Rescue Dog Action Recognition by Integrating Ego-centric Video, Sound and Sensor Information

Yuta Ide[1], Tsuyohito Araki[1],
Ryunosuke Hamada[2], Kazunori Ohno[2], and Keiji Yanai[1]

[1] Department of Informatics, The University of Electro-Communications, Tokyo
[2] NICHe, Tohoku University
{ide-y,araki-t,yanai}@mm.inf.uec.ac.jp

Abstract. A dog which assists rescue activity in the scene of disasters such as earthquakes and landslides is called a “disaster rescue dog” or just a “rescue dog”. In Japan where earthquakes happen frequently, a research project on “Cyber-Rescue” is being organized for more efficient rescue activities. In the project, to analyze the activities of rescue dogs in the scene of disasters, “Cyber Dog Suits” equipped with sensors, a camera and a GPS were developed [10].

In this work, we recognize dog activities in the ego-centric dog videos taken by the camera mounted on the cyber-dog suits. To do that, We propose an image/sound/sensor-based four-stream CNN for dog activity recognition which integrates sound and sensor signals as well as motion and appearance. We conducted some experiments for multi-class activity categorization using the proposed method. As a result, the proposed method which integrates appearance, motion, sound and sensor information achieved the highest accuracy, 48.05%. This result is relatively high as a recognition result of ego-centric videos.

1 Introduction

A dog which assists rescue activity in the scene of disasters such as earthquakes and landslides is called a “disaster rescue dog” or just a “rescue-dog”. In rescue activities in the disaster areas, trained rescue-dogs may conduct exploration as human assistants. A rescue-dog makes a pair with a human and investigates disaster areas by making use of special characteristics as a dog. Rescue-dogs can investigate even in the environments where it is difficult for a person to traverse on such as narrow spaces, crevices, and collapsed buildings. In addition, the rescue operation that relies on the dogs’ developed sense of smell is possible. However, they have no language to communicate with humans. A person who pairs up with a rescue dog and gives instructions at a disaster site are called “a handler”, and a handler must understand the information the rescue-dog gathered from its behavior.

At the present, a handler who directs a rescue-dog manually marks the action of the rescue-dog and orally transmits information to a commander of a rescue

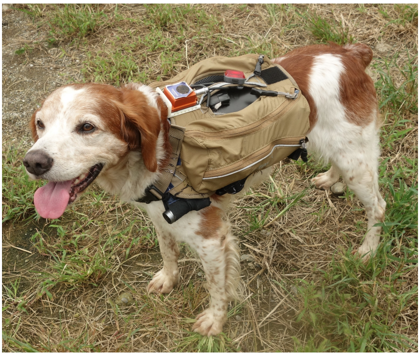

Fig. 1. A rescue dog wearing a "cyber-rescue suit" [10].

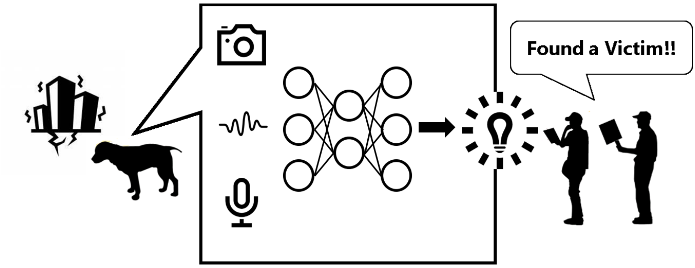

Fig. 2. During the rescue activity of a rescue dog in the disaster site, first-person (dog-centric) video, sound and sensor information are recorded by a camera, microphone and various sensors (such as inertial sensor, accelerometer and pose sensor) mounted on a cyber-dog-suit, and the proposed system recognizes the current dog's behavior from the recorded video, sound and sensor information.

team. The big problem on a joint rescue activity by a handler and a rescue-dog is lack of information on the surrounding environment and victims for triage[3]. The records by the handler are subjective, and they tend to lack objectivity. The verbal transmission also makes them less accurate.

