https://doi.org/10.20965/jrm.2018.p0513

Paper:

Detection of Target Persons Using Deep Learning and Training Data Generation for Tsukuba Challenge

Yuichi Konishi, Kosuke Shigematsu, Takashi Tsubouchi, and Akihisa Ohya

University of Tsukuba
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan
E-mail: konishi-yu@roboken.cs.tsukuba.ac.jp
[Received March 30, 2018; accepted May 23, 2018]

The Tsukuba Challenge is an open experiment competition held annually since 2007, and wherein the autonomous navigation robots developed by the participants must navigate through an urban setting in which pedestrians and cyclists are present. One of the required tasks in the Tsukuba Challenge from 2013 to 2017 was to search for persons wearing designated clothes within the search area. This is a very difficult task since it is necessary to seek out these persons in an environment that includes regular pedestrians, and wherein the lighting changes easily because of weather conditions. Moreover, the recognition system must have a light computational cost because of the limited performance of the computer that is mounted onto the robot. In this study, we focused on a deep learning method of detecting the target persons in captured images. The developed detection system was expected to achieve high detection performance, even when small-sized input images were used for deep learning. Experiments demonstrated that the proposed system achieved better performance than an existing object detection network. However, because a vast amount of training data is necessary for deep learning, a method of generating training data to be used in the detection of target persons is also discussed in this paper.

Keywords: deep learning, training data generation, object detection, Tsukuba Challenge

1. Introduction

The Tsukuba Challenge is an open experiment competition held annually since 2007, wherein autonomous navigation robots developed by the participants must navigate through an urban setting in which pedestrians and cyclists are present. The robots are required to navigate autonomously over a course with a length of approximately 2 km, and accomplish various tasks during their navigation. One of the tasks set in the 2017 Tsukuba Challenge was to recognize four persons that wore designated clothes and sat on chairs beside a sign board. An example is shown in **Fig. 1**. The records of the 2017 event indicate that among the 65 participating robots, only three were successful in locating all four target persons, and only one of them did so without making an erroneous recognition. This indicates that recognizing the target persons is an extremely difficult task because the robot must periodically carry out recognition processing at short intervals with the limited computational resources that it can carry as it navigates the course, and also because the energy required for computation must be minimized to allow the robot to navigate a long distance. Therefore, the recognition algorithm must have low computational cost.

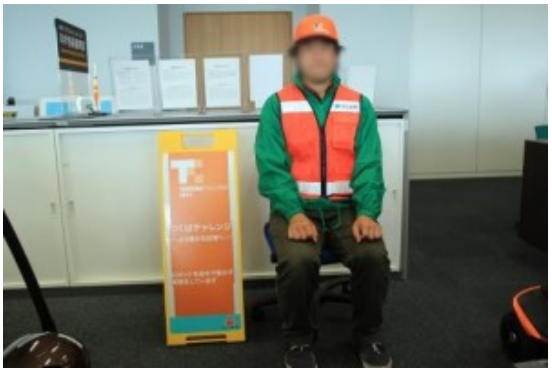

Fig. 1. Example of target person.

The objective of this study was to determine whether a target person existed in the images captured by the robot, and whenever such a person was identified, to compute their position in the image. If the position of the target person was identified in the image, it would be possible to determine their three-dimensional position by additionally using a stereo camera or laser range finder (LRF). Our goal was to achieve robust image recognition by applying deep learning, which has recently attracted attention in the machine learning field, and develop an algorithm with low computational cost that would allow the robot to carry out recognition in sufficiently short periods. Moreover, we compared our method with DetectNet [a], which is an existing object detection network. It is known that deep learning requires a large amount of training data to achieve good recognition performance. However, it is difficult to collect a sufficient amount of data during the limited time period of the Tsukuba Challenge. When the

amount of training data obtained by machine learning is not sufficient, additional data are produced by image synthesis and processing. In this study, we also investigated a method of producing training data for use in deep learning without using a vast number of images. Additionally, we assessed the effectiveness of various image processing methods.

By using the proposed target person recognition method and a training data generation method, we were able to develop a recognition system that executed computations at a sufficient speed by utilizing a CPU or graphics processing unit (GPU), which can be mounted onto a robot and achieve good recognition performance.

This paper is structured as follows. Section 2 reviews related studies. Section 3 describes the proposed target person detection method. Section 4 describes the training data generation method. Section 5 presents a comparison of the proposed detection method with existing object detection networks. This comparison demonstrates the validity of the training data generation method. The conclusions drawn by this study are summarized in Section 6.

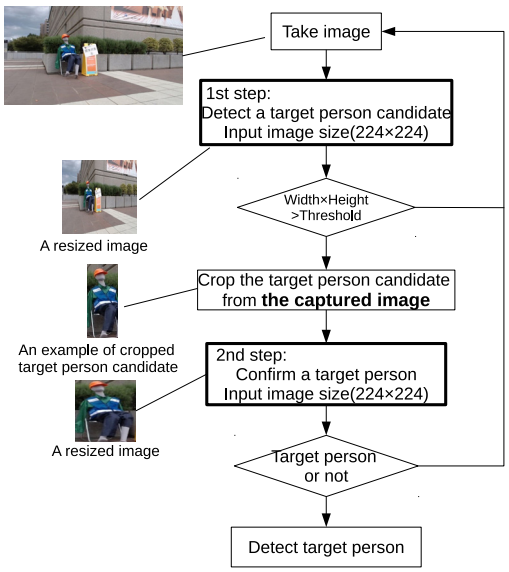

Fig. 2. Flow of target person detection.

