S & M 3460

Detecting Eating Behavior of Elephants in a Zoo Using Temporal Action Localization

Ken Nishioka,[1,2*] Wataru Noguchi,[3] Hiroyuki Izuka,[4] and Masahito Yamamoto[4,5]

[1]Graduate School of Information Science and Technology, Hokkaido University,
Kita 14, Nishi 9, Kita-ku, Sapporo, Hokkaido 060-0814, Japan
[2]Research Institute of Systems Planning, Inc.,
23-23 Sakuragaoka-cho, Shibuya, Tokyo 150-0031, Japan
[3]Education and Research Center for Mathematical and Data Science, Hokkaido University,
Kita 12, Nishi 7, Kita-ku, Sapporo, Hokkaido 060-0812, Japan
[4]Center for Human Nature, Artificial Intelligence, and Neuroscience, Hokkaido University,
Kita 12, Nishi 7, Kita-ku, Sapporo, Hokkaido 060-0812, Japan
[5]Faculty of Information Science and Technology, Hokkaido University,
Kita 14, Nishi 9, Kita-ku, Sapporo, Hokkaido 060-0814, Japan

(Received May 8, 2023; accepted October 3, 2023)

Keywords: animal behavior recognition, temporal action localization, zoo, elephants

The behavioral observation of animals in zoos is indispensable for their health management and the improvement of their breeding environment. However, the day-to-day recording of animal behaviors is time-consuming for zookeepers. Hence, we aim to automatically generate animal behavioral observation reports, called "ethograms", in cooperation with the Sapporo Maruyama Zoo. While studies using contact sensors [e.g., accelerometers, global positioning system (GPS) , and radio frequency identification (RFID)] have had some success in zoos, noncontact sensors (e.g., cameras and microphones) tend to be avoided because of frequent occlusion and the need for nighttime detection. However, noncontact sensors are preferable to contact sensors owing to animal welfare concerns. Here, we propose a method for automatic elephant behavior recognition based on elephant tracking information using video from surveillance cameras. In particular, we focus only on "eating", which is difficult to detect accurately because it requires relatively long-term observation. Therefore, we solve the problem by using a method based on temporal action localization (TAL), which is a task of predicting when and where a target action is performed over a relatively lengthy period. The TAL method has been applied mainly to humans and less to animals. In our experiments, the average precision of eating behavior detection using TAL was 0.853. The results show that TAL is also effective in animal behavior recognition.

1. Introduction

The behavioral observation of animals in zoos is indispensable for their health management and the improvement of their breeding environment. Zookeepers regularly observe the behavior

[*]Corresponding author: e-mail: nishioka@ist.hokudai.ac.jp
https://doi.org/10.18494/SAM4501

ISSN 0914-4935 © MYU K.K.
https://myukk.org/

of animals, for example, whether they are eating the appropriate amount of food or attacking other animals. In addition to direct observation, by watching video footage from surveillance cameras in the animals' breeding area, zookeepers also record the presence or absence of abnormal behavior, breeding behavior, growth condition, and sleeping time. Through these observations, they can take reform measures, such as changing an animal's amount and type of food or keeping an animal apart from other animals. However, the day-to-day recording of animal behaviors is time-consuming for zookeepers.

Various sensors have been used in several animal behavioral recognition studies. Many studies conducted in zoos using contact sensors [e.g., accelerometers, global positioning system (GPS), and radio frequency identification (RFID)] have had some success.(1,2) Conversely, the use of noncontact sensors (e.g., cameras and microphones) tend to be avoided in zoos because of the need to hide behind partitions and feeders in the breeding area and the need for nighttime detection. However, noncontact sensors are preferable to contact sensors owing to animal welfare concerns.

We aim to automatically generate animal behavioral observation reports, termed "ethograms", in cooperation with the Sapporo Maruyama Zoo, using only video from surveillance cameras. From our research, we have developed a highly accurate method for the individual tracking of Asian elephants (hereafter referred to simply as "elephants"). In this study, we propose a method for automatic elephant behavior recognition based on elephant tracking information.

We focus only on and attempt to detect eating behaviors. Detecting "walking" and "sleeping" is also important for checking elephants' health, but these are easier because tracking information allows us to observe how the elephants move throughout the day, and sleeping has a distinct appearance compared with other behaviors. By contrast, it is difficult to determine that an elephant behavior indicates "eating" without long-term observation of the behavior. This is because when it is observed for only a short duration, it is easy to miss the behavior of an elephant facing opposite to the camera or to misidentify it as "searching for food".

In ethograms, it is also important to accurately record the amount of time animals spend on each behavior, as well as the number of times the animal performs that behavior. Therefore, eating behavior must be detected even if the elephant is eating facing away from the camera. In addition, dividing a single eating time into multiple eating times must be avoided.

Our method is based on temporal action localization (TAL), which is the task of predicting when a target action is performed over a relatively lengthy period. The TAL method has been applied mainly to humans and has only been applied to a much lesser extent to animals. Existing research on automatic animal behavior recognition is often based on models that predict behavior over a short period (10 s to several minutes) and are sensitive to instantaneous movements. We show in this study that TAL is also effective for animal behavior recognition.

2. Related Research

2.1 Studies on vision-based human action recognition

2.1.1 Temporal action classification

Temporal action classification was used to classify actions in trimmed video datasets. The benchmark video datasets often used are trimmed videos of approximately 10 s [30 frames per second (FPS)], such as HMDB-51,(3) UCF101,(4) and Kinetics.(5) Convolutional neural networks (CNNs) have shown high performance in image classification, and many action classification methods use CNNs. Tran *et al.* proposed a method using a 3D CNN and demonstrated that it could learn spatiotemporal features.(6) However, it is difficult to train the models, and a large amount of video datasets are required. In the case of image classification, pretraining with a large amount of data is effective. Carreira and Zisserman proposed a technique called I3D, which converts a pretrained 2D CNN to a 3D CNN for video classification, and showed its higher performance.(7) However, 3D CNN is computationally expensive. Wang *et al.* proposed a method called Temporal Segment Network (TSN) that uses features extracted by a 2D CNN instead of a 3D CNN. The TSN splits the time series into partitions, extracts a feature from each partition, and performs classification by consensus (e.g., averaging) of each feature.(8) Learning methods using optical flows have also been proposed for both I3D and TSN. These are called two-stream systems, in which the precomputed optical flow and RGB images are used simultaneously for learning.

Recent studies have proposed models that compensate for the shortcomings of the TSN and I3D. For example, the Temporal Shift Module (TSM) compensates for the disadvantage of TSN, which cannot obtain temporal information by swapping feature channels in the time-axis direction.(9) X3D successfully reduced the computational cost of I3D for efficient learning and estimation.(10) The methods described above mainly use CNNs, but several methods that apply vision transformers to videos have also been proposed.(11) These methods significantly improve accuracy.

