frame for the "person" class (top rows) and all other classes (bottom



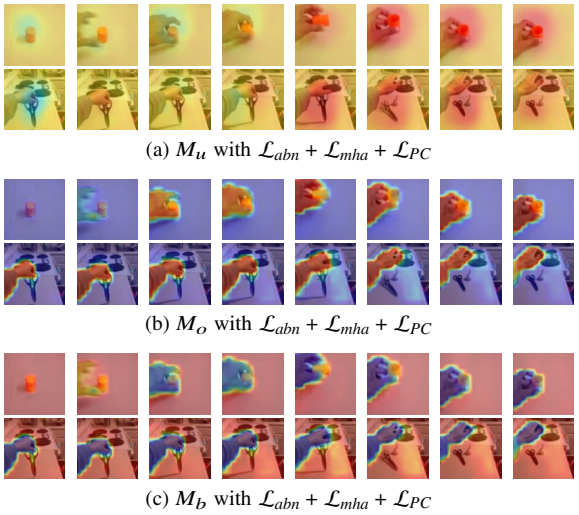
(a)

(b) M_u with $\mathcal{L}_{abn}$

(c) M_o with $\mathcal{L}_{abn} + \mathcal{L}_{mask}$

(d) M_o with $\mathcal{L}_{abn} + \mathcal{L}_{mask} + \mathcal{L}_{PC}$

Fig. 7: Visualization of results for the validation set of SSv2. (a) Input videos. (b) M_u with $\mathcal{L}_{abn}$. (c) M_o with $\mathcal{L}_{abn}$, and $\mathcal{L}_{mask}$, as well as (d) $\mathcal{L}_{PC}$.



(a) M_u with $\mathcal{L}_{abn} + \mathcal{L}_{mha} + \mathcal{L}_{PC}$

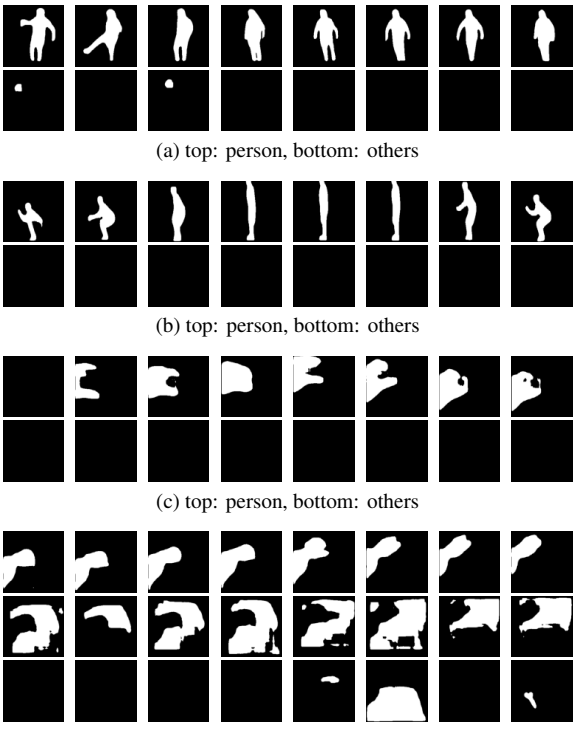
(b) M_o with $\mathcal{L}_{abn} + \mathcal{L}_{mha} + \mathcal{L}_{PC}$

(c) M_b with $\mathcal{L}_{abn} + \mathcal{L}_{mha} + \mathcal{L}_{PC}$

Fig. 8: Visualization of results for the validation set of SSv2 with $\mathcal{L}_{abn}$, $\mathcal{L}_{mha}$, and $\mathcal{L}_{PC}$. Three types of maps (a) M_u, (b) M_o, and (c) M_b.

rows).

In the bottom rows, almost all the frames are black, indicating that there are no other categories, except in Fig.9(a) there are small ball regions (one of the COCO categories) in a few frames. In contrast, object-constrained maps M_o in Figs.4 and 6 have the small ball region only when the ball is near the person (in the second frame). This suggests that the maps M_o, trained with the aggregated instance segmentation masks M'_{gt} including all object categories, respond only in



(a) top: person, bottom: others

(b) top: person, bottom: others

(c) top: person, bottom: others

(d) top: person, middle: oven, bottom: others

Fig. 9: Visualization of object masks M_{gt} obtained by instance segmentation. (a)(b) Masks correspond to the sequences in Fig.4(a), and (c)(d) to the sequences in Fig.7(a).

the regions of the person performing actions (actors), which is the main target of the classification task.