2. Related Studies

The task of recognizing target persons was introduced in the 2013 Tsukuba Challenge, and has been thus far tackled by many robots. In the 2013 to 2015 Tsukuba Challenge, various studies employed human recognition algorithms based on color extraction [1–3]. These recognition methods had the shortcoming of being easily affected by weather changes, and thus lacked robustness. Various studies have used LRFs to detect the sign boards and recognize the target persons [4, 5]. Reflective tape was adhered to the sign boards in the 2013 to 2016 Tsukuba Challenge, and this has made it easy to recognize them by the intensity of received light recorded by the LRF. However, reflective tape was not used in the 2017 event, and this made it difficult to detect the signs by using LRFs.

In recent years, convolutional neural networks (CNNs) have been used for image processing and were found to be quite effective. In the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), which is a competition in general object recognition, AlexNet [6], GoogLeNet [7], and ResNet [8], which are CNNs, won the competition in 2012, 2014, and 2015, respectively. Moreover, the object recognition networks DetectNet and Yolo [9] have been proposed. Consequently, robots incorporating deep learning began to appear in the 2016 and 2017 Tsukuba Challenge. Mitsudome et al. [10] used an LRF to extract the target person candidate areas in the images, and then applied GoogLeNet to classify and successfully locate a target person. In this study, 89,252 images, which were collected during the preliminary and main runs from the 2016 and 2017 Tsukuba Challenge, were used as the basis of producing an extended data set. Some issues consisted of the need to collect a vast amount of images, and having to spend many hours of labor on labeling the images. Various studies used object detection networks to recognize the target persons [11, 12]. However, none of these studies conducted evaluation tests to sufficiently verify the detection performance. This study proposes a method of recognizing the target persons by using a CNN, and compares it with object detection networks for evaluation. Data extension has been used to increase the training data by means of image processing. Additionally, an existing study [13] has generated training data by image synthesis and processing. This study proposes a method to generate training data for the recognition of target persons, and assesses its validity. We believe that this will provide useful information for future target person recognition systems based on deep learning.

3. Proposed Method to Recognize Target Persons Based on CNN

3.1. Outline of Method to Detect Target Persons

The objective of the proposed method of detecting target persons is to achieve rapid computation and high-performance detection with a computer that can be mounted onto autonomous navigation robots. One approach towards making the computational load of deep learning lighter is to reduce the size of the images used as input to the CNN. However, a reduction of the input image will result in estimation using unclear images, which may lower the detection performance. This paper proposes a method that achieves high detection performance even when the input images are small.

The process flow of the proposed method of recognizing the target persons is shown in **Fig. 2**. This method involves two steps: a process to detect the candidate areas of target persons, and a confirmation process. First, the camera image captured from the robot is resized to

Fig. 3. Definition of x, y, width, and height.

224×224 pixels, which is used as the input for the detection of candidate areas. This size is used because it is the default input size used by GoogLeNet. In the process of detecting candidate areas, a CNN is used to detect the presence or absence of the target persons in the image. If target persons are present, the CNN estimates the x and y coordinates, width, and height, which represent the bounding box of the target person. These variables are defined in **Fig. 3**. Next, the area in the original camera image corresponding to this bounding box is extracted, resized to 224×224 pixels, and used as the input image to confirm the target persons. The confirmation process carries out a binary classification of the image into "target person present" and "target person not present." The key feature here is that an image extracted from the original camera image is used as input to confirm the presence of a target person. Thereby, we can expect that the detection performance will improve in comparison with the resizing of the entire original image to 224×224 pixels, because the presence of the target persons is determined at a high resolution.

3.2. Detection of Candidate Area for Target Person (First Step)

Although the target persons are distributed in the target search area, they are located apart from each other. Therefore, we consider a method of detecting a single person, at most, by assuming that there will not be any cases wherein two or more target persons will exist in a single image captured by the robot. To detect a candidate area for a target person, the above four variables (x, y, width, and height) are estimated as a regression problem using GoogLeNet. It was thought that high performance can be expected by using a network (GoogLeNet, AlexNet, ResNet, etc.) that has scored well in the ILSVRC. GoogLeNet was used because it has deeper network layers than AlexNet; therefore, high performance can be expected. Moreover, ResNet, which has very deep layers and better accuracy than GoogLeNet, was considered as unsuitable because it would require considerable estimation time when installed to a robot. For training, pairs consisting of images that contain a target person and the four variables (x, y, width, and height) representing the position and size of the bounding box surrounding that person, and pairs consisting only of background images (without the target person) and the four variables (0, 0, 0, 0) were provided as training data. The background image was paired with the four variables (0, 0, 0, 0) with the expectation that the output values of width and height would be low when there is no target person in the image. Based on this information, the system will be able to determine if a target person is present. When the width and height are greater than the given thresholds, it is assessed that a candidate area for a target person has been detected, and the corresponding bounding box is selected as the candidate area.

It is known that deep learning requires a vast amount of training data. Therefore, the images produced by the training data generation method, which is described in Subsection 4.1, are used as the training data.