2.1.2 Temporal action localization

TAL is a task that predicts when a target action is taken from untrimmed videos that are longer than the action classification. The benchmark video datasets for TAL were THUMOS 14(12) and ActivityNet v1.3.(13) The target actions in these datasets often contain multiple events or actions rather than a single action. For example, the “SoccerPenalty” class is included in THUMOS 14, with the target action section spanning from the moment the ball is kicked to the moment it enters the goal.

The typical TAL process is as follows: First, a pretrained action classification model is used to extract short-term features. TSN and I3D are frequently used for feature extraction. These features are then joined as feature sequences to provide a long-term context. Then, from the feature sequence, another network is used to propose intervals of “action”, which is called "proposal generation". Subsequently, action labels are predicted for each proposed interval.

The boundary-matching network (BMN) is a typical TAL architecture.(14) BMN outputs scores for the start and end boundaries of an action and a confidence score for each interval. The BMN can output highly reliable scores by predicting scores as a two-dimensional dense map with the axes of start time and duration of the target action. In the case of the TAL, several models have been proposed to improve the BMN. The original BMN uses CNN for time feature vector aggregation. To obtain long-time features, several methods using graph convolutional networks (GCNs) and transformers have been proposed.(15) While these methods have higher accuracy than BMN, they are computationally expensive and require additional training data. We used a CNN in the BMN as in the original.

2.2 Studies on vision-based animal behavior recognition

DeepEthogram(16) is a method of behavioral analysis of laboratory mice that uses optical flow in addition to RGB frames. Unlike our dataset, the background of the videos used in this study was almost unchanged, and only single specimens were validated.

There have been several studies on behavioral recognition in domestic animals. Yin *et al.* conducted a study on the behavioral recognition of cows.(17) The authors proposed Efficient-Long Short-Term Memory (LSTM). The experiments were primarily conducted on the basis of temporal action classification, and five class labels (drinking, standing, lying down, walking, and feeding) were used. Experiments were also conducted on untrimmed videos using classification with a sliding window. However, they reported misjudged cases because they could predict class labels only for short-term videos. For example, there is a case of misjudging "feeding" as "standing" because the behavior of raising the head while feeding looks very similar to "standing".

We did not find any studies on automatic detection of animal behavior in zoos. However, there is high demand for cost-effective behavioral observations. This is supported by many examples of manual behavioral observations using camera traps and CCTV images in zoos.(1) Applications for efficient behavioral observations have emerged. For example, ZooMonitor(18) is a web application for recording animal behavior. The application allows users to create behavioral observation records using a smartphone or tablet, which are stored on a server and can be viewed in an analysis report.

3. Materials and Methods

3.1 Datasets

3.1.1 Basic information on video data

The video data used in this study were surveillance camera images of elephants at the Sapporo Maruyama Zoo. The video was recorded continuously for 24 h in the indoor breeding area of the zoo, except when the camera was operated manually. The recorded data could be obtained every hour as a 5 FPS video file. During daytime (approximately 5:00 a.m. to 6:00 p.m.), the video was recorded in color; however, during nighttime (approximately 6:00 p.m. to 5:00 a.m.), the camera automatically switched to night vision and recorded the video in grayscale. Figure 1 shows an example of a single video frame (left) and a bird's-eye view (right) of the breeding area. The camera position and shooting angle were always fixed, and the images used in this study were captured by looking down on the breeding area from a certain height.

Four elephants (three females and one male) were kept at the Sapporo Maruyama Zoo. Each elephant was either kept with the others in the same breeding area or in a separate breeding area for various reasons. In this study, we focused on the behavior of female and male elephants in a specific breeding area.

3.1.2 Detection of target behavior

We limited behavioral labels to "eating;" furthermore, only one of various feeding methods was targeted. The main feeding method was to eat from a suspended net; nonetheless, they sometimes ate food in other ways, such as food in tires or food hidden in wall holes. Some of these feeding methods were employed on a trial basis for only a short period, and some of them were not captured by the camera. If we tried to respond to all of these feeding methods, it would be difficult to demonstrate the effectiveness of the proposed method, because people have differing judgments on whether such irregular ways mean "eating" or not. Therefore, in this study, we limit our method to feeding from a suspended net.

The typical eating behavior is as follows: First, the elephant approaches the feeding area (i.e., the area under the net) and stops. Next, it grabs hay on the ground or in the net with its trunk and brings it to its mouth. It chews for a few seconds and grabs the next piece of hay. After repeating this behavior several times, it leaves the feeding area.

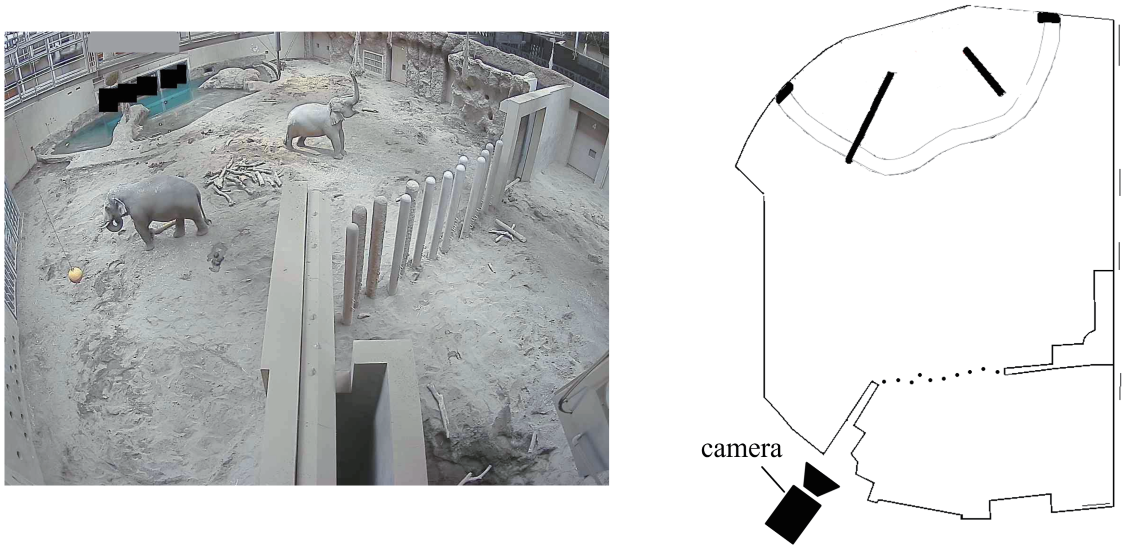

Fig. 1. (Color online) A video frame (left) and a bird's-eye view (right).

