Animal Behavior Classification Using DeepLabCut

Shiori Fujimori
Graduate School of Fundamental Science and Engineering
Waseda University
Tokyo, Japan
srn.1953.jng@fuji.waseda.jp

Takaaki Ishikawa
Global Information and Telecommunication Institute
Waseda University
Tokyo, Japan
takaxp@ieee.org

Hiroshi Watanabe
Graduate School of Fundamental Science and Engineering
Waseda University
Tokyo, Japan
hiroshi.watanabe@waseda.jp

***Abstract*— In this paper, we introduce a method to classify animal behaviors from videos taken by a fixed-point camera. In order to classify animal behavior, it is necessary to detect and track the animals. Conventional approaches for detecting moving objects are based on background subtraction and frame subtraction. Conventional methods are not suitable for detection of animals kept indoors since they are susceptible to sunlight and shadow. We propose a method to track animals and classify their behavior using skeletal information obtained by DeepLabCut. The experimental results show that the proposed method is superior to the conventional method.**

Keywords*— *DeepLabCut, deep learning, animal behavior

I. INTRODUCTION

Recently, pet watching cameras have become popular. This makes it possible to check on a pet from the place we visit using a smartphone or other device. These cameras have various functions such as recording the pet's condition and communicating with the pet from outside. However, if we cannot keep up with the camera from outside, we need to check the recorded video to see what the pet is doing.

If we can know the behavior of pets while going out without checking the video, the surveillance task will be easier. Moreover, knowing the balance between energy consumption by exercise and intaking by a meal will help to manage the health of pets.

The purpose of this study is to classify the cat behavior from videos and detect cat behavior to know the amount of exercise and meal intake. This is not limited to cats, but can be applied to any pets or zoo-owning animals.

II. RELATED WORKS

A. DeepLabCut

DeepLabCut [1] has been proposed by Mathis et. al. as a tool that uses deep learning to predict and track body parts of laboratory animals. DeepLabCut utilizes the feature detectors from the multi-headed pose estimation algorithm DeeperCut [2]. These feature detectors are based on Deep Residual Neural Networks (ResNet) [3], which enables DeeperCut to estimate human body parts with high accuracy from images.

DeepLabCut replaces the output of the ResNet with a deconvolution layer. The deconvolution layers are specified for each body part and output different images of the joint positions of the animal to be tracked. Each of the output images has a layer that outputs a score map. This score map shows the probability that each body's position is in a particular pixel.

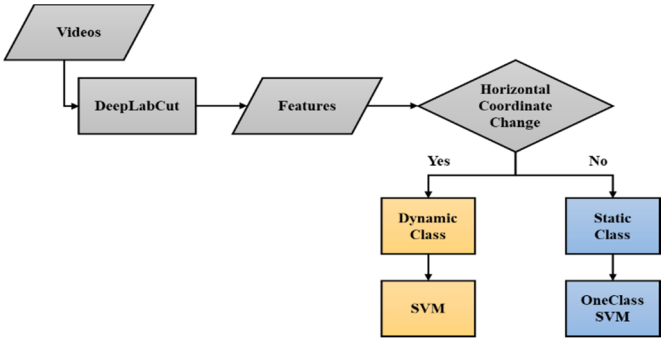

Figure 1 An overview of the proposed method

Moreover, ResNet were pre-trained on ImageNet, a massive dataset for object recognition. This transfer learning enables DeepLabCut to estimate and track feature points by learning a small number of additional frames of label data for the body position of the target animal to be tracked.

B. Outlier detection

Outlier detection is a method for detecting and identifying data points that would not normally occur. We introduce two methods for outlier detection.

The first is OneClassSVM. OneClassSVM is an unsupervised one-class classification method based on SVM devised by Boser et. al. [4]. SVM has multiple classes of data as training data. We classify the data by determining the discrimination boundary. On the other hand, OneClassSVM has only one "normal" class. Therefore, the discrimination boundary is set, and whether it is normal or not by setting the boundary.

The second is IsolationForest. This is an unsupervised learning algorithm. This method detects outliers based on the assumption that outliers are likely to be less than normal. This method isolates observations by randomly selecting a feature and then randomly selecting a split value between the maximum and minimum values of the selected feature.