For this situation, in Japan where earthquakes happen frequently, a research project on "Cyber-Rescue" is being organized for more efficient rescue activities. In the project, to analyze the activities of rescue dogs in the scene of a disaster, "Cyber-Dog Suits" equipped with sensors, a camera and a GPS were developed [10] (Fig. 1).

In this study, we aim to estimate the behavior of rescue-dogs using sensor data obtained from the cyber-dog suits. We analyze ego-centric videos taken by the camera mounted on the cyber-dog suits, and recognize dog activities by

[3] "Triage" means the process of deciding who receives medical treatment first, according to how seriously the person is injured.

using not only videos but also sounds and sensor data (Fig. 2). This is expected to make it possible to estimate what the rescue-dog is doing now automatically. The information necessary for triage is organized, and disaster rescue activities will be more efficient by the rescue-dog activity analysis.

To analyze ego-centric videos with audio and sensor data, we propose a image/sound/sensor-based four-stream CNN for dog activity recognition which integrates sound and sensor information as well as motion and appearance. We conducted some experiments for multi-class activity categorization using the proposed method. As a result, the proposed method which integrates appearance, motion sound and sensor information achieved the highest accuracy, 48.05%. This result is a relative high as a recognition result of ego-centric videos.

2 Related Work

2.1 Third-person activity recognition

Two-stream CNN is a method for video classification [13]. It classifies video categories by integrating motion information represented as optical flows and appearance information. There are many kinds of researches on derived networks based on the Two-stream CNN. Convolutional Two-Stream Network Fusion [6] is one of the variant methods. The study achieved the state of the art on the standard benchmark dataset for video classification called UCF-101 by combining the output of the convolution layer of each stream and adding FC layers. In our work, we follow Two-Stream Network Fusion as the way to integrate the stream outputs, and add the two streams for audio and sensor information.

2.2 First-person activity recognition

First person vision has been actively studied so far. There are so many researches such as [11][7]. Minghuang *et al.* [11] proposed a twin stream network that integrates hand segmentation, target object localization and motion. The twin stream network estimates the arm area and object locations with CNN, and integrates them with the CNN output result of the optical flow image via CNN. It is a kind of first person version of the two stream network.

Recently the large-scale egocentric video dataset, EPIC-KITCHEN [2], was released, which had been promoting the researches on ego-centric video analysis greatly [5].

2.3 Dog-Centric activity modeling

There are a few studies on first person video analysis from dog's view. Ehsan *et al.* [4] estimated dog activity from dog-centric video. They modeled dog activity and estimate how dogs will move. However, these studies just model dog behavior and do not estimate interaction between dogs and surroundings. They created DECADE, a dataset of ego-centric dog video and joint movements. The dataset includes 380 video clips from a camera mounted on the dog's head. It also includes corresponding information about body position and movement. Movement were measured by inertial measurement units (IMUs), and sound was also recorded. The dataset is similar to our dataset of cyber-rescue dogs. However, they used only a video in the experiments, and did not use multi-modal information such as sound and IMU data.

Iwashita *et al.* [9] also published a Dog-Centric Activity Dataset (DCAD). This dataset is used for dog behavior classification from dog-centric movies. DCAD does not support rescue-dog specific classes, contains no audio data and assumes a single-class classification.

In our work, we extend Convolutional Two-Stream Network Fusion [6] by adding a sensor stream. In the experiments, we confirm the effectiveness of introducing sensor data for dog activity recognition.

3 Dataset

In this study, we perform multi-class activity estimation of rescue-dogs. The rescue-dog training dataset consists of a group of data collected by sensors embedded in the cyber-rescue suits that rescue-dogs wear. The dataset is still growing, and we are collecting data at the time of rescue training of rescue-dogs in the simulated disaster sites. Due to privacy and ethical issue, it does not contain the data recorded in the actual disaster sites.