3.3. Confirmation Processing of Target Person (Second Step)

The objective of the target person confirmation process, which is the second step in the proposed method, is to check whether the image detected as the candidate area for a target person is in fact a target person. Its purpose is to reduce the rate of erroneous detection. An image of the candidate area that is extracted from the original camera image is used as the input such that the number of pixels representing the target person is greater than the number of pixels in the image used to detect the candidate area. This measure was adopted to improve the detection accuracy. The confirmation process is carried out by binary classification using GoogLeNet, where the probabilities of the two classes are output. The output probability of a target person is considered to be the likelihood of detecting a target person. When used in an actual search by a robot, a threshold is set for the probability of a target person, to determine whether the image is in fact a target person. If the threshold is exceeded, the detection of a target person is confirmed.

The training data must also include the images remaining after the target person has been cropped, and the images of other items (in addition to target persons) such as pedestrians, cyclists, backgrounds, etc. Because deep learning requires a vast number of such images, these images are produced by the training data generation method described in Subsection 4.2 and used as training data.

4. Method of Generating Training Data for Target Person Detection

Because deep learning requires learning a vast number of parameters, training data that consist of over than 10,000 images captured in various environments are necessary. To collect a very large body of training data by actually taking photographs, one must place the target person in various locations within the search area under various weather conditions, and in a setting wherein pedestrians are present, which is unrealistic. Moreover, many labor hours are necessary to manually apply target person

Fig. 4. Original images used to generate training images.

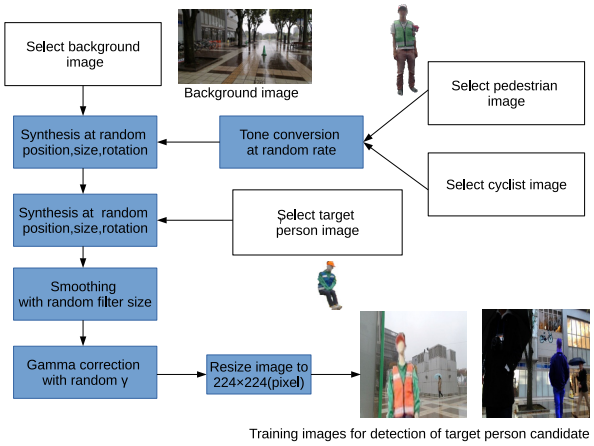

Fig. 5. Flow of generating training image used for detection of target person candidate.

bounding boxes or label the images according to whether a target person is present or not. Thus, there are methods to recreate the learning environment by image synthesis or image processing, and use it as the training data in deep learning. This paper proposes a method of generating training data that is suitable to the detection of the target persons.

4.1. Generation of Training Data for Detection Processing of Candidate Target Person

The arbitrary placement of target persons within the search area, the movement of pedestrians and cyclists, and the variations in the lighting conditions, are simulated by image processing, and used as the training data. In the proposed training data generation method, multiple images of backgrounds that do not include target persons, images of target persons, and images of pedestrians and cyclists are used as the original images. In this study, 9,842 photos that were captured during the two preliminary runs in the 2016 event were used as the background images. Additionally, 44 photos of target persons, 23 photos of pedestrians, and five photos of cyclists were collected from preliminary runs and the internet. Because the target persons were assumed to wear orange or blue vests, according to the Tsukuba Challenge rules, the color of the vest was modified by changing the hue of the HSV model to increase the number of target person images.

Examples of the original images are shown in **Fig. 4**. The images of target persons, pedestrians, and cyclists, were manually cropped to extract the final images.

Figure 5 shows the procedural flow to generate a single image for the detection of target person candidate areas. First, a background image is chosen. Next, images of pedestrians and cyclists are subjected to random tone conversion and synthesized with the background image by placing them in random positions and sizes. In this case, tone conversion means the multiplication of channels and pixel values by a real number to change the color tone.

The image of a target person is similarly synthesized with a background image at a random position and size. The synthesized image is then subjected to gamma correction by using a random gamma value, and is smoothed by using a median filter with a random filter size. Finally, the image is resized. The processing parameters and ranges of random values are presented in **Table 1**. The size of the background image is 864×480 pixels. Examples of generated training images for detection of target person candidate are shown in **Fig. 6**.

This process is repeated until the necessary number of images required as training data is obtained. By using the images generated in this manner as training data, it was hoped that the target persons could be detected without being greatly affected by the target person placement variations, lighting resulting from weather conditions, and other detailed features.

4.2. Method of Generating Training Data for Confirmation of Target Person

In the confirmation of target persons, the image obtained by the detection of candidate target persons is checked and classified as an image that represents a target person or one that does not. Thus, we must generate training data that simulate the detected images. This process is shown in **Fig. 7**. The area representing the target person is extracted from the image obtained by the method described in Subsection 4.1 before that image is resized. To prevent the confirmation process from being affected by the candidate detection process, the images are cropped by applying random values to produce smaller images. Thus, the image of the target person is cropped by extracting 0.6–1 and 0.4–1 of the width and height, respectively. Then, these images are used as training images of the target persons in the target person confirmation process. Moreover, the background images used as training data in the confirmation process are produced by cropping images of random size and position from the images whose

Table 1. Show parameter of processing and parameter range of random.