This is clearly observed in Fig.9(d) in which the object masks M_{gt} are separately shown for "person", "oven" (terrible failure of segmentation), and all other classes (including "scissors", "spoon", and "dining table"). Eq.(3) aggregates all regions of these categories into a single mask M'_{gt} to be used for computing the mask loss. However, masks M_o in Figs.7 and 8 focus on the hands only, the subject of the action, with higher values.

This observation suggests that the attention mechanism of the proposed method is robust to the noise or failure of instance segmentation results used during the model training, and able to focusing on the actor in the scene. This is particularly useful for the task of spatio-temporal action localization, and will be left for our future work.

5. Conclusion

In this paper, we have proposed Object-ABN, an extension of ABN by using instance segmentation, and enables the generation of sharper attention maps, which enable us to clearly see which parts of the scene the model is focusing on. Experiments with two datasets demonstrated that the proposed method with the mask loss, multiple attention maps, and the PC loss improves the quality of attention maps in terms of entropy, as well as the classification performances. Our fu-

ture work includes an experimental study of spatio-temporal action localization of the proposed method, and the further investigation how the changes in the segmentation mask affect the performance, and whether the attention maps focus on main actors only or on other people irrelevant to the action.

Acknowledgement

This work was supported in part by JSPS KAKENHI Grant Number JP22K12090.

References

[1] Y. Zhu, X. Li, C. Liu, M. Zolfaghari, Y. Xiong, C. Wu, Z. Zhang, J. Tighe, R. Manmatha, and M. Li, "A comprehensive study of deep video action recognition," CoRR, vol.abs/2012.06567, 2020.

[2] M.S. Hutchinson and V.N. Gadepally, "Video action understanding," IEEE Access, vol.9, pp.134611–134637, 2021.

[3] J. Selva, A.S. Johansen, S. Escalera, K. Nasrollahi, T.B. Moeslund, and A. Clapés, "Video transformers: A survey," CoRR, vol.abs/2201.05991, 2022.

[4] M. Vrigkas, C. Nikou, and I.A. Kakadiaris, "A review of human activity recognition methods," Frontiers in Robotics and AI, vol.2, 2015.

[5] Y. Kong and Y. Fu, "Human action recognition and prediction: A survey," CoRR, vol.abs/1806.11230, 2018.

[6] C. Feichtenhofer, "X3d: Expanding architectures for efficient video recognition," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2020.

[7] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), October 2019.

[8] H. Xia and Y. Zhan, "A survey on temporal action localization," IEEE Access, vol.8, pp.70477–70487, 2020.

[9] C. Gu, C. Sun, D.A. Ross, C. Vondrick, C. Pantofaru, Y. Li, S. Vijayanarasimhan, G. Toderici, S. Ricco, R. Sukthankar, C. Schmid, and J. Malik, "Ava: A video dataset of spatio-temporally localized atomic visual actions," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018.

[10] A. Adadi and M. Berrada, "Peeking inside the black-box: A survey on explainable artificial intelligence (xai)," IEEE Access, vol.6, pp.52138–52160, 2018.

[11] A. Das and P. Rad, "Opportunities and challenges in explainable artificial intelligence (XAI): A survey," CoRR, vol.abs/2006.11371, 2020.

[12] X. Huang, D. Kroening, W. Ruan, J. Sharp, Y. Sun, E. Thamo, M. Wu, and X. Yi, "A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability," Computer Science Review, vol.37, p.100270, 2020.

[13] E. Tjoa and C. Guan, "A survey on explainable artificial intelligence (xai): Toward medical xai," IEEE Transactions on Neural Networks and Learning Systems, vol.32, no.11, pp.4793–4813, 2021.

[14] P. Linardatos, V. Papastefanopoulos, and S. Kotsiantis, "Explainable ai: A review of machine learning interpretability methods," Entropy, vol.23, no.1, 2021.

[15] G. Ras, N. Xie, M. van Gerven, and D. Doran, "Explainable deep learning: A field guide for the uninitiated," Journal of Artificial Intelligence Research, vol.73, p.329–396, feb 2022.

[16] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," Proceedings of the IEEE International Conference on Computer Vision (ICCV), Oct 2017.

[17] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, and X. Hu, "Score-cam: Score-weighted visual explanations for convolutional neural networks," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2020.

[18] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," PLOS ONE, vol.10, no.7, pp.1–46, 07 2015.

[19] H. Fukui, T. Hirakawa, T. Yamashita, and H. Fujiyoshi, "Attention branch network: Learning of attention mechanism for visual explanation," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.

[20] L. Hiley, A.D. Preece, and Y. Hicks, "Explainable deep learning for video recognition tasks: A framework & recommendations," 2019.