It consists of about 2 minutes to 20 minutes of six movies with audio and sensor data. The dataset includes first-person videos of a handler's view and third-person videos showing a handler and a rescue dog in addition to egocentric dog videos. Since handler view videos and third-person videos do not always show the rescue dog, the dog sometimes goes into the area where it is invisible from a handler and a person recording third-person videos. Therefore, in this work, we use only dog-centric videos with audio and sensor data. As a sensor unit, Xsens Mti-300 was built in the cyber dog suits, which can record angular velocity, acceleration, geomagnetic flux, pose, pressure and temperature every 0.005 seconds. Among these information, we use angular velocity, acceleration and pose as sensor information.

The total time is 47 minutes and 5 seconds, the number of frames per second is 29.97fps, and the total number of frames is 84,665. Note that only two videos have sensor data the total time of which is about 20 minutes. We call the dataset containing only two videos with sensor data as "Sensor dataset", while we call the full dataset as "Full dataset". The videos are annotated by specifying a time range for each of 11 activity classes as shown in Table 1. Multiple classes are sometimes overlapped with each other at the same time. Therefore, we treat this task as a multi-label classification task. For example, for the scene of "rescue-dog is barking while finding a victim", two labels, "*see victim*" and "*bark*" are annotated.

Table 1. The frequencies of the 11 dog activity classes in the rescue-dog dataset.

	bark	cling	command	eat-drink	handler	run	see-victim	shake	sniff	stop	walk
Full	9299	3631	8035	876	7066	414	7601	1349	16573	31533	45915
Sensor	2281	1936	4903	145	3718	0	1894	405	8208	13194	14154

3.1 11 Dog Activity Classes

We explain each of the 11 classes of rescue dog activities. Their frequencies in the dataset are not uniform as shown in Table 1.

1. **bark :** The situation in which the dog is barking. Typically a rescue dog barks when finding out victims. There exists easy-to-understand phonetic features, and unique shakes in the videos occur.
2. **cling :** The situation in which a doc is sniffing with the nose close to the smell. It is more detailed than "*sniff*," and this is labeled it always overlaps with "*sniff*."
3. **command :** The situation in which the dog is being instructed by the handler. There are various situations such as verbal instructions such as "Wait" and "Go", praises and pointing instructions.
4. **eat-drink :** The situation in which the dog is eating or drinking something. In addition to feeding success rewards for finding victims under rescue training, there are various situations such as eating grass and drinking water on the ground or a river.
5. **look at handler :** The situation in which the dog is looking at the handler. Hereinafter, this action is called just "*handler*".
6. **run :** The situation in which the dog is running. There is a feeling of floating on the screen compared to the *walk-trot* class, and it is intense that shaking and sounds.
7. **see victim :** The situation in which the victim is appearing in the camera. Hereinafter, called "*victim*".
8. **shake :** The situation in which the dog is vigorously waving. During this activity, sound clatters on the camera on the dog's back.
9. **sniff :** The situation in which the dog sniff surroundings. This can be the indicator that measures the dog's motivation for exploration. This action happens not only when the nose is brought close to the ground, but also when the dog smells the floating odor.
10. **stop :** The situation in which the dog is not stepping and stays in the same spot, which includes stepping on the same spot. There is little motion observed in the videos.
11. **walk-trot :** Walking, not running. The order of footing is different from "*run*" class. The dog goes forward while jumping with the front and back legs in the action of "*run*,". During this action, the dog steps forward with the right and left legs alternately. Hereinafter, called "*walk*."

As examples, the scenes of "*see victim*" and "*stop*" are shown in Fig. 3.

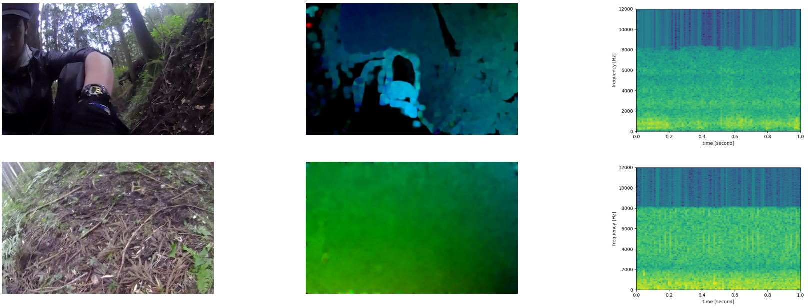

Fig. 3. Examples of the ego-centric rescue-dog video dataset, which shows"*see victim*" class (upper) and "*stop*" class (lower). From the left, RGB images, optical flow images, and the sound images visualized by the MFCC spectrogram.