If distribution of the observations looks like Figure 2, the "a" is divided into the tree structure on the right of the figure. Since it is assumed that outliers are less than normal, we assume that the outlier is the value divided into the shallowest trees. Construct several tree structures and compute the average path length to the end nodes.

Random partitioning produces noticeably shorter paths for outliers. Therefore, when a forest of random trees collectively produces shorter path lengths for particular samples, they are highly likely to be outliers.

978-1-7281-9802-6/20/$31.00 ©2020 IEEE

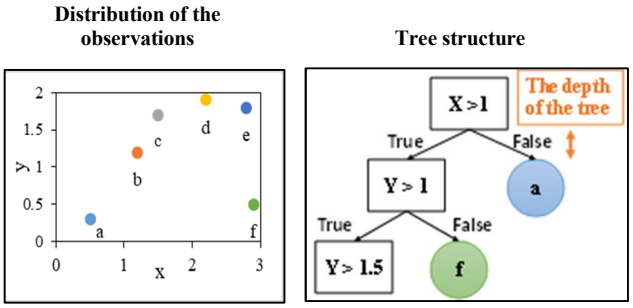

Figure 2 Isolation Forest

C. *Moving object detection*

The background subtraction method is to detect a moving object by preparing a background model in advance and comparing the background with the input image. Although it is a simple algorithm, it is very sensitive to changes in the external environment, such as sunlight conditions, and has poor anti-interference ability.

Another method is the frame subtraction method. Moving objects are detected by calculating the difference between two consecutive frames. This method is more robust to changes in the background image than the background subtraction method. However, it has the disadvantage that if an object is stationary for a while, it will be lost.

III. PROPOSED METHOD

In this research, we propose a method to classify the behavior of indoor cats captured by a fixed-point camera using DeepLabCut. We classify the five behaviors: “sitting”, “lying down”, “eating”, “walking” and “jumping”. An overview of the proposed method is shown in Fig.1.

As the conventional approach for detecting a moving object, there are the background subtraction method and the frame subtraction method. However, the background subtraction method is vulnerable to changes in a sunshine condition. The frame subtraction method loses sight of a cat when the cat is stationary for a while. For these reasons, these conventional methods are not suitable for classifying behavior of cats kept indoors. Therefore, we propose a method that uses the skeletal information obtained by DeepLabCut. The process flow of the proposed method is shown below.

1) Estimate the posture of a cat in an input video using DeepLabCut trained by the prepared dataset.

2) Calculate the features of the posture in each frame required for the behavior classification by using the coordinate values obtained by posture estimation.

3) Classify behaviors using a class classifier.

The five behaviors can be divided into static and dynamic classes, based on whether there is a horizontal coordinate change. Three behaviors “sitting,” “lying down,” and “eating” are defined as static classes, while the remaining two behaviors "walking" and "jumping" are defined as dynamic classes. Next, for each of the static and dynamic classes, we classify the behavior according to the flow described above.

To classify static classes, the joint angle is used as a feature and classification is performed by Support Vector Machine (SVM). In the "sitting" posture, the front legs are extended, but in the "lying down" posture, the front legs are bent. Therefore, there is a difference in the joint angle of the front legs. In addition, the head is lowered, and the joint angle of the front leg is smaller in the "eating" posture than in the "sitting" posture. Therefore, it is possible to classify the class by using the joint angle.

On the other hand, outlier detection is used to classify the dynamic classes. “Jumping” will cause a sudden change in the vertical coordinates. Therefore, the change rate of vertical coordinates between the previous and current frame is calculated by DeepLabCut and used as a class separation feature. By detecting the change rate, "jumping" can be separated from "walking".

IV. EXPERIMENT

A. *Classification of static classes*

For the dataset, we use 1177 frames of 1920 × 1080 [pixel] video for each behavior. The joint angles of the front legs and hind legs are calculated from the coordinates of the legs obtained using DeepLabCut. Using these features, we classify the classes by SVM. The distribution of joint angles for each action is shown in Figure 3.

The joint angles of both legs, front leg only and hind leg only are used as feature points and these three results are compared. Table 1 shows the accuracy of the classification of "lying down" and "sitting" and the accuracy of the classification of "lying down", "sitting", and "eating".