Process	Parameters	Range of random selection	Comment
Synthesize of target person	x	0∼750 pixel	
	y	0∼300 pixel	
	Width	50∼300 pixel	
	Height		Calculate based on aspect ratio
	Rotation	$-10^\circ \sim 10^\circ$	
Synthesize of pedestrian	x	0∼810 pixel	
	y	0∼440 pixel	
	Width	50∼250 pixel	
	Height		Calculate based on aspect ratio
Synthesize of cyclist	x	0∼720 pixel	
	y	0∼720 pixel	
	Width	100∼200 pixel	
	Height		Calculate based on aspect ratio
Tone conversion	Expansion of Red	0.5∼1.5	
	Expansion of Green	0.5∼1.5	
	Expansion of Blue	0.5∼1.5	
Gamma correction	γ	0.5∼2.5	Process with 33% probability
Smoothing	Filter size	1∼5	Process with 25% probability

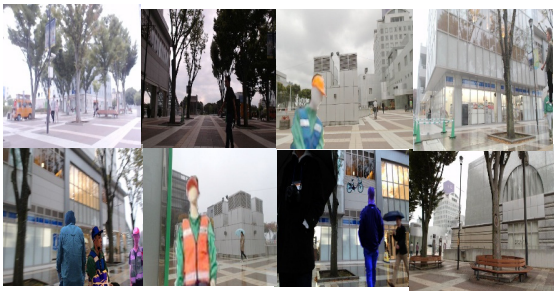

Fig. 6. Examples of generated training images for detection of target person candidate.

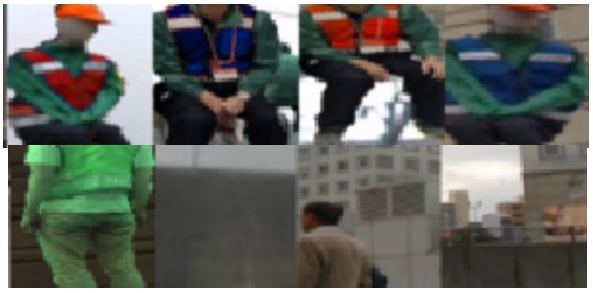

Fig. 8. Examples of generated training images used in confirmation of target person.

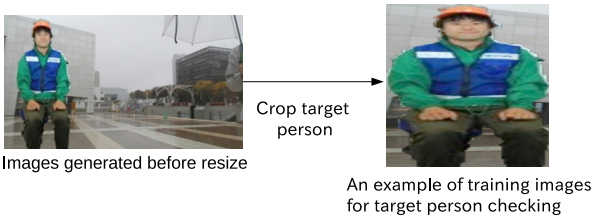

Fig. 7. Flow of generating training images for confirmation of target person.

synthesis was described in Subsection 4.1 and which do not include a target person. Examples of the produced images are shown in **Fig. 8**.

5. Experiment

In this study, we compared DetectNet, which is an existing object detection network, and the proposed two-step target person detection method using GoogLeNet and AlexNet, with the aim of verifying the usefulness of the proposed method. In Subsection 5.1, we discuss our investigation of a suitable number of iterations to train GoogLeNet, AlexNet, and DetectNet. For each network, we determined the number of iterations that was necessary for the performance to converge (or peak). The networks' performances were then compared by using the obtained number of iterations. The results are presented in Subsection 5.2.

5.1. Examination of Training Cycles of GoogLeNet, AlexNet, and DetectNet

5.1.1. Experimental Method

By using the training data generation method described in Subsection 4.1, two sets, each consisting of 80,000 images (image size of 224×224 pixels), were generated: one for the candidate target person detection process, and another one for the confirmation process. The correct label for the candidate detection training data was defined as the bounding box of the target person and consisted of 864×480 pixels. The detection images of the candidate target persons were used to train DetectNet. 75% of each data set was used as training data, while the remaining 25% was used as validation data. A suitable number of iterations was considered according to whether

the accuracy or loss or mean average precision (mAP), which was obtained with the validation data, converged sufficiently. Training was carried out by using NVIDIA DIGITS, which is a training tool. The training for the detection of candidate target persons using GoogLeNet and AlexNet was carried out for 30, 60, 100, 200, 300, 400, 500, 600, and 700 epochs, and the losses obtained from the validation data were determined for GoogLeNet and AlexNet. The losses were computed by using the DIGITS Euclidean Loss Layer. With regard to the training parameters, the learning rate was initially set to 1×10^{-7}, and was further multiplied by 1×10^{-1} when the iterations reached $1/3$ and $2/3$ of the total number of iterations. For example, when the training involved 30 epochs, the learning rate was reduced at epoch 10 and epoch 20; when the training involved 600 epochs, the learning rate was reduced at epoch 200 and epoch 400. This is the default learning rate schedule in DIGITS. Stochastic gradient descent (SGD) was used to optimize the neural network.

DetectNet was trained by using 30, 60, and 100 epochs, after which the mAP was determined against the validation data. As the training parameter, the learning rate was initially set to 1×10^{-4}, which was further multiplied by 1×10^{-1} when the iterations reached $1/3$ and $2/3$ of the total number of iterations. The Adam algorithm was used to optimize the neural network. The mAP was computed by using the DIGITS mAP Layer.