4 Method

In this study, we perform multi-label classification of rescue-dog activities from ego-centric dog videos, audio and sensor data. Fig. 4 shows a conceptual diagram of our proposed method, which consists of four streams, a sound stream, an appearance stream, a motion stream and a sensor stream. The three streams consist of convolutional layers, the other stream consists of LSTM, and they are integrated by fully-connected (FC) layers in the same as Convolutional Two-Stream Network Fusion [6].

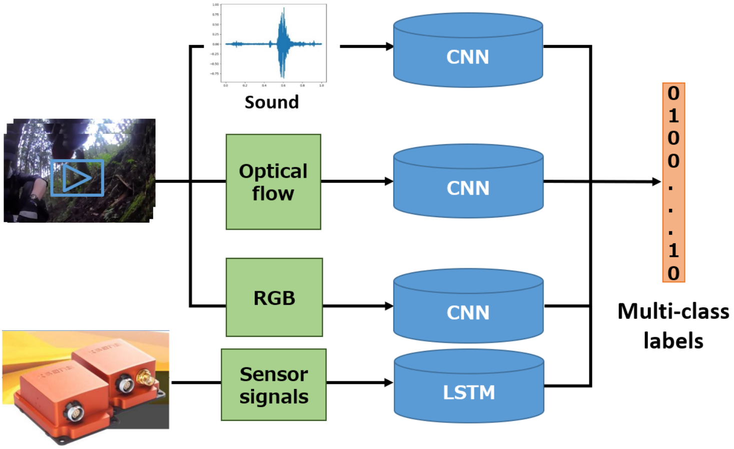

Fig. 4. Outline of the proposed method. We extract multiple information from the input video, and provide them to four different streams. The final output is multi-class labels.

The proposed network which takes two images (appearance RGB images and optical flow images), sounds and sensor data as inputs is called the image/sound/sensor-based four-stream network. The detail is shown in Fig. 5. In this study, this network is used for multi-class action estimation instead of general single-class classification.

First, we take the frame (F_t) of the RGB images out from the input video, and generate an optical flow image (O_t) between the frame, F_t, and the following frame, F_{t+1}. Next, we provide RGB images and optical flow images to the corresponding streams. Regarding sound and sensor data, we set time windows as 1.6 seconds the center of which corresponds to the frame provided to the RGB stream. We obtain log spectrogram with short-time Fourier transform (STFT) (S_t) from the corresponding audio part of the videos (A_t), and we provide it to the sound stream. Regarding sensor data, we used Bidirectional LSTM as the sensor stream. Finally, combine the outputs of these four streams with FC layers, and perform multi-class estimation in the last layer of the network. Action classification is performed for each frame.

In general, SoftMax CrossEntropyLoss is used as the loss function for single-class classification. Since the task of this work is multi-class estimation, SoftMarginLoss is used. Assuming x as an output of the network, y as a target, and C as the numer of the classes, the loss function MultiLabel SoftMarginLoss for multi-class estimation is defined by the following equation:

$$\begin{aligned} loss(x,y) = &-\frac{1}{C} * \Sigma_i(\{y_i\} * log((1+exp(-x_i))^{-1}) \\ &+ (1-y_i) * log(\frac{exp(-x_i)}{1+exp(-x_i)})) \end{aligned} \tag{1}$$

It is designed that the first term in Σ is used if the inferred class is correct, and the second term is used for calculation if it is incorrect. The function changes depending on the inferred label. In this study, there are 11 classes, and the output y is an 11-dimensional binary. The threshold is set to 0.5, and the class above the threshold is set as the estimated class.