[21] J.R. Lee, S. Kim, I. Park, T. Eo, and D. Hwang, "Relevance-cam: Your model already knows where to look," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.14944–14953, June 2021.

[22] H. Chefer, S. Gur, and L. Wolf, "Transformer interpretability beyond attention visualization," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.782–791, June 2021.

[23] K.H. Lee, C. Park, J. Oh, and N. Kwak, "Lfi-cam: Learning feature importance for better visual explanation," Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp.1355–1363, October 2021.

[24] M. Mitsuhara, H. Fukui, Y. Sakashita, T. Ogata, T. Hirakawa, T. Yamasita, and H. Fujiyoshi, "Embedding human knowledge into deep neural network via attention map," International Conference on Computer Vision Theory and Applications (VISAPP), 2021.

[25] K. Li, Z. Wu, K.C. Peng, J. Ernst, and Y. Fu, "Tell me where to look: Guided attention inference network," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018.

[26] J. Zhang, S.A. Bargal, Z. Lin, J. Brandt, X. Shen, and S. Sclaroff, "Top-down neural attention by excitation backprop," International Journal of Computer Vision, vol.126, no.10, pp.1084–1102, 2018.

[27] J. Zhang, Z.L. Lin, J. Brandt, X. Shen, and S. Sclaroff, "Top-down neural attention by excitation backprop," Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part IV, ed. B. Leibe, J. Matas, N. Sebe, and M. Welling, Lecture Notes in Computer Science, vol.9908, pp.543–559, Springer, 2016.

[28] Z. Li, W. Wang, Z. Li, Y. Huang, and Y. Sato, "Towards visually explaining video understanding networks with perturbation," IEEE Winter Conference on Applications of Computer Vision, WACV 2021, Waikoloa, HI, USA, January 3-8, 2021, pp.1119–1128, IEEE, 2021.

[29] Z. Li, W. Wang, Z. Li, Y. Huang, and Y. Sato, "Spatio-temporal perturbations for video attribution," IEEE Transactions on Circuits and Systems for Video Technology, vol.32, no.4, pp.2043–2056, 2022.

[30] S.A. Bargal, A. Zunino, D. Kim, J. Zhang, V. Murino, and S. Sclaroff, "Excitation backprop for rnns," 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, pp.1440–1449, Computer Vision Foundation / IEEE Computer Society, 2018.

[31] Y. He, S. Shirakabe, Y. Satoh, and H. Kataoka, "Human action recognition without human," CoRR, vol.abs/1608.07876, 2016.

[32] Y. Li, Y. Li, and N. Vasconcelos, "Resound: Towards action recognition without representation bias," Proceedings of the European Conference on Computer Vision (ECCV), September 2018.

[33] A. Mustafa, S. Khan, M. Hayat, R. Goecke, J. Shen, and L. Shao, "Adversarial defense by restricting the hidden space of deep neural networks," Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), October 2019.

[34] K. Soomro, A.R. Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," CoRR,