In the two-class classification, the accuracy reaches to 100% when the angles of both legs are used as features. As can be seen from Figure 3, the joint angle distribution of the front legs differs significantly between "lying down" and "sitting". In the case of "lying down", the joint angles of the front legs are less than 100°, while in the case of "sitting" they are more than 100°. Therefore, the accuracy is higher when the joint angles of the front legs are used as features than when only the joint angles of the hind legs are used as features.

However, when "eating" is added, the accuracy is reduced. In particular, when the angle of hind leg only is used as a feature, the accuracy is reduced to 51.3%. As shown in Figure 3, there is an overlap in the distribution of "eating" and "lying down". This led to a decrease in the accuracy. It is necessary to improve the accuracy by adding more feature points, such as the head motion.

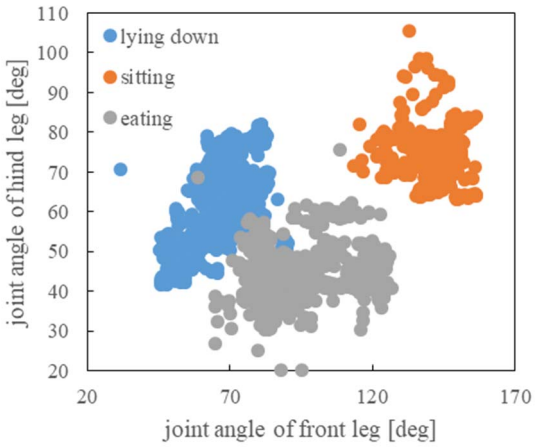

Figure 3 The distribution of joint angles

978-1-7281-9802-6/20/$31.00 ©2020 IEEE

Table 1 Accuracy of static classes classification

	two-class	three-class
Both legs	1.000	0.937
Front leg	1.000	0.871
Hind leg	0.755	0.513

B. Classification of dynamic classes

In the dynamic class, a comparative experiment between the proposed method and the conventional method is firstly conducted to show the superiority of the proposed method. After that, we classify the behaviors of the four videos using the proposed method.

First, experiments are conducted to compare the accuracy of the proposed method with that of conventional methods for moving object detection. Video with 1920×1080[pixel] resolution consisting of 35 frames is used for the comparative experiment. Of all frames, 40% is "jumping" scene and the remaining 60% is "walking" scene. Using this dataset, feature point coordinates are obtained using three methods: DeepLabCut, the background subtraction method, and the frame subtraction method.

In the proposed method, DeepLabCut is used to obtain the coordinates of the front leg. On the other hand, the conventional method uses the background subtraction method and the frame subtraction method to obtain the center-of-gravity coordinates of the difference image. Coordinates obtained by each method are shown in Figures 4, 5 and 6.

Next, the change rate of vertical coordinates between frames is calculated from the coordinates obtained by each method, and outlier detection is performed using OneClassSVM.

Table 2 shows the experimental results. Comparing the F-measures, it is confirmed that the accuracy of the proposed method is higher than that of the conventional methods.

Comparing Figures 4 and 5, we can see that the timing of the start of the jump is slow in the background subtraction method. This is because that shadows of a cat are sometimes reflected in the difference images obtained by the background subtraction method.

Also, in Figure 6, the acquisition of the vertical coordinates of 0-5 frames failed. This is because that the frame subtraction method cannot accurately obtain a differential image when the cat's motion is small. For these reasons, the accuracy of the proposed method is higher than others.

These experiments show that the proposed method is able to obtain the feature point coordinates during motion accurately. The proposed method improves the classification accuracy by 6.8% compared with the background subtraction method and by 13.0% compared with the frame subtraction method. Therefore, the proposed method is superior to the conventional method.

Next, we classify the behaviors of the four videos using the proposed method. Each video is 1920x1080[pixel] in size. We define each video as Video (1), Video (2), Video (3) and Video (4). Table 3 shows the total number of frames and the number of frames for each of the four videos for "walking" and "jumping".

For each video, DeepLabCut is used to obtain the coordinates of the front leg. Using these coordinates, the change rate of vertical coordinates between frames is calculated. The change rate of vertical coordinates is defined as normal for the change rate of walking and outlier for the change rate of jumping. Two methods, OneClassSVM and IsolationForest, are used for outlier detection.