4.1 The detail of the image/sound/sensor-based four-stream CNN

Both input images on RGB and optical flow are (224, 224, 3) dimensions. The window size which is the unit size for activity classification is 49 frames, and a RGB image of the center frame and the corresponding optical flow image immediately after a RGB image are taken out. We fine-tune the ImageNet [3] pretrained VGG16 [12] model for both the RGB and optical flow streams. Note that we used the values of L2-norm of optical flow vectors as the third channel inputs of the optical flow stream to make the input of the optical flow stream three channels.

Regarding audio information, we extract short-term Fourier transform (SFFT) spectrogram from the 49 frames which corresponds to 0.8 seconds before and after the center frames of the window, and provide it to the sound stream. The

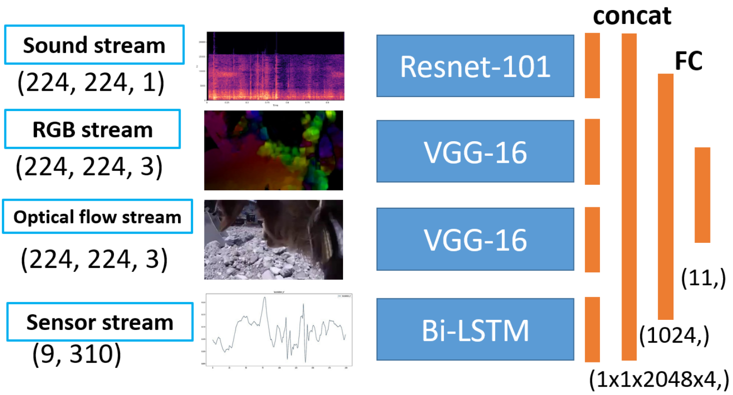

Fig. 5. Proposed method architecture of image/sound/sensor-based four-stream CNN. We provide (224,224,3) dimension images, (224, 224, 1) dimension sound data, and (9, 310) dimension sensor data to each of the four stream, respectively, and get a 11-dimension output as an estimation result.

sound stream consists of pre-trained ResNet-101 with the additional first layer which converts 1-channel to 3-channels, since spectrograms can be regarded as images. We generate a (256, 350, 1) spectrogram image using short-term Fourier transform (SFFT) from the sound data of 49 frames, resize it into (224, 224, 1) with bilinear down-sampling, and provide it into the ResNet-based sound stream. We use short-term Fourier transform (SFFT) by following Arash *et al.* [5] as sound feature representation.

As sensor data, we use acceleration, angular velocity and pose data each of which are 3 dimension with value range normalization ([-1,1]). Since the sensor recording is 200Hz, we obtain 320 points within 1.6 seconds window. To reduce noises, we average them for consecutive 11 frames and finally we obtain a (9, 310) sensor feature. We provide it to Bi-directional LSTM (Bi-LSTM) for feature encoding which is inspired by [8]. In contrast to a standard LSTM which only learns the forward input and makes time series predictions, with a Bi-LSTM, we learn not only in the forward direction but also in the reverse direction to make time series predictions. As a result, the Bi-LSTM can make time series predictions with higher accuracy than the normal LSTM.

The four outputs (each of them is 2048 dimension) obtained from each stream are simply concatenated in the direction of channel, and it is provided to three FC layers. The dimension of the final output of the network is 11-dimensions which is the same as the number of the activity classes annotated to the rescue-dog dataset.

5 Experiments

This section describes the experiment results with the proposed methods and discussions. We made the experiments for seeking for the best feature extraction method of sound and sensor information, and subsequently, we made experiments on the different window sizes with audio and sensor signals. Then, we compared the results of the integration with the optimal networks of sound and sensor information. Note that we used "Sensor data" for all the experiments except for the last experiments.

In all the tables showing experimental results, the accuracy for each class and the total accuracy are represented by Jaccard coefficient. Note that the Jaccard coefficient is represented by

$$\frac{TP}{FP+FN+TP}$$

and a more rigorous value can be obtained compared to the F scale. We used this coefficient to emphasize both Precision and Recall in the rescue-dog's activity estimation. Therefore in this experiments, a model with a larger Jaccard coefficient is expressed as having better accuracy. Note that "-" is displayed because Precision can not be calculated for a class that has never been estimated.