3.1.3 Annotation policy

On the basis of the above, the following annotation policy was established. Figure 2 shows an example of an annotation.

1. The behavior of the elephants eating hay out of the net using their trunk is labeled.
2. The cases in which the eating behavior lasts longer than 3 s are targeted.
3. Eating behaviors other than net eating (e.g., taking food from a hole in the wall) are not labeled.
4. The start of the behavior is defined as the time when the target individual stops walking and starts eating.
5. The end of the behavior is defined as the point in time when the target individual finishes eating and begins to leave the area.
6. Every "eating" behavior should include both "raises its trunk upward" and "chewing" movements; if only one of the two movements is observed, the behavior is not labeled.

Note that by virtue of No. 6, even if the target elephant ate hay on the ground, the behavior is not labeled unless the target individual raises its trunk upward. We established this policy because this happens only in a situation where the target elephant finds hay on the ground while walking. The target situation is limited to when the individual intentionally goes to the feeding area for feeding.

The dataset for the experiments was created from surveillance camera images from February 1st to March 31st, 2022, recorded between 11 am and 4 pm or 7 pm and 11 pm. Data captured from February 1st to 28th, 2022 were used as training data, from March 1st to 9th, 2022 as validation data, and from March 10th to 31st, 2022 as test data. Consequently, the total length of annotated videos for male and female was approximately 130 h. Of this total, 61 h were spent on the eating behaviors of both male and female.

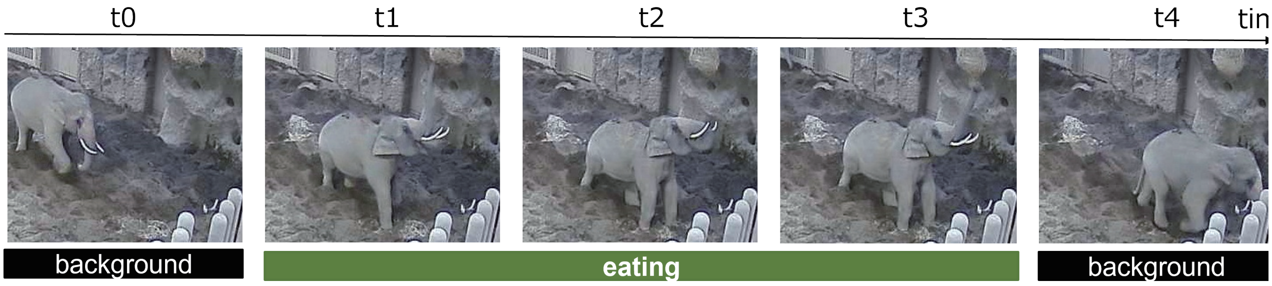

Fig. 2. (Color online) Annotation of action.

3.2 Difficulties

In this section, we discuss some of the difficulties in determining behavior in our dataset.

3.2.1 Little difference between "Searching for food" and "Eating food"

We labeled behaviors such as searching for food as "background" instead of "eating". To distinguish "searching" from "eating", it was necessary to observe the behavior over a relatively large number of video frames.

3.2.2 Long and varied eating times

The duration of each eating activity ranged from a minimum of 15 frames to a maximum of 10565 frames. Figure 3 shows the frequency of the number of frames labeled as "eating" and the cumulative ratio. More than 60% of "eating" have over 1000 frames.

3.3 Problem formulation

In this section, we formulate the problem to be solved before describing our method. Let $V=\{X_n\}_{n=}^{l_v}$ be the video data, X_n, a frame image, and l_v, the total number of frames in video V. The video data V is accompanied by tracking information $T=\{T_n\}_{n=1}^{l_v}$ that indicates the rectangular position and ID of each individual in each frame image. For each individual from the video V and tracking information T, the model outputs proposals $\mathcal{P}=\{(t_{s,n},t_{e,n},p_{conf})\}_{n=1}^{N_p}$,

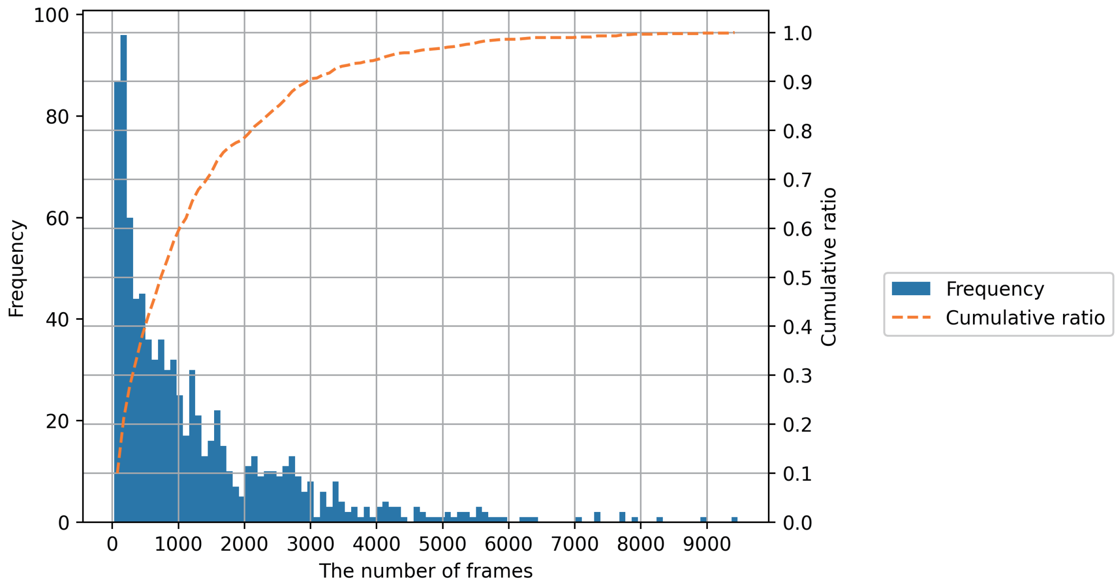

Fig. 3. (Color online) Frequency of "eating" frames and cumulative ratio.

indicating the estimation of the start and end of a behavior, where N_p is the number of target actions in the video V, $t_{s,n}$, and $t_{e,n}$ are the times of starting and ending, respectively, and p_{conf} is the confidence score of the interval.

3.4 Method overview

In this section, we outline the process flow (Fig. 4). First, the video, the estimated rectangular position of the elephant, and its identification ID in our proposed method are input. This process uses a pretrained model for tracking. Second, each frame image is cut for each individual at the estimated rectangular position and resized to a fixed size. Third, from the cut images in the fixed length segments, features are extracted using a pretrained temporal action classification model. Finally, the features are input to TAL, and the action segment of the target animal is estimated.