With regard to the target person confirmation process, GoogLeNet and AlexNet were trained by using 20, 30, 60, and 90 epochs, and the accuracy against the validation data was determined. Thus, 4×2 training models were generated. As the training parameter, the learning rate was initially set to 1×10^{-2}, and multiplied by 1×10^{-1} when the iterations reached $1/3$ and $2/3$ of the total number of iterations. Moreover, SGD was used to optimize the neural network. Loss was computed by using DIGITS Softmax with the Loss Layer.

5.1.2. Experiment Results

The losses obtained from the validation data when GoogLeNet and AlexNet were trained in the detection of candidate target persons, and the mAP of the validation data using DetectNet are presented in **Table 2**. With GoogLeNet, it was found that the loss decreased as the number of iterations increased up to 500 epochs, but did not decrease further beyond that point. Thus, 500 epochs were sufficient to train GoogLeNet in carrying out the detection of candidate target persons. Similarly, 300 epochs were considered sufficient to train AlexNet. Moreover, 100 epochs were considered sufficient to train DetectNet because mAP displayed only slight variations.

The accuracies of the validation data when GoogLeNet and AlexNet were trained in the confirmation of the target persons are presented in **Table 3**. The accuracy of GoogLeNet did not increase beyond 60 epochs; thus, this was considered sufficient for training. Similarly, 60 epochs were also considered sufficient for training AlexNet. This number of iterations was used in the experiment described below.

Table 2. Result of training for various numbers of iterations.

Epoch	GoogLetNet (loss)	AlexNet (loss)	DetectNet (mAP)
5	–	–	45.762
10	–	–	45.917
30	821.749	1764.77	46.0066
60	626.097	1744.63	46.0083
100	475.252	1570.5	46.0029
200	477.602	1521.99	46.0166
300	522.35	1422.39	46.0266
400	486.597	1539.85	46.0116
500	411.166	1496.03	–
600	429.601	1465.8	–
700	414.355	1486.99	–

Table 3. Training results for various number of iterations.

Epoch	GoogLetNet (Accuracy)	AlexNet (Accuracy)
20	99.965	95.95
30	99.98	99.97
60	99.985	99.985
90	99.985	96.965

Table 4. Six patterns of training datasets.

Dataset	Gamma correction	Synthesis of pedestrians/cyclists	Smoothing
Dataset 1	Enable	Enable	Enable
Dataset 2	Enable	Disable	Enable
Dataset 3	Disable	Enable	Enable
Dataset 4	Disable	Disable	Enable
Dataset 5	Enable	Enable	Disable
Dataset 6	Disable	Disable	Disable

5.2. Confirmation of Validity of Training Data Generation Method and Comparison of Target Person Detection Performance

In this subsection, we discuss the validity of the proposed training data generation method. The effects of the gamma correction, synthesis of pedestrians and cyclists, and smoothing in the process of generating the training data, were compared. To investigate whether the detection performance varied with the network, the two-step method using GoogLeNet or AlexNet was compared with that using DetectNet. Moreover, the detection performance and computational speed of the proposed two-step method to detect the target persons using GoogLeNet were evaluated and compared with the two-step target person detection method using AlexNet or DetectNet, which is an existing object detection network.

5.2.1. Experimental Method

Six sets of training data images were generated based on the combination of whether gamma correction was carried out or not, whether images of pedestrians or cyclists were incorporated or not, and whether smoothing was carried out or not, as shown in **Table 4**. Ideally, the eight possible combinations should all be compared to ensure that they are thorough. However, in some cases, training one network required an entire day with our environment (two

Fig. 9. Examples of test images.

GeForce GTX 1080Ti GPUs), and the limited amount of available time made it difficult to carry out all cases. Therefore, it was decided to consider this as an issue open for future investigation and proceed with the experiment by using the abovementioned six sets. For each combination, 80,000 images were generated for the detection of candidate target persons, and another 80,000 images were generated for the confirmation of target persons. Thus, 75% of each data set was used as the training data, while the remaining 25% was used as the validation data. Because the neural network's initial values, or the manner in which convergence occurs with training, is not the same, even when the same training data is used, and results in varied recognition performance, GoogLeNet, AlexNet, and DetectNet were each trained three times. Thus, they each generated three models in the training for the confirmation process of the target persons. In the detection of the candidate target persons, GoogLeNet and AlexNet were each trained once, and produced a single model in each case because the training in this case was time consuming. The number of iterations and training parameters used were those discussed in Subsection 5.1. As the test images, 511 images of target persons and 8,659 images, which did not include the target persons, captured during the preliminary and main runs in the Tsukuba Challenge were used. Examples of the test images are shown in **Fig. 9**. **Table 5** presents the dates when the test images were taken, the weather, and the image numbers. The average precision (AP) was used as the evaluation index [b] relative to the evaluation index of the object detection section of the Pascal visual object classes (VOC). AP is the area under the curve (AUC) of the precision-recall curve. The precision-recall curve is plotted by varying the probability threshold or detection reliability. The point in the precision-recall curve with the highest recall is where the threshold is zero, and is equivalent to omitting the process of confirming the target persons, i.e., the second step, in the proposed two-step detection method. In this study, the AP, or AUC, was approximated by Eq. (1). $P(r)$ is the Precision function with Recall as the variable.

$$\begin{cases} AP = \dfrac{1}{101} \displaystyle\sum_{r\in\{0.01,0.02,\ldots,0.99,1\}} P_{interp}(r) \\ P_{interp}(r) = \displaystyle\max_{\tilde{r}:\tilde{r}\geq r} P(\tilde{r}) \end{cases} \quad (1)$$

Table 5. Test image information.