5.1 Selection of Sound Stream Network

In this subsection, we compare VGG16, ResNet-50 and ResNet-101 for the base networks of SFFT spectrogram.

Table 2 shows the results. As results, ResNet-101 with SFFT spectrogram achieved the best result. Therefore, we adopt ResNet-101 with SFFT spectrogram in this work. Sound features were relatively effective for 'bark' and 'stop', since 'bark' is directly related to sound and 'stop' is related to 'no sound' or 'less sound'

Table 2. The results with the different sound networks.(%)

methods	bark	cling	command	eat-drink	handler	run	see-victim	shake	sniff	stop	walk	**ALL**
VGG16	64.29	0.00	**3.58**	0.00	6.51	-	**15.25**	15.25	17.02	**60.63**	70.48	39.44
ResNet-50	62.79	2.17	3.55	0.00	10.77	-	12.28	35.20	18.78	58.61	68.90	40.99
ResNet-101	66.31	**4.80**	1.76	0.00	**11.06**	-	13.49	**50.54**	**19.35**	58.66	**71.90**	**42.43**

5.2 Selection of Sound Window Size

In this study, we compare the performance with various lengths of audio windows. We compare the window size with 1.0, 1.2, 1.4, 1.6, and 2.0 seconds.

Table 3 shows the results, which indicates that 1.6 seconds is the best window size for this rescue-dog dataset.

Table 3. The results with the different time window sizes for sound features.(%)

	bark	cling	command	eat-drink	handler	run	see-victim	shake	sniff	stop	walk	**ALL**
1.0 sec.	66.31	4.80	1.76	0.00	11.06	-	13.49	50.54	19.35	58.66	71.90	42.43
1.2 sec.	65.24	0.00	2.89	0.00	6.48	-	7.845	55.62	14.74	59.91	66.63	38.97
1.4 sec.	79.02	**5.24**	0.25	0.00	12.24	-	**15.43**	21.35	17.44	59.63	67.92	40.57
1.6 sec.	70.51	3.44	**4.19**	0.00	6.37	-	14.96	**82.94**	**21.26**	**63.83**	73.83	**43.74**
2.0 sec.	**77.50**	0.00	0.11	0.00	**12.87**	-	14.14	76.84	11.48	59.73	**74.64**	42.58

5.3 Selection of Sensor Data Network

In the same way as sound networks, we made comparative experiments on different sensor network architectures for encoding sensor data.

Since the sensor data is obtained every 0.005 seconds (200Hz) and three kinds of three dimension sensor signals, we obtain a 200 × 9 feature in one second. In addition to Bidirectional LSTM (Bi-LSTM) [8], we applied a 1D convolutional network to this feature and a 2D convolutional network to a 200 × 9 × 1 by adding a channel direction. We also compare the Bi-LSTM with the standard single directional LSTM.

Table 4 shows the results. Although 2D conv and LSTM achieved almost the same accuracy, the Bi-LSTM achieved the best performance. Therefore, we adopt a Bi-LSTM as an encoder of sensor data in this work. It turned out that sensor information was much less effective than sound on average, although sensor information outperformed sound for 'command', 'handler' and 'see-victim'.

Table 4. The results with the different sensor networks. (%)

	bark	cling	command	eat-drink	handler	run	see-victim	shake	sniff	stop	walk	ALL
1D CNN	0.00	0.00	3.18	0.00	0.00	-	0.00	0.00	5.949	22.95	50.18	24.57
2D CNN	0.00	0.00	**9.28**	0.00	16.21	-	8.84	0.00	16.16	26.85	47.88	25.26
LSTM	0.00	**2.01**	8.25	0.00	13.06	-	**15.97**	9.63	16.41	28.12	48.01	25.84
Bi-LSTM	0.00	1.43	8.88	0.00	**13.18**	-	15.63	**10.93**	**18.29**	**28.03**	**51.46**	**27.31**

5.4 Selection of Sensor Window Size

In the same way as the experiments on the sound window size, we compare the window size with 1.0, 1.2, 1.4, 1.6, and 2.0 seconds using Bi-LSTM. Table 5 shows the results, which indicate that 1.6 seconds achieved the best results as the time window size on the sensor data. Since the best sensor window size is the same window size as the best sound window size, we adopt 1.6 seconds as the time window size for both sound and sensor data.