In this study, the range of action classification is limited to a situation in which two elephants are far from each other. Although the method assumes that the tracking is accurate, the accuracy will decrease when there is overlapping with another elephant, and the tracking result flickers, making it impossible to recognize the action when it is viewed as a video clip. This problem can be solved by improving the tracking accuracy, and in this study, we deal with locations that are tracked accurately.

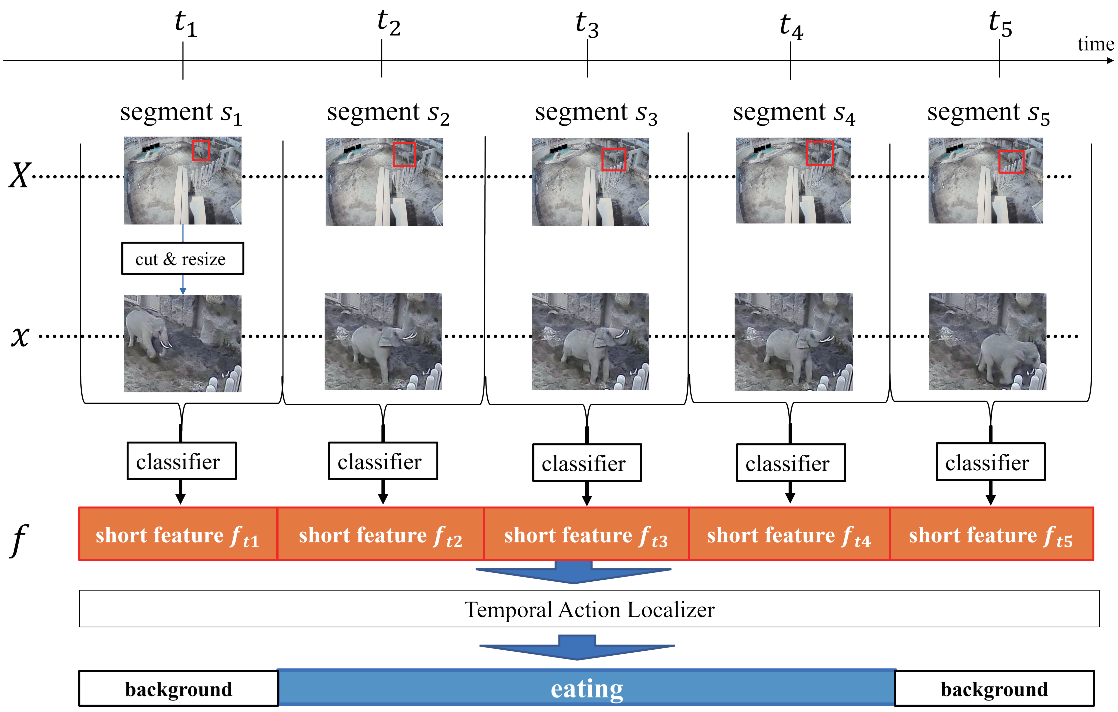

Fig. 4. (Color online) Method overview.

3.5 Preprocess

The tracking information T_n associated with each frame image X_n is used to cut out the rectangular position of each individual elephant. The cut image is resized to a fixed size. In our experiments, the image size is 256 × 256, and the two cut-out methods are as follows: (1) cut out using the raw rectangle (i.e., cut out along the rectangular position of each individual elephant) and (2) cut out using a square. In case 1, the aspect ratio is changed by resizing, which may affect learning. In case 2, the aspect ratio does not change, but the possibility that another elephant is included increases. In addition, we compare the accuracy with and without padding for cut images. The padding method is to fill the cut image with black so that it is a square and to resize it to 256 × 256.

3.6 Feature extraction

Figure 5 shows the feature extraction procedure and TAL inputs. Let $x = \{x_n\}$ be the image after cut out and x is divided into non-overlapping short time segments, s_1, ..., s_L. Here, L is the number of segments, each s_n has up to σ frames, and the t_n-th frame at its center $\{t_n\}$ is selected from $\{1, ..., l_v\}$ at equal intervals. For each segment, feature extraction is performed using the learned video classification model. As a result, $F = \{f_{t_n}\}_{n=1}^{L} \in R^{C \times L}$ is obtained, where $f_{t_n} \in R^C$ is the feature vector around frame x_{t_n} and C is the feature dimension. In the original study on BMN, the feature vectors were the output scores of the classification model. Here, in addition to the output scores of the classification model, we also compare the accuracy of the outputs of the middle layer of the classification model.

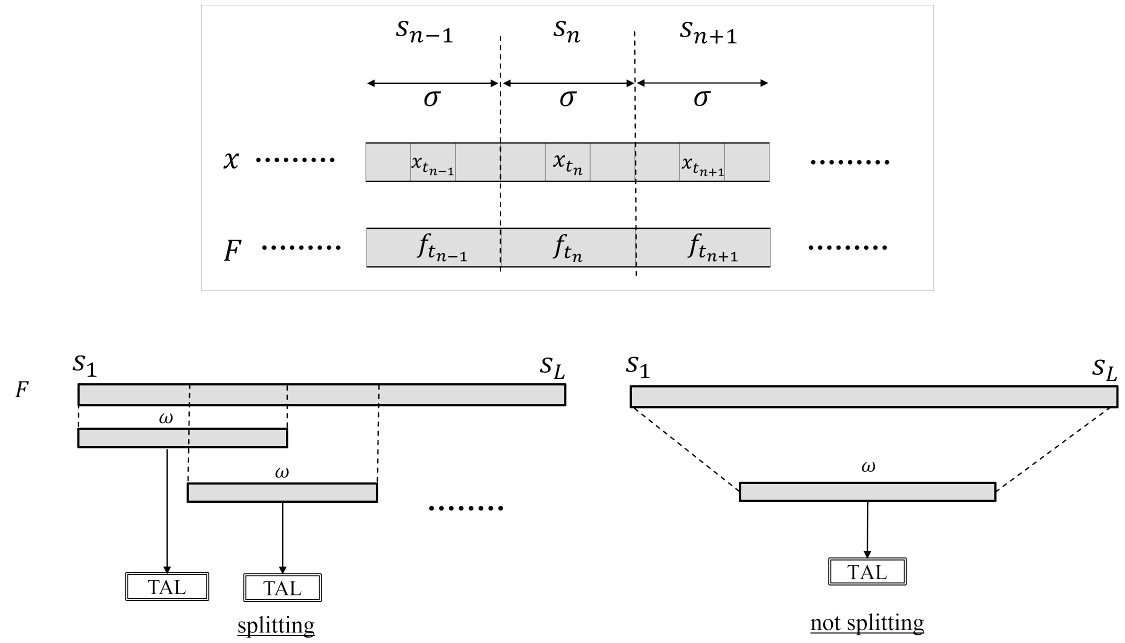

Fig. 5. Feature extraction procedure and TAL inputs.