Date	Weather	in target person	Number of images
2016/10/29	Cloudy	Yes	133
2016/11/4	Sunny	Yes	66
2016/11/5	Sunny	Yes	111
2016/11/6	Cloudy	Yes	22
2017/11/5	Sunny	Yes	179
2017/11/5	Sunny	No	8659
Total			9170

Table 6. Result of GoogLeNet (two-step).

Dataset	Model 1	Model 2	Model 3	Avg. AP
Dataset 1	0.786	0.795	0.794	0.791
Dataset 2	0.780	0.778	0.769	0.775
Dataset 3	0.703	0.709	0.704	0.705
Dataset 4	0.707	0.698	0.701	0.702
Dataset 5	0.772	0.770	0.767	0.769
Dataset 6	0.710	0.723	0.713	0.715

A high AP value indicates that the detection performance is good. The APs of the two-step method using GoogLeNet and AlexNet were computed, and the AP of DetectNet, which can detect multiple objects, was computed from the detected case with the highest reliability. A successful detection is defined as one that has taken place when the intersection over union (IoU), which represents the overlap between the true and detected bounding boxes, is 0.33 or greater.

Next, we compared the computational speed. The computer used in the estimation consisted of the CPU (Core i7 7700k), GPU (Geforce GTX1080), and JetsonTX2 (NVIDIA). The time for the detection of the candidate target persons, which is the first step in the two-step method, and the time when the confirmation of the target persons was carried out, i.e., the second step, were measured. The detection time when DetectNet was used was also measured.

5.2.2. Experiment Results

The APs for the different data sets using the various methods are presented in **Tables 6–8**. The APs for the six patterns of the three models in each method and their average APs are shown. The precision-recall curves of the models with the highest APs for each training data set and method are shown in **Figs. 10–12**. **Table 6** presents the results of the proposed two-step method when GoogLeNet was used. Since the highest AP was obtained by using Dataset 1, the method of generating the training data can be considered as valid. From the results of Datasets 3, 4, and 6, we can see that the AP tended to be low when gamma correction was not carried out. This indicates that gamma correction is a valid process when GoogLeNet is used. Moreover, Datasets 3 and 4 produced a lower AP than Dataset 6. This shows that the synthesis of pedestrians/cyclists or the smoothing process can lead to poorer detection performance. Because Dataset 1, wherein all of

Table 7. Result of AlexNet (two-step).

Dataset	Model 1	Model 2	Model 3	Avg. AP
Dataset 1	0.734	0.737	0.739	0.736
Dataset 2	0.759	0.760	0.756	0.758
Dataset 3	0.737	0.743	0.743	0.741
Dataset 4	0.722	0.714	0.717	0.717
Dataset 5	0.731	0.728	0.730	0.729
Dataset 6	0.691	0.688	0.684	0.687

Table 8. Result of DetectNet.

Dataset	Model 1	Model 2	Model 3	Avg. AP
Dataset 1	0.759	0.775	0.731	0.755
Dataset 2	0.733	0.736	0.744	0.737
Dataset 3	0.705	0.650	0.635	0.663
Dataset 4	0.603	0.597	0.611	0.603
Dataset 5	0.781	0.746	0.762	0.763
Dataset 6	0.575	0.593	0.574	0.580

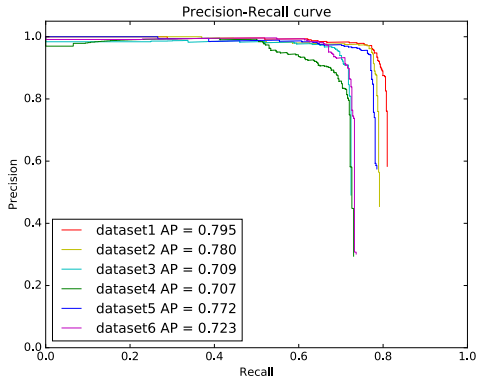

Fig. 10. Precision-recall curves for six training datasets (GoogLeNet (two-step)).

the processing techniques were carried out, produced the highest AP, this suggests that the detection performance varied depending on the compatibility of image processing technique combinations. Next, **Table 7** presents the results of the two-step method when AlexNet was used. The results reveal a different trend in comparison with the case where GoogLeNet was used. The highest AP was obtained when Dataset 2, wherein the images of pedestrians/cyclists were not incorporated, was used for training. This suggests that a suitable method to generate training data was different for different networks. **Table 8** presents the results when DetectNet was used. It can be seen that the highest AP was obtained when Dataset 5 was used. This means that the network performed better when smoothing was not applied.

The results presented above indicate that, in the two-step method using GoogLeNet, the detection performance improved when training data was generated by using gamma correction, pedestrian/cyclist synthesis, and smoothing. Moreover, it was shown that the best method to generate training data may differ depending on the type of network. However, it is not clear why this occurred, and it remains an open issue to be investigated in future work.