For outlier detection by OneClassSVM, the distance to the hyperplane that divides the normal and the outlier is calculated. A frame with a negative value of this distance is detected as an outlier. OneClassSVM needs to set the percentage of outlier frames. For videos (1)-(4), we compare the accuracy of outlier detection for each percentage while varying the percentage of outlier frames. The changes in accuracy obtained by varying the percentage of outliers are shown in Table 4, 5, 6 and 7.

For outlier detection by IsolationForest, the average of the path lengths to the end nodes of the tree structures is obtained. A frame with a negative path length is detected as an outlier. Video. For Video (1)-(4), the accuracy of the outlier detection by IsolationForest is shown in Tables 8.

We compare the accuracy of OneClassSVM and IsolationForest, where the percentage of outliers is set so that the F-value is the maximum. Comparison results are shown in Table 9.

The accuracy with IsolationForest is 0.71% lower than that with OneClassSVM in Video (2). However, the accuracy of Video (1), Video (3) and Video (4) is 16.3%, 13.8% and 2.7% higher, respectively. When using OneClass SVM, the percentage of outliers can be varied so that the accuracy is maximized. However, we cannot predict how many times the cat will jump while we are away. In other words, the percentage of outliers is not known in advance. Therefore, IsolationForest is suitable for monitoring.

Experimental results show that normal frames are often misidentified as abnormal. When a cat jumps, it bends its legs and bends down in preparation for jumping. Since this preparatory motion frame is sometimes detected as an abnormal frame, we believe that the detection accuracy can be improved by considering the preparatory motion.

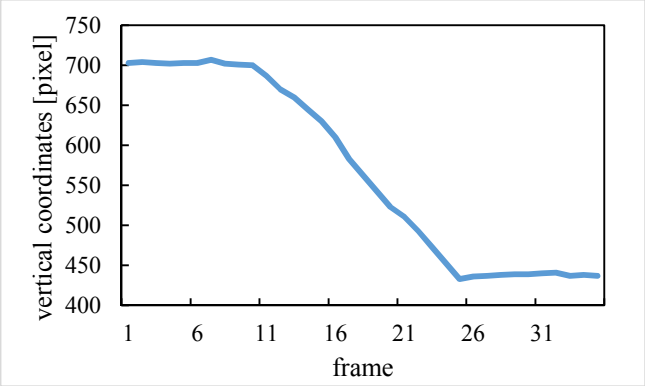

Figure 4 Coordinates obtained by DeepLabCut

978-1-7281-9802-6/20/$31.00 ©2020 IEEE

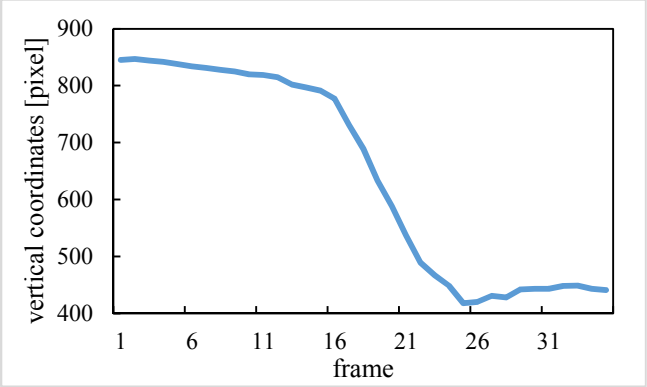

Figure 5 Coordinates obtained by background subtraction

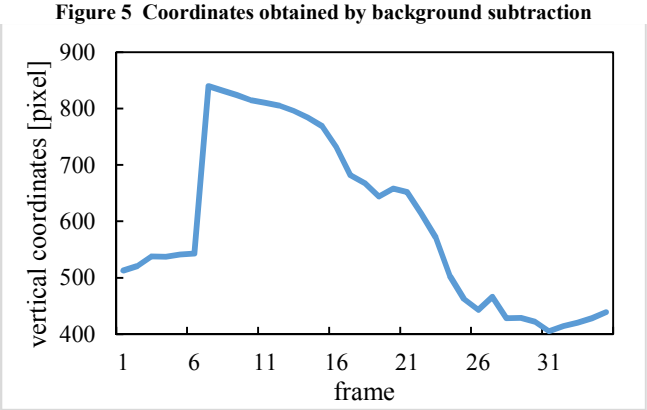

Figure 6 Coordinates obtained by frame subtraction

Table 2 Accuracy of dynamic classes classification

Method	Precision	Recall	F-measure
Proposed	0.714	0.588	0.645
Background subtraction	0.762	0.500	0.604
Frame subtraction	0.667	0.500	0.571