3.7 Temporal action localization

The TAL model takes a feature sequence $F = \{f_{t_n}\}_{n=1}^{L}$ as input and outputs a proposal $\mathcal{P}$. However, the dimensions of features input to the model must be a fixed. Therefore, the length of the feature sequence ω is determined in advance. If $L \leq \omega$, linear interpolation is performed to make the length of the feature sequence ω. If $\omega < L$, there are two methods. One is to split the feature sequence (bottom left in Fig. 5), and the other is not to split it (bottom right in Fig. 5). In the former, the sequence is split into multiple sequences of lengths ω, where a sliding window is used. In the latter, the sequence is directly resampled to length ω. We show in our experiment that splitting the feature sequence is better.

The final output is obtained after applying SoftNMS[19] to the model's predicted proposals. In the case of splitting, we take the union of the estimated overlapping intervals with a confidence score above a certain threshold.

4. Experiments

In this section, we first describe the feature extractor training experiments using TAL. The TAL models are trained as classification models. Next, we describe the training experiments of the TAL models in which a feature extractor is used. Finally, we compare multiple parameters of TAL and present the results.

4.1 Temporal action classification

4.1.1 Dataset for classification

Class labels were defined as "eating" and "background". The number of data points was adjusted to be equal among classes. The existing classification model input was approximately 300 frames of data, which is called "clip". Therefore, in our dataset, we set the maximum number of frames to 300. Table 1 lists the datasets created. The total number of clips was 3530 and each clip contained 266.5 frames on average.

4.1.2 Model

Classification models were compared with the widely used TSN and I3D models. All the models had a Resnet50 backbone, which was pretrained in ImageNet. Although some methods,

Table 1
Action classification datasets.

Phase	Clips	Period
Train	1816	From February 1 to 28, 2022
Validation	604	From March 1 to 9, 2022
Test	1100	From March 10 to 31, 2022

such as those using optical flow or recently proposed models, are considered more accurate, we did not use these for the following reasons. First, optical flow is computationally expensive, and its real-time operation is not considered feasible. Second, TSN and I3D are often used as the base networks of TAL. We used TSN and I3D to determine whether TAL improved the accuracy of behavior detection.

Figure 6 shows the preprocessing of the inputs for the classification models. For a TSN, the frames are first divided into partitions. Next, features are extracted from the divided partitions, and the TSN outputs a confidence score for each partition. Finally, the scores are averaged across the partitions. Because of the high computational cost of training, the number of partitions is usually changed between training and testing. During training, one frame is randomly selected from each partition. During testing, one frame is selected from each partition such that all frames are equally spaced. In the case of I3D, consecutive frames at two intervals are the input, and the confidence scores are the output. Then, the scores are averaged.

4.1.3 Results

Table 2 presents the results of the study. In the model column, TSN-*x*-*y* indicates that the TSN models were trained using *x* partitions and evaluated using *y* partitions. I3D-*z* indicates I3D models that use *z* consecutive frames at two intervals.

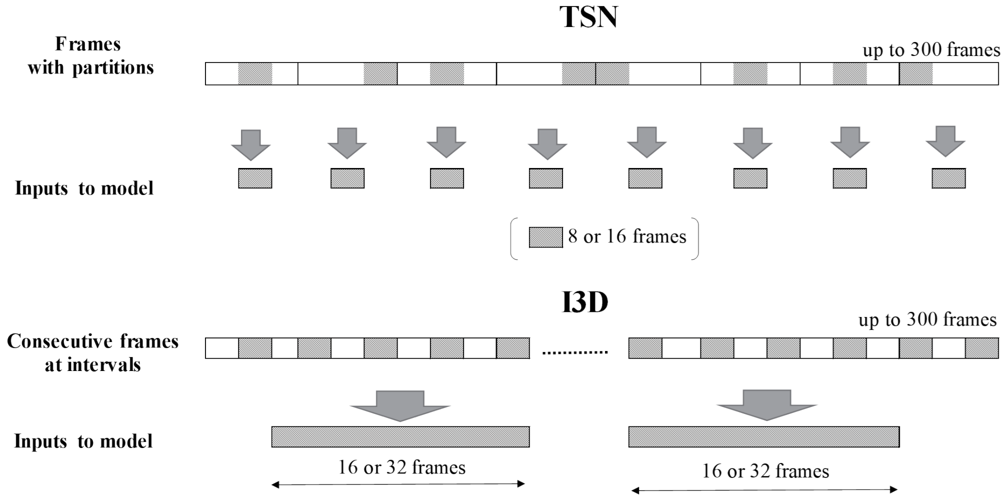

Fig. 6. Preprocessing of classification models.

Table 2
Classification of results.

No.	Model	Padding	Preprocess	Accuracy (%)
1	TSN – 8 – 30		cut out by a raw rectangle	88.47
2	TSN – 8 – 30	✓	cut out by a raw rectangle	87.66
3	TSN – 8 – 30	✓	cut out by a square	88.02
4	TSN – 16 – 60	✓	cut out by a raw rectangle	87.12
5	**I3D – 16**	**✓**	**cut out by a raw rectangle**	**89.19**
6	I3D – 16	✓	cut out by a square	88.29
7	I3D – 32	✓	cut out by a raw rectangle	87.03

The results show that “I3D-16, padding, cut out by a raw rectangle” has the highest accuracy. The dataset also contains frames for eating behavior facing away from the camera, and searching for food, labeled “eating” and “background”, respectively. However, these behaviors are more likely to be misjudged.

4.2 Temporal action localization

In this section, we experiment with TAL using the results presented in Sect. 4.1.

4.2.1 Model

The BMN was used as the TAL model. This is because TAL had not been applied to animals, and we wanted to validate its accuracy using a typical model that has been revalidated by many researchers.

The BMN has 1D convolutional layers to predict the starting and ending times, and 2D and 3D convolutional layers to evaluate the proposal confidence using the scores of the starting and ending times. The trained I3D model is then used for feature extraction. In the original study on the BMN, optical flow was used, but it was not used in this experiment. The frame interval σ was set to 16. For each video, the I3D outputs confidence scores of eating or not eating as the feature sequence. The sequence was then split into multiple sequences of maximum length ω. Each element of the feature sequence is a two-dimensional vector representing the probability of eating or not eating. This feature sequence is divided into a vector of length ω and input into the model as a 2 ω vector. In this section, we describe the case where ω = 100. The learning rate and other parameters are set to the same parameters as used for ActivityNet, which are described in the original paper on BMN.(14)

4.2.2 Dataset for TAL

Table 3 shows details of the dataset for TAL. In the training and validation datasets, each frame sequence contains at least one eating behavior. Conversely, the test dataset contains frame sequences without eating behavior. In addition, the minimum number of frames (min. l_v) is adjusted to 800 for one video. Therefore, the feature sequences from less than 1600 frames are unsampled by linear interpolation and stretched by a factor of 2 in the shortest case of 800 frames.