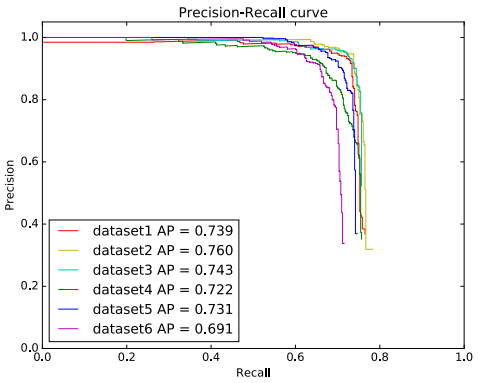

Fig. 11. Precision-recall curves for six training datasets (AlexNet (two-step)).

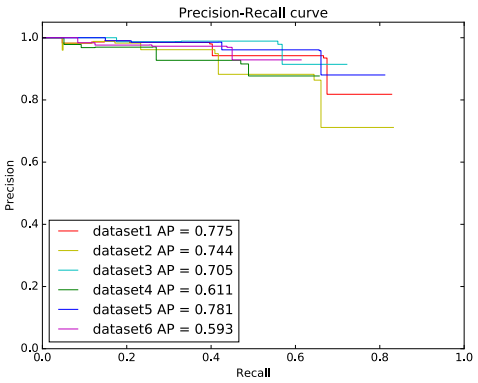

Fig. 12. Precision-recall curves for six training datasets (DetectNet).

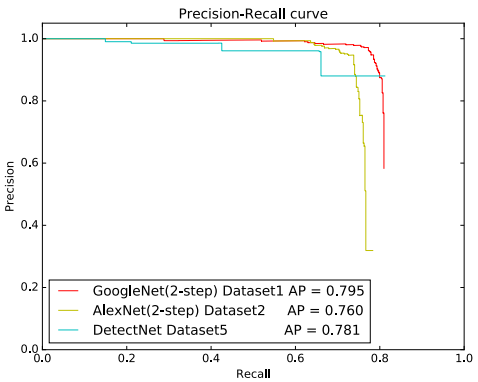

Fig. 13. Precision-recall curves of three method obtained from best training dataset.

The precision-recall curves when the highest AP was obtained with the three methods are shown in **Fig. 13**, where it can be seen that the two-step method using GoogLeNet trained with Dataset 1 produced the best detection performance.

Table 9 presents the computational speed results. The computational speeds of GoogLeNet (first step only) and DetectNet were approximately the same. The majority of the images captured when the robot searched for tar-

Table 9. Computational speeds of three methods.

Method	CPU (core i7 7700k)	GPU (Geforce GTX1080)	JetsonTX2
GoogLeNet (First step only) [ms]	459.5	22.3	60.5
GoogLeNet (Two-steps) [ms]	821.4	45.5	115.7
AlexNet (First step only) [ms]	229.1	8.9	48.8
AlexNet (Two-steps) [ms]	462.9	16.7	89.9
DetectNet [ms]	507.4	22.3	52.4

get persons did not contain a target person. In this case, the computational time for the two-step method consisted mostly of the time spent on the first step. Thus, the average computation time when the robot conducted a search would be approximately the same for GoogLeNet and DetectNet. The fastest computational speed was obtained with AlexNet, which had a shallow layer.

In summary, the two-step method that used GoogLeNet trained with the training data produced by applying gamma correction, synthesis of pedestrians/cyclists, and smoothing, achieved approximately the same speed as DetectNet, and produced a high detection performance, which is sufficient for being considered as a valid method in the Tsukuba Challenge.

6. Conclusion

In this paper, we proposed a method of detecting target persons based on the use of CNNs, and a method of generating training data. It was found that the proposed two-step detection method based on the use of GoogLeNet performed better than DetectNet, which is an existing object detection network for detecting target persons. With regard to our proposed method of training data generation, we investigated a suitable type of image processing to detect target persons, and verified the validity of our method. Moreover, we investigated in what way the detection performance is affected by different methods of generating training data. This information can be useful in efforts to detect target persons based on CNNs.

We note various issues that require further investigation. In the present method of training data generation, the target person is synthesized with a background image at random positions. At times, this produces images wherein the target person is positioned at locations where they should not be. Thus, it is necessary to investigate whether the detection performance can be improved by synthesizing the target person only at locations where they can actually exist. We plan to generate a more natural and realistic image by identifying the ground section through networks capable of image segmentation [14] and synthesizing the target person to that section, or by using networks that generate images [15]. Moreover, the fact that a proper method of generating training data differed with different networks is an issue that merits further investigation.

Acknowledgements

The authors would like to express their gratitude to Kiyoshi Suwabe, who was a member of Team Kenaf in the Tsukuba Challenge 2016, and to Masaru Imai and Yuta Yamasaki, who were members of Team MASARU in the Tsukuba Challenge 2017.

References:

[1] K. Yamauchi, N. Akai, R. Unai, K. Inoue, and K. Ozaki, "Person Detection Method Based on Color Layout in Real World Robot Challenge 2013," J. Robot. Mechatron., Vol.26, No.2, pp. 151-157, 2014.

[2] Y. Kanuki and N. Ohta, "Development of Autonomous Robot with Simple Navigation System for Tsukuba Challenge 2015," J. Robot. Mechatron., Vol.28, No.4, pp. 432-440, 2016.

[3] S. Akimoto, T. Takahashi, M. Suzuki, Y. Arai, and S. Aoyagi, "Human Detection by Fourier Descriptors and Fuzzy Color Histograms with Fuzzy c-Means Method," J. Robot. Mechatron., Vol.28, No.4, pp. 491-499, 2016.