Table 3 Number of frames of Video (1)-(4)

	Vide (1)	Video (2)	Video (3)	Video (4)
Total frame	173	148	189	232
Walking frame	168	142	168	196
Jumping frame	5	6	13	36

Table 4 Video (1) outlier detection accuracy by OneClassSVM

Percentage of outlier values	Precision	Recall	F-measure
0.05	0.9583	0.4000	0.5644
0.06	0.9583	0.4000	0.5644
0.07	**0.9464**	**0.6000**	**0.7344**
0.08	0.9345	0.6000	0.7308
0.09	0.9286	0.6000	0.7290

Table 5 Video (2) outlier detection accuracy by OneClassSVM

Percentage of outlier values	Precision	Recall	F-measure
0.08	0.9507	0.6667	0.7837
0.09	0.9366	0.6667	0.7789
0.10	**0.9366**	**0.8333**	**0.8820**
0.11	0.9225	0.8333	0.8757
0.12	0.9085	0.8333	0.8693

Table 6 Video (3) outlier detection accuracy by OneClassSVM

Percentage of outlier values	Precision	Recall	F-measure
0.13	0.9286	0.6923	0.7932
0.14	0.8988	0.6923	0.7822
0.15	**0.8929**	**0.7692**	**0.8264**
0.16	0.8750	0.7692	0.8187
0.17	0.8690	0.7692	0.8161

Table 7 Video (4) outlier detection accuracy by OneClassSVM

Percentage of outlier values	Precision	Recall	F-measure
0.14	0.9031	0.3889	0.5437
0.15	0.9031	0.3889	0.5437
0.16	**0.8929**	**0.4167**	**0.5682**
0.17	0.8724	0.4167	0.5640
0.18	0.8163	0.4167	0.5517

Table 8 Outlier detection accuracy by IsolationForest

	Precision	Recall	F-measure
Video (1)	0.9167	0.8000	0.8544
Video (2)	0.9225	0.8333	0.8757
Video (3)	0.9583	0.9231	0.9404
Video (4)	0.9745	0.4167	0.5837

Table 9 Comparison of f-measure for each classifier

	OneClassSVM	IsolationForest
Video (1)	0.7344	0.8544
Video (2)	0.8820	0.8757
Video (3)	0.8264	0.9404
Video (4)	0.5682	0.5837

V. CONCLUSION

In this paper, we proposed a method to classify the behavior of cats using DeepLabCut. DeepLabCut can keep tracking of feature points without being affected by shadows or brightness. In this respect, it is confirmed that the moving object can be detected with higher accuracy than the conventional methods, such as the background subtraction method and the frame subtraction method. However, for actual home use, adverse conditions must be taken into account, such as parts of the body being obscured by furniture, etc. Therefore, it is necessary to add feature points such as a head as well as legs to make classification possible even under adverse conditions. In addition, the method needs to be improved, not only by increasing the number of feature points, but also by considering the relationship between time series data.

REFERENCES

[1] A. Mathis, P. Mamidanna, T. Abe, K. M. Cury, V. N. Murthy, M. W. Mathis, and M. Bethge, "DeepLabCut: Markerless tracking of user-defined features with deep learning," arXiv preprint arXiv: 1804.03142, Apr. 2018. Nature Neuroscience, No.21, pp.1281-1289, Aug. 2018.

[2] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele, "DeeperCut: A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model," arXiv preprint arXiv: 1605.03170, May 2016. European Conference on Computer Vision (ECCV2016), pp.34-50, Sep. 2016.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," arXiv preprint arXiv: 1512.03385, Dec. 2015. IEEE Conferene on Computer Vision and Pattern Recognition (CVPR 2016), pp.770-778, June 2016.

[4] B. Boser, I. Guyon, and V. N. Vapnik, "A Training Algorithm for Optimal Margin Classifiers," Proceedings of the 5th Annual Workshop on Computational Learning Theory (COLT'92), pp.144-152, July 1992

978-1-7281-9802-6/20/$31.00 ©2020 IEEE