Table 3
Dataset for TAL.

Phase	Videos	Period	Statistics of “eating” frames				Total frames
			Max.	Min.	Average	Total	
Train	271	February 1–28, 2022	4370	15	979	317292	544262
Validation	117	March 1–9, 2022	4109	15	898	124752	263394
Test	353	March 10–31, 2022	4749	15	894	183262	762617

4.2.3 Evaluation

For comparison with the sliding windows method, the evaluation was based on whether each frame was correctly classified. However, because the background accounted for 90% of the total, the average precision (AP) was used for the evaluation. The precision is denoted as $p(r)$ when the recall is also r. In this case, the interpolated precision p_{interp} was set as

$$p_{interp}(r) = \max_{r \le r'} p(r') \,. \tag{1}$$

AP is calculated as

$$AP = \int_0^1 p_{interp}(r)dr \,. \tag{2}$$

In practice, the integral interval is approximated by 101 divisions.

4.2.4 Comparison of TAL and sliding window

We also compared TAL-based methods with methods that use classification and sliding windows. As the classifier, "I3D-16, padding, cut out by a raw rectangle" was used. The classifier estimated 32 frames, each with a 50% overlap.

4.2.5 Results

The APs of "I3D-16 + BMN" and "I3D-16+ sliding window" were 0.853 and 0.801, respectively, the BMN AP being 0.04 points higher than that of the sliding window. Figure 7 shows a precision–recall curve and Fig. 8 shows a part of the result at 14:00 on March 14, 2022, where the horizontal axis indicates the time (in seconds) elapsed since 14:00, the green area is the "eating" section, and the gray area is the "background" section. For the I3D-16 + sliding window, the line was interrupted. In the case of TAL, the estimation results are distinct. However, in some cases, multiple segments can become single segments.

Figure 9 shows the results of TAL and sliding window for "eating", including the time for eating while facing away from the camera. The green area is the interval where the elephant is judged to be as "eating". In these frames, the elephant eats hay from and under the net while changing direction, which cannot be detected in the case of the sliding window when the elephant is facing away from the camera. However, on the other hand, the TAL correctly detected the elephant eating hay from and under the net while changing direction.

4.3 Ablation study

In this section, we present the comparison of results obtained under several different conditions. This is because the TAL method has been studied mainly in humans and less in

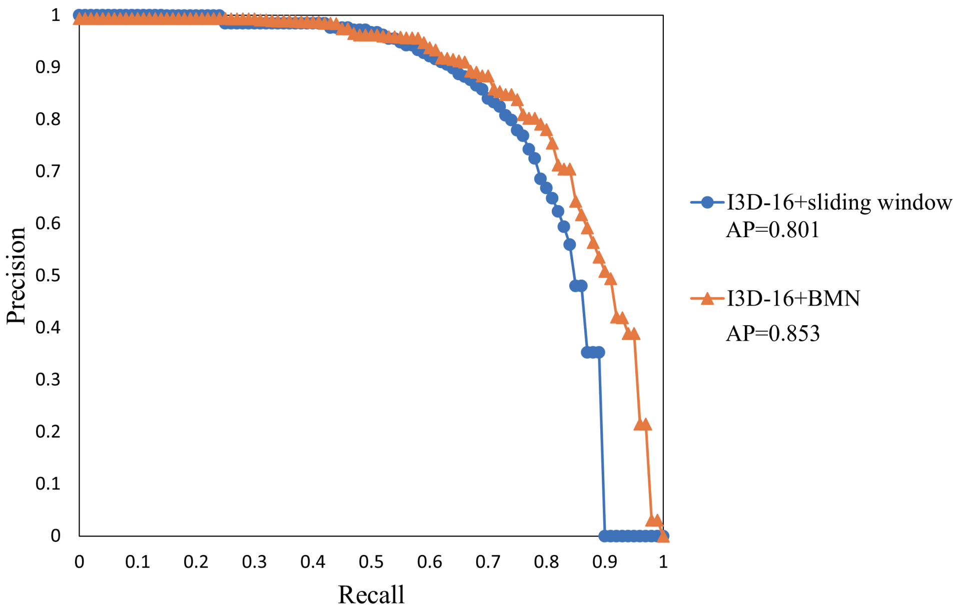

Fig. 7. (Color online) *P*–*R* curve.

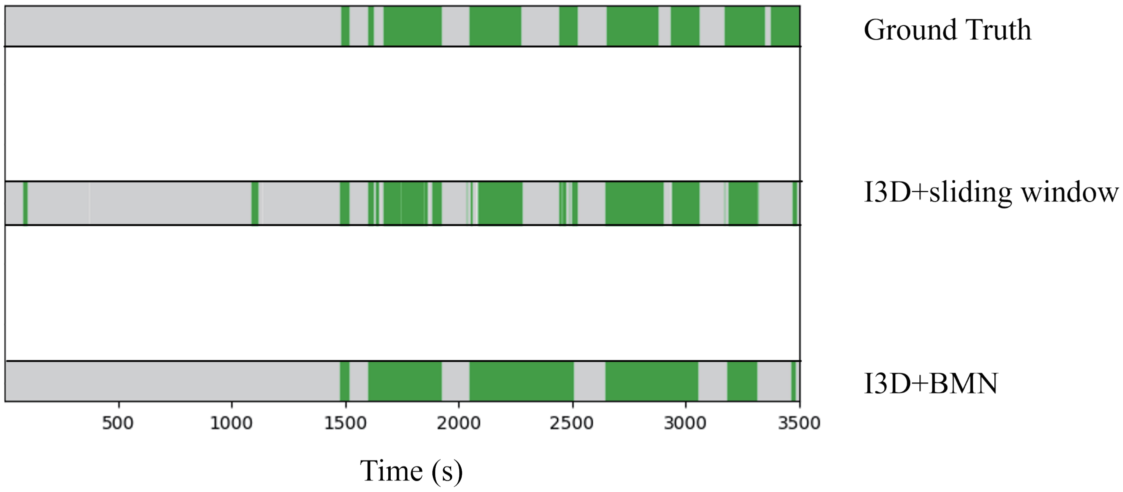

Fig. 8. (Color online) Example of results.