[4] J. Eguchi and K. Ozaki, "Development of the Autonomous Mobile Robot for Target-Searching in Urban Areas in the Tsukuba Challenge 2013," J. Robot. Mechatron., Vol.26, No.2, pp. 166-176, 2014.

[5] K. Hosaka and T. Tomizawa, "A Person Detection Method Using 3D Laser Scanner," J. Robot. Mechatron., Vol.27, No.4, pp. 374-381, 2015.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," Advances in neural information processing systems, pp. 1097-1105, 2012.

[7] C. Szegedy et al., "Going deeper with convolutions," Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1-9, 2015.

[8] K. He et al., "Deep residual learning for image recognition," Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, pp. 770-778, 2016.

[9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, pp. 779-788, 2016.

[10] R. Mitsudome, H. Date, A. Suzuki, T. Tsubouchi, and A. Ohya, "Autonomous Mobile Robot Searching for Persons with Specific Clothing on Urban Walkway," J. Robot. Mechatron., Vol.29, No.4, pp. 649-659, 2017.

[11] S. Bando, T. Nakabayashi, S. Kawamoto, and H. Bando, "Approach of Tsuchiura Project in Tsukuba Challenge 2016," Proc. of the 17th SICE SI Division Annual Conf., pp. 1392-1397, 2016 (in Japanese).

[12] H. Hachiya, Y. Saito, K. Iteya, and T. Nakamura, "Perspective anchors for a specific object detection and its distance measurement," Proc. of the 18th SICE SI Division Annual Conf., pp. 1952-1954, 2017 (in Japanese).

[13] G. Georgakis, A. Mousavian, A. C. Berg, and J. Kosecka, "Synthesizing Training Data for Object Detection in Indoor Scenes," Robotics: Science and Systems (RSS) 2017, 2017.
[14] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.39, No.12, pp. 2481-2495, 2017.
[15] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," 4th Int. Conf. on Learning Representations (ICLR 2016), 2016.

Supporting Online Materials:

[a] DetectNet: Deep Neural Network for Object Detection in DIGITS, https://devblogs.nvidia.com/detectnet-deep-neural-network-object-detection-digits/ [Accessed March 3, 2018]
[b] The PASCAL Visual Object Classes Homepage, http://host.robots.ox.ac.uk/pascal/VOC/ [Accessed March 3, 2018]

Name:
Yuichi Konishi

Affiliation:
Department of Computer Science, Graduate School of Systems and Information Engineering, University of Tsukuba

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
2017 Received Master degree in Engineering from University of Tsukuba

Name:
Kosuke Shigematsu

Affiliation:
Department of Intelligent Interaction Technologies, Graduate School of Systems and Information Engineering, University of Tsukuba

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
2017 Received Ph.D. in Engineering from University of Tsukuba

Name:
Takashi Tsubouchi

Affiliation:
Professor, Faculty of Engineering, Information and Systems, University of Tsukuba

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
1988 Received Ph.D. from Department of Electronics and Information Engineering, University of Tsukuba
1989-1993 Assistant Professor, Utsunomiya University and The University of Tokyo
1994-2006 Lecturer and Associate Professor, University of Tsukuba
2006- Professor, University of Tsukuba

Main Works:
• H. Kawanishi, Y. Hara, T. Tsubouchi et al., "Calibration of Lens Distortion for Super-Wide-Angle Stereo Vision," 2015 IEEE Int. Conf. on Automation Science and Engineering, pp. 843-848, August 2015.
• Y. Hara, S. Bando, T. Tsubouchi, and A. Oshima, "6DOF Iterative Closest Point Matching Considering A Priori with Maximum A Posteriori Estimation," IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS2013), pp. 4172-4179, November 2013.
• S. Bando, T. Tsubouchi, and S. Yuta, "Scan matching method using projection in dominant direction of indoor environment," Advanced Robotics, Vol.28, No.18, pp. 1243-1251, 2014.

Membership in Academic Societies:
• The Institute of Electrical and Electronics Engineers (IEEE)
• The Robotics Society of Japan (RSJ)
• The Japan Society of Mechanical Engineers (JSME)

Name:
Akihisa Ohya

Affiliation:
Professor, Department of Information Engineering, Faculty of Engineering, Information and Systems, University of Tsukuba

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
1992-1994 Research Associate, University of Tsukuba
1994-2000 Assistant Professor, University of Tsukuba
1995-1997 Visiting Scholar, Purdue University
2000-2012 Associate Professor, University of Tsukuba
2000-2003 Researcher, PRESTO, JST
2012- Professor, University of Tsukuba

Main Works:
• "Development of Small Size 3D LIDAR," 2014 IEEE Int. Conf. on Robotics and Automation, pp. 4620-4626, May 2014.
• "Image Correspondence Based on Interest Point Correlation in Difference Streams: Method and Applications to Mobile Robot Localization," J. Robot. Mechatron., Vol.28, No.2, pp. 234-241, April 2016.

Membership in Academic Societies:
• The Institute of Electrical and Electronic Engineers (IEEE)
• The Robotics Society of Japan (RSJ)
• The Society of Instrument and Control Engineers (SICE)
• The Japan Society of Mechanical Engineering (JSME)
• The Institute of Electronics, Information and Communication Engineers (IEICE)
• The Acoustical Society of Japan (ASJ)