Table 5. The results with different time window size of sensor features.(%)

	bark	cling	command	eat-drink	handler	run	see-victim	shake	sniff	stop	walk	**ALL**
1.0 sec.	0.00	1.43	8.88	0.00	13.18	-	15.63	10.93	18.29	28.03	51.46	27.31
1.2 sec.	0.00	4.06	8.91	0.00	13.14	-	15.84	11.89	16.70	27.00	**51.98**	27.17
1.4 sec.	0.00	2.48	10.22	0.00	13.89	-	17.39	11.17	17.86	29.52	49.36	27.14
1.6 sec.	0.00	2.47	10.71	0.00	13.94	-	16.45	10.16	16.05	**34.28**	51.12	**28.21**
1.8 sec.	0.00	3.78	**11.25**	0.00	11.79	-	17.30	**13.99**	**17.02**	33.76	48.62	27.60
2.0 sec.	0.00	**6.61**	9.37	0.00	**14.78**	-	**19.93**	2.67	14.76	34.09	47.21	26.82

5.5 Experiments by Integration of All Modalities

In this section, we made experiments by mixing of all streams including RGB, optical flow, sound and sensor data. Note that we used "Sensor data", since our proposed methods require sensor data.

In addition to our full 4-stream model, we prepare the full model using Bidirectional Gated Recurrent Unit (Bi-GRU) [1] instead of Bi-LSTM, since the training cost of GRU is smaller than that of Bi-LSTM. As a baseline, we use the model which used no sensor data. For fair comparison, we prepare our 3-stream model from which the sensor stream is excluded.

Table 6 shows the results with the four models. The proposed model with Bi-LSTM achieved the best accuracy, while the Bi-GRU model achieved the second best with the 0.54 points difference to the best one. From the table, we can see that sensor information was effective for 'sniff' and 'stop' with 8.03 points and 6.80 points improvement between our 3-stream and 4-stream models. From these results, integration of different sensor information is very important to obtain better performance. Although sensor features themselves achieved low performance, by integrating it with video and sound features, it was able to boost overall performance.

Table 6. Comparison with the baseline, ablation model and variant model.(%)

	bark	cling	command	eat-drink	handler	run	see-victim	shake	sniff	stop	walk	**ALL**
2-stream (RGB+opt)	16.60	0.26	**9.07**	0.00	14.30	-	36.68	39.32	14.53	45.29	72.04	41.32
3-stream (no sensor)	62.64	1.17	5.08	0.00	15.74	-	**47.99**	55.68	12.33	57.90	73.42	46.01
4-stream (Bi-LSTM)	73.86	**7.99**	3.29	0.00	14.40	-	41.03	**80.06**	20.36	**64.70**	72.55	**48.05**
4-stream (Bi-GRU)	**78.68**	1.16	3.71	0.00	**18.38**	-	44.77	48.86	**21.49**	59.08	**73.61**	47.51

In addition, we compare our model and the baseline with "Full dataset" which contains no sensor data. Table 7 shows that our modification on sound features improved the result by 10.92 points, which showed the effectiveness of integration of sound and video features. Especially, some actions related to sound, such as 'bark' and 'shake', are improved by 53.52 points and 51.67 points, respectively.