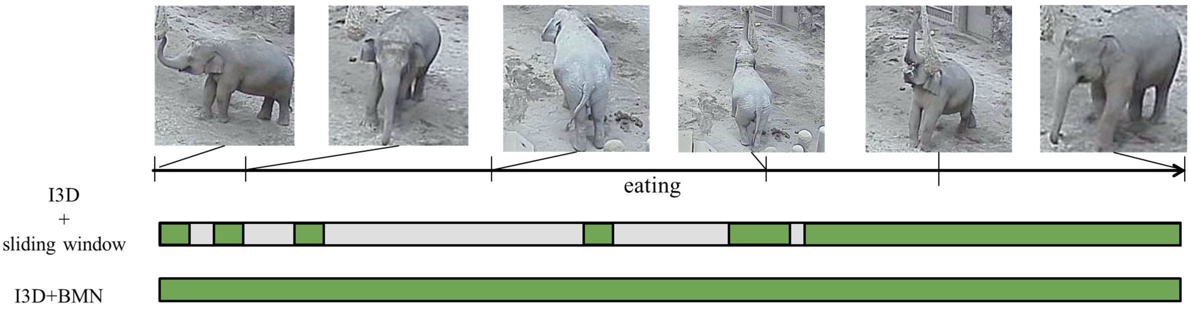

Fig. 9. (Color online) Results of detection of eating behavior facing away from the camera.

animals, and there have been few reports on the effects of preprocessing and models. We describe the means in the columns below. The frame interval σ was set to 16, and the other parameters were set to be the same as those in Sect. 4.2, except for the parameters to be compared.

Table 4 presents the results. The results show that the difference between the highest AP (i.e., 0.854 at C-2) and the lowest AP (i.e., 0.794 at C-3 and C-4) is 0.058. This indicates that the results can vary considerably depending on the feature processing method and the model used.

4.3.1 Feature extractor

We compared I3D-16 with TSN-8-30 as a feature extractor: the preprocessing involved padding and cutting out with a rectangle. The results for A-1 and A-2 show that the I3D-16 AP is better than that of TSN-8-30. This is possibly because I3D-16 uses features more effectively than TSN-8-30, through dense sampling.

4.3.2 Splitting

As described in Sect. 3.7, if the number of segments L is greater than the feature sequence length ω, there are two methods: splitting and not splitting the feature sequence. When I3D-16 was used as a feature extractor (B-1, B-2), the result of "splitting" was better than that of "not splitting". However, when TSN-8-30 was used as the feature extractor (B-3, B-4), the opposite was true. This indicates that the results may vary depending on the architecture of the feature extractor.

In the case of "not splitting", the longer the feature sequences are, the more accurate an estimation can be expected. The sequence length ω was set to 200 in the "not splitting" case. However, owing to GPU memory limitations, we did not evaluate the greater ω cases. If longer frames were used without reducing the resolution, it was expected that the model would predict the intervals more accurately.

Table 4
Results of ablation study.

ID	Feature extractor	Split	Sequence length ω	Overlapping ratio	Feature type	AP
A-1	**I3D-16**	✓	100	0.5	Score	**0.853**
A-2	TSN-8-30	✓	100	0.5	Score	0.816
B-1	**I3D-16**	**✓**	**100**	**0.5**	**Score**	**0.853**
B-2	I3D-16		200	0	Score	0.843
B-3	TSN-8-30	✓	100	0.5	Score	0.816
B-4	TSN-8-30		200	0	Score	0.824
C-1	I3D-16	✓	100	0.5	Score	0.853
C-2	**I3D-16**	**✓**	**100**	**0.9**	**Score**	**0.854**
C-3	TSN-8-30	✓	100	0.5	Score	0.796
C-4	TSN-8-30	✓	100	0.9	Score	0.796
D-1	**I3D-16**	**✓**	**100**	**0.5**	**Score**	**0.853**
D-2	I3D-16	✓	100	0.5	Middle Layer	0.843
D-3	TSN-8-30	✓	100	0.5	Score	0.816
D-4	TSN-8-30	✓	100	0.5	Middle Layer	0.809

4.3.3 Overlapping ratio

In the case of frame splitting, the overlapping ratio of frame sequences can be changed. The higher the overlapping ratio, the more frame sequences were used. The results for C-1 to C-4 show that the overlapping ratio has only a slight effect.

4.3.4 Feature types

Feature types refer to the types of feature vector in classification models. Besides the output score of the feature extractor ("Score" in Table 4), the output of the middle layer ("Middle Layer" in Table 4) can also be employed as inputs to TAL. In the experiments, the features after the average pooling of Resnet50 were used, and the dimensions of the feature vector were 2048. Owing to the increased amount of the GPU memory used, we experimented with only four cases. The results for D-1 to D-4 indicate that they may vary depending on the architecture of the feature extractor.

5. Discussion

The aim of this study was to show that TAL is more effective than a classification-sliding window for automatic elephant behavior recognition. Although our study was limited to eating behavior recognition, our results suggest that accurate automatic behavior recognition is possible. In addition, TAL not only achieved a higher AP than a classification-sliding window, but also detected eating behavior with the elephant facing away from the camera.

However, there were cases where discontinuous eating events were detected as a single eating event. If the number of eating events is essential for daily health checks, this may be problematic.

For more accurate recognition, the following may be considered.

1. Learning with longer feature sequences.
   In Sect. 4.3, we had to reduce the amount of information owing to high GPU memory usage; however, more accurate predictions may be possible if the hardware aspect can be resolved.
2. Training the feature extractor with more action labels.
   In TAL, the localization performance depends on the quality of the feature extractor. In this study, the feature extractor was trained with a limited number of action labels, "eating" and "background". The performance of TAL is expected to improve at the cost of annotation, by using more action labels to train the feature extractor.

In this paper, we described a method for detecting only one behavior; however, it is necessary to detect multiple behaviors for more accurate daily health checks. This can be achieved by estimating the behavior label for each generated proposal. Specifically, segments within a proposal are sampled and fed into a classifier. Next, the weighted average of the classifier scores is calculated by taking the average of the softmax scores or using the output of another model (e.g., neural network or linear classification) that is pretrained using the classifier scores.(15,20) However, human behavior was the main target of detection in these studies. In the case of animals, further verification is required.

6. Conclusion

Noncontact sensors (e.g., cameras and microphones) tend to be avoided because of frequent occlusions and the need for nighttime detection. However, noncontact sensors are preferable to contact sensors owing to animal welfare concerns. We proposed a method for automatically estimating the "eating" time of elephants in zoos using only video from surveillance cameras. This method is based on an existing TAL method, the BMN. A dataset of two months of surveillance video frames was created. In the experiments, we compared the classification-sliding window method with the TAL method. The TAL method achieved a higher AP and produced more distinct behavior boundaries than the classification-sliding window method.

