Automatic Object Detection from Digital Images by Deep Learning with Transfer Learning

Nobuyoshi Yabuki, Naoto Nishimura, and Tomohiro Fukuda

Osaka University, Suita, Osaka 565-0871, Japan
yabuki@see.eng.osaka-u.ac.jp

Abstract. At construction sites and disaster areas, an enormous number of digital photographs are taken by engineers. Tasks such as collecting, sorting, annotating, storing, deleting, distributing these digital images, as done manually, are cumbersome, error-prone, and time-consuming. Thus, it is desirable to automate the object detection process of pictures so that engineers do not have to waste their valuable time and can improve the efficiency and accuracy. Although conventional machine learning could be a solution, it takes much time for researchers to determine features and contents of digital images, and the accuracy tends to be unsatisfactory. On the other hand, deep learning can automatically determine features and contents of various objects from digital images. Therefore, this research aims to automatically detect each object as an object and its position from digital images by using deep learning. Since deep learning usually requires a very large amount of dataset, this research has adopted deep learning with transfer learning, which enables object detection even if the dataset is not very large. Experiments were executed to detect construction machines, workers, and signboards in photographs, comparing among the conventional machine learning by feature values, deep learning with and without transfer learning. The result showed that the best performance was achieved by the deep learning with transfer learning.

Keywords: Deep learning · Image detection · Transfer learning

1 Introduction

An enormous number of pictures are taken at construction sites and disaster areas by engineers and are used for inspection, management, disaster recovery, and scientific interests. Those pictures include objects such as construction machinery, signage, signboards, construction workers, etc. In order to use those pictures for inspection and management of construction, those objects must be manually detected by humans. In order to improve the efficiency, the detection process should be automatically executed. Recently, photographs can be classified by object detection functions using machine learning, e.g., People of iOS [1] and face grouping in Google Photo [2]. However, those systems are mostly for ordinary things such as faces, automobiles, bicycles, televisions,

I. F. C. Smith and B. Domer (Eds.): EG-ICE 2018, LNCS 10863, pp. 3–15, 2018.
https://doi.org/10.1007/978-3-319-91635-4_1

etc., but not for specific civil engineering entities such as construction machinery, signboards at construction sites, stakes, scaffoldings, etc.

In object detection in images, machine learning using features of the image has often been employed. Features used for object detection include local features, such as Haar-like features [3], Histograms of Oriented Gradients (HOG) features [4], Edge Oriented Gradients features [5], Edgelet features [6], etc. However, appropriate features must be determined by the user and it is difficult to achieve the satisfactory level of object detection. In addition, since using single feature makes much false detection, joint features or boosting [7] which combines plural object detection mechanisms are necessary, which requires much work and effort during the machine learning phase.

On the other hand, in deep learning or Convolutional Neural Networks (CNNs), features of objects can automatically be found and computed. Thus, the user does not have to determine the features and the accuracy of object detection has been improved significantly. Research on ordinary object detection include VGG-16 [8], Regions with CNN features (R-CNN) [9], Fast Region-based Convolutional Network (Fast R-CNN) [10], Faster R-CNN [11], Single Shot Multibox Detector (SSD) [12], You Only Look Once (YOLO) [13], etc. However, huge amount of data is required for deep learning. For ordinary object detection, large datasets for learning such as Pascal VOC [14], Caltech101 [15], COCO dataset [16, 17], UCI Machine Learning Repository [18] are provided for learning. For example, Pascal VOC has 20 classes such as person, animal, vehicle, indoor with 9,963 images containing 24,640 annotated objects. However, datasets for non-ordinary objects such as construction machinery, civil engineering objects are not prepared yet and must be developed to achieve the high accuracy in object detection.

Machine learning methods include supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning. Recently, transfer learning receives attention from Artificial Intelligence (AI) researchers. In transfer learning [19, 20], knowledge is stored for solving one problem and it is applied to a different but related problem. Transfer learning is thought to be particularly useful when large dataset cannot be collected easily. Fine tuning is a subset of transfer learning, which classifies objects that are already classified with similar but different labels. In this research, transfer learning but not fine tuning is employed.

The objective of this research is to develop a method to automatically detect construction domain specific objects such as construction machines, construction workers, construction signboards from digital images using deep learning with transfer learning and to classify the pictures according to the image contents. The reason for adopting transfer learning is that existing datasets lack of civil engineering specific objects and that it is hard to collect huge amount of such data to be needed for deep learning in a short time with a limited fund. To verify the proposed method, performance is compared among conventional machine learning using HOG features, deep learning with and without transfer learning. The reason that the single HOG features and Support Vector Machine (SVM) was employed for object detection is that it has an advantage for local shape change over other features described before.

2 Proposed Object Detection Method

2.1 Overview of the Proposed Method

The proposed method is to detect objects of the construction-specific domain and their positions from digital images. SSD is employed to detect positions as well as the types of the objects in images. Next, correct answer labels to re-learn SSD is created. LabelImg [21] is used to create learning data labels. Finally, the created dataset is re-learned to the SSD. By using the new weight acquired by transfer learning, the object position on the digital image are determined.

In this research, Keras was used as a machine learning library. It is one of the machine learning libraries written in Python. TensorFlow can be used as the back end with Keras. TensorFlow, which is developed and provided by Google Inc., was used for recognizing the object position as the backend. Figure 1 shows the overview of the flow of object position detection method.

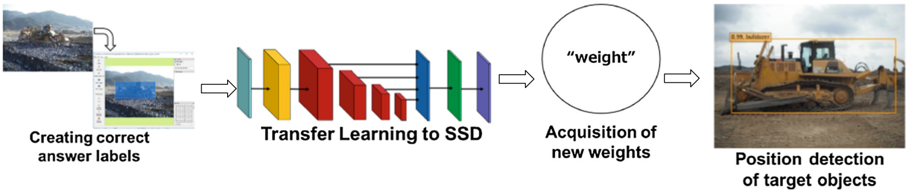

Creating correct answer labels
Transfer Learning to SSD
"weight"
Acquisition of new weights
0.99, bulldozer
Position detection of target objects

Fig. 1. Overview of the flow of object position detection method.

SSD is an object detection algorithm using a simple network built by Liu et al. SSD is faster than conventional object detection algorithms. A simplified network model of SSD is shown in Fig. 2. VGG-16, which is a learning model of image detection, is used for the base network of SSD. After this