Table 7. Comparison using “Full dataset (no sensor data)”.(%)

	bark	cling	command	eat-drink	handler	run	see-victim	shake	sniff	stop	walk	**ALL**
2-stream (RGB+opt)	11.05	1.82	4.31	0.0	**15.50**	0.0	25.91	0.0	**42.62**	70.53	66.80	43.50
3-stream (no sensor)	**64.57**	**6.25**	**4.58**	0.00	14.45	0.00	**45.96**	**51.67**	12.59	**74.18**	**71.48**	**54.42**

6 Conclusions

We proposed an image/sound/sensor-based four-stream CNN, and estimated the rescue-dog's behavior using the proposed network. From the experimental results, integrating sound and sensor information with ego-centric video was effective for action recognition of rescue dogs. In addition, we examined appropriate window size for sound and sensor data. As results, we found 1.6 seconds were the best window size for the rescue-dog dataset.

There still exists much room to improve feature extraction from dog ego-centric videos. We should consider how to extract more meaningful features. It is possible to add processing specific to dog ego-centric videos like extraction hand features from human ego-centric videos [11]. The dog region segmentation network like the hand segmentation network is applicable to dog activity estimation and expected to contribute to dog activity estimation. However, to do that, we need to create a pixel-level annotated dog ego-centric video dataset.

Increasing the amount of training data is also one of the most important issues. Training data of some actions such as ‘run’ and ’eat-drink’ are much less than the other actions. In addition, the video data annotated with sensor data is also limited. Gathering more data is really needed to enable the proposed method work in the real rescue situation. Furthermore, although out of scope in this time, it is also required for real-time estimation for practical use.

Acknowledgement: This work was supported by JSPS KAKENHI Grant Number 17H06100 and 19H04929.

References

1. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. In: Adavances in Neural Infomatoin Processing Systems Workshop on Deep Learning (2014)
2. Damen, D., Doughty, H., Maria Farinella, G., Fidler, S., Furnari, A., Kazakos, E., Moltisanti, D., Munro, J., Perrett, T., Price, W., Wray, M.: Scaling egocentric vision: The EPIC-KITCHENS dataset. In: Proc. of European Conference on Computer Vision (2018)
3. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: Proc. of IEEE Computer Vision and Pattern Recognition (2009)
4. Ehsani, K., Bagherinezhad, H., Redmon, J., Mottaghi, R., Farhadi, A.: Who let the dogs out? modeling dog behavior from visual data. In: Proc. of IEEE Computer Vision and Pattern Recognition (2018)

5. Evangelos, K., Arsha, N., Andrew, Z., Dima, D.: Epic-Fusion: Audio-visual temporal binding for egocentric action recognition. In: Proc. of IEEE Computer Vision and Pattern Recognition (2019)
6. Feichtenhofer, C., Pinz, A., Zisserman, A.: Convolutional two-stream network fusion for video action recognition. In: Proc. of IEEE Computer Vision and Pattern Recognition (2016), http://arxiv.org/abs/1604.06573
7. Gedas, B., Stella, X. Y.and Hyun, S.P., Jianbo, S.: Am I a baller? basketball skill assessment using first-person cameras. In: Proc. of IEEE International Conference on Computer Vision (2016), http://arxiv.org/abs/1611.05365
8. Graves, A., Mohamed, A., Hinton, G.: Speech recognition with deep recurrent neural networks. In: Proc of IEEE International Conference on Acoustics, Speech and Signal Processing. pp. 6645–6649 (2013)
9. Iwashita, Y., Takamine, A., Kurazume, R., Ryoo, M.S.: First-person animal activity recognition from egocentric videos. In: Proc. of International Conference on Pattern Recognition (ICPR) (2014)
10. Komori, Y., Fujieda, T., Ohno, K., Suzuki, T., Tadokoro, S.: Detection of continuous barking actions from search and rescue dogs' activities data. In: Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 630–635 (2015)
11. Minghuang, M., Haoqi, F., Kris, M.K.: Going deeper into first-person activity recognition. In: Proc. of IEEE Computer Vision and Pattern Recognition (2016), http://www.cs.cmu.edu/ kkitani/pdf/MFK-CVPR2016.pdf
12. Simonyan, K., Vedaldi, A., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: Proc. of International Conference on Learning Representations (2015)
13. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Advances in Neural Information Processing Systems. pp. 568–576 (2014)