Our final goal is to detect changes in the elephant's physical condition by measuring the time, place, and number of occurrences of prespecified behaviors. Although the current model is highly accurate, to verify the results, we need to compare them with the elephants' actual physical condition records. We defer this task to a future study.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number 22H03637. The authors wish to thank Sapporo Maruyama Zoo for providing the animal data.

References

1 A. Diana, M. Salas, Z. Pereboom, M. Mendl, and T. Norton: Animals **11** (2021) 3048. https://doi.org/10.3390/ani11113048
2 J. Whitham and L. Miller: Anim. Welfare **25** (2016) 395. https://doi.org/10.7120/09627286.25.4.395
3 H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre: 2011 Int. Conf. Computer Vision (IEEE, 2011) 2556. https://doi.org/10.1109/ICCV.2011.6126543
4 K. Soomro, AR. Zamir, and M. Shah: arXiv Preprint (2012) arXiv:1212.0402. http://arxiv.org/abs/1212.0402
5 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: arXiv Preprint (2017) arXiv:1705.06950. http://arxiv.org/abs/1705.06950
6 D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri: 2015 IEEE Int. Conf. Computer Vision (IEEE, 2015) 4489. https://doi.org/10.1109/ICCV.2015.510
7 J. Carreira and A. Zisserman: 2017 IEEE Conf. Computer Vision and Pattern Recognition (IEEE, 2017) 4724. https://doi.org/10.1109/CVPR.2017.502
8 L. Wang, Y. Xiong, Z. Wang, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. V. Gool: 14th European Conf. Computer Vision (2016) 20. https://doi.org/10.1007/978-3-319-46484-8_2
9 J. Lin, C. Gan, and S. Han: Proc. IEEE Int. Conf. Computer Vision (IEEE, 2019) 7082. https://doi.org/10.1109/ICCV.2019.00718
10 C. Feichtenhofer: Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition (IEEE, 2020) 200. https://doi.org/10.1109/CVPR42600.2020.00028
11 J. Selva, A. S. Johansen, S. Escalera, K. Nasrollahi, T. B. Moeslund, and A. Clapes: IEEE Trans. Pattern Anal. Mach. Intell. **45** (2023) 12922. https://doi.org/10.1109/TPAMI.2023.3243465
12 H. Idrees, A.R. Zamir, Y.G. Jiang, A. Gorban, I. Laptev, R. Sukthankar, and M. Shah: Comput. Vision Image Understanding **155** (2017) 1. https://doi.org/10.1016/j.cviu.2016.10.018
13 F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles: 2015 IEEE Conf. Computer Vision and Pattern Recognition (IEEE, 2015) 961. https://doi.org/10.1109/CVPR.2015.7298698
14 T. Lin, X. Liu, X. Li, E. Ding, and S. Wen: 2019 IEEE/CVF Int. Conf. Computer Vision, (IEEE, 2019) 3888. https://doi.org/10.1109/ICCV.2019.00399
15 E. Vahdani and Y. Tian: IEEE Trans. Pattern Anal. Mach. Intell. (2022) 1. https://doi.org/10.1109/TPAMI.2022.3193611

16 J. Bohnslav, N. Wimalasena, K. Clausing, Y. Dai, D. Yarmolinsky, T. Cruz, A. Kashlan, M. E. Chiappe, L. Orefice, C. Woolf, and C. Harvey: eLife **10** (2021) e63377. https://doi.org/10.7554/eLife.63377

17 X. Yin, D. Wu, Y. Shang, B. Jiang, and H. Song: Comput. Electron. Agric. **177** (2020) 105707. https://doi.org/10.1016/j.compag.2020.105707

18 J. Wark, K. Cronin, T. Niemann, M. Shender, A. Horrigan, A. Kao, and M. Ross: Animal Behavior and Cognition **6** (2019) 158. https://doi.org/10.26451/abc.06.03.01.2019

19 N. Bodla, B. Singh, R. Chellappa, and L. S. Davis: 2017 IEEE Int. Conf. Computer Vision (IEEE, 2017) 5562. https://doi.org/10.1109/ICCV.2017.593

20 L. Wang, Y. Xiong, D. Lin, and L. Gool: Proc. 30th IEEE Conf. Computer Vision and Pattern Recognition (IEEE, 2017) 6402. https://doi.org/doi:10.1109/CVPR.2017.678

About the Authors

Ken Nishioka received his B.S. degree from Hokkaido University, Japan, in 2017 and his M.S. degree from the Department of Mathematics, Faculty of Science, Hokkaido University, Japan, in 2017. Since 2017, he has been a software engineer at the Research Institute of Systems Planning, Inc. Since 2021, he has been a Ph.D. student at the Graduate School of Information Science and Technology, Hokkaido University. His research interests are in computer vision and deep learning application (nishioka@ist.hokudai.ac.jp)

Wataru Noguchi received his Ph.D. degree in information science and technology from Hokkaido University, Japan, in 2019. From 2019 to 2023, he was a postdoctoral researcher at Hokkaido University. Currently, he is a specially appointed assistant professor at the Education and Research Center for Mathematical and Data Science, Hokkaido University. His research interests include artificial intelligence, deep learning, and cognitive modeling.

Hiroyuki Iizuka received his Ph.D. degree in multidisciplinary sciences from the University of Tokyo, Japan, in 2004. Since 2005, he has been a research fellow at the Japan Society for the Promotion of Science. From 2005 to 2006, he was a visiting research fellow at the Centre for Computational Neuroscience and Robotics, University of Sussex. He has served as an assistant professor at the Human Information Engineering Laboratory, Osaka University (2008–2013), and as an associate professor at the Autonomous Systems Engineering Laboratory, Hokkaido University, Japan (2013–2023). Currently, he is a specially appointed associate professor at the Center for Human Nature, Artificial Intelligence, and Neuroscience (CHAIN) at Hokkaido University, Japan. His research interests include artificial life, artificial intelligence, embodied cognitive science, complex adaptive systems, deep learning, virtual reality, and the origins of life.

Masahito Yamamoto received his Ph.D. degree from the Graduate School of Engineering, Hokkaido University, Japan, in 1996. He has been a research fellow at the Japan Society for the Promotion of Science (1996–1997) and an assistant professor (1997–2000) and an associate professor (2000–2012) at Hokkaido University. Since 2012, he has been a professor at the Autonomous Systems Engineering Laboratory, Hokkaido University, Japan. Since 2020, he has also been a concurrent faculty member of the Center for Human Nature, Artificial Intelligence, and Neuroscience at Hokkaido University. His research interests include artificial life and intelligence, swarm intelligence, combinatorial optimization, and board game artificial intelligence (AI) programming.