vol.abs/1212.0402, 2012.
[35] R. Goyal, S. Ebrahimi Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fruend, P. Yianilos, M. Mueller-Freitag, F. Hoppe, C. Thurau, I. Bax, and R. Memisevic, "The "something something" video database for learning and evaluating visual common sense," Proceedings of the IEEE International Conference on Computer Vision (ICCV), Oct 2017.
[36] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.
[37] C. Gan, N. Wang, Y. Yang, D. Yeung, and A.G. Hauptmann, "Devnet: A deep event network for multimedia event detection and evidence recounting," IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015, pp.2568–2577, IEEE Computer Society, 2015.
[38] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," Workshop at International Conference on Learning Representations, 2014.
[39] R. Fong, M. Patrick, and A. Vedaldi, "Understanding deep networks via extremal perturbations and smooth masks," Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), October 2019.
[40] C.J. Anders, G. Montavon, W. Samek, and K. Müller, "Understanding patch-based learning of video data by explaining predictions," in Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, ed. W. Samek, G. Montavon, A. Vedaldi, L.K. Hansen, and K. Müller, Lecture Notes in Computer Science, vol.11700, pp.297–309, Springer, 2019.
[41] L. Hiley, A.D. Preece, Y. Hicks, A.D. Marshall, and H. Taylor, "Discriminating spatial and temporal relevance in deep taylor decompositions for explainable activity recognition," CoRR, vol.abs/1908.01536, 2019.
[42] A. Stergiou, G. Kapidis, G. Kalliatakis, C. Chrysoulas, R.C. Veltkamp, and R. Poppe, "Saliency tubes: Visual explanations for spatio-temporal convolutions," 2019 IEEE International Conference on Image Processing, ICIP 2019, Taipei, Taiwan, September 22-25, 2019, pp.1830–1834, IEEE, 2019.
[43] A. Stergiou, G. Kapidis, G. Kalliatakis, C. Chrysoulas, R. Poppe, and R.C. Veltkamp, "Class feature pyramids for video explanation," 2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019, pp.4255–4264, IEEE, 2019.
[44] S. Sharma, R. Kiros, and R. Salakhutdinov, "Action recognition using visual attention," CoRR, vol.abs/1511.04119, 2015.
[45] S. Sharma, R. Kiros, and R. Salakhutdinov, "Action recognition using visual attention," International Conference on Learning Representations (ICLR) Workshop, May 2016.
[46] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," Proceedings of the 32nd International Conference on Machine Learning, ed. F. Bach and D. Blei, Proceedings of Machine Learning Research, vol.37, Lille, France, pp.2048–2057, PMLR, 07–09 Jul 2015.
[47] R. Girdhar and D. Ramanan, "Attentional pooling for action recognition," Advances in Neural Information Processing Systems, ed. I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Curran Associates, Inc., 2017.
[48] L. Meng, B. Zhao, B. Chang, G. Huang, W. Sun, F. Tung, and L. Sigal, "Interpretable spatio-temporal attention for video action recognition," Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, Oct 2019.
[49] G. Yang, Y. Yang, Z. Lu, J. Yang, D. Liu, C. Zhou, and Z. Fan, "Sta-tsn: Spatial-temporal attention temporal segment network for action recognition in video," PLOS ONE, vol.17, no.3, pp.1–19, 03 2022.
[50] B. He, X. Yang, Z. Wu, H. Chen, S.N. Lim, and A. Shrivastava, "Gta: Global temporal attention for video action understanding," Proceedings of the British Machine Vision Conference (BMVC)), November 2021.
[51] T. Iwayoshi, M. Mitsuhara, M. Takada, T. Hirakawa, T. Yamashita, and H. Fujiyoshi, "Attention mining branch for optimizing attention map," 17th International Conference on Machine Vision and Applications (MVA), 2021.
[52] D.P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, ed. Y. Bengio and Y. LeCun, 2015.
[53] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018.
[54] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, "The kinetics human action video dataset," 2017.
[55] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask r-cnn," Proceedings of the IEEE International Conference on Computer Vision (ICCV), Oct 2017.
[56] Y. Wu, A. Kirillov, F. Massa, W.Y. Lo, and R. Girshick, "Detectron2." `https://github.com/facebookresearch/detectron2`, 2019.
[57] T. Lin, M. Maire, S.J. Belongie, L.D. Bourdev, R.B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C.L. Zitnick, "Microsoft COCO: common objects in context," CoRR, vol.abs/1405.0312, 2014.

Tomoya Nitta received B.E. from Nagoya Institute of Technology in 2022.His research interests include computer vision and action recognition.

Tsubasa Hirakawa received his PhD degree from Department of Computer Science, Hiroshima University, Japan in 2017. From 2017 to 2019 he was a researcher fellow at the Chubu University. He is a specially appointed associate professor of the Chubu Institute for Advanced Studies, Chubu University, Japan from 2019 to 2021. He is now a lecturer of the Center for Mathematical Science and Artificial Intelligence, Chubu University, Japan. He was a Fellowship of the Japan Society for the Promotion of Science from 2014 to 2017. He was a visiting researcher at ESIEE Paris, France, in 2014 and 2015.

Hironobu Fujiyoshi received his PhD in Electrical Engineering from Chubu University, Japan, in 1997. From 1997 to 2000 he was a post-doctoral fellow at the Robotics Institute of Carnegie Mellon University, Pittsburgh, PA,

USA, working on the DARPA Video Surveillance and Monitoring (VSAM) effort and the humanoid vision project for the HONDA Humanoid Robot. He is now a professor of the Department of Robotics, Chubu University, Japan. From 2005 to 2006, he was a visiting researcher at Robotics Institute, Carnegie Mellon University. His research interests include computer vision, video understanding and pattern recognition. He is a member of the IEEE, the IEICE, the IPSJ, and the IEE.

Toru Tamaki received his B.E., M.S., and Ph.D. degrees in information engineering from Nagoya University, Japan, in 1996, 1998 and 2001, respectively. After being an assistant professor at Niigata University, Japan, and an associate professor at Hiroshima University, Japan, he is currently a professor at the Department of Computer Science, Nagoya Institute of Technology, Japan. He was an associate researcher at ESIEE Paris, France, in 2015. His research interests include computer vision, image recognition, machine learning, and medical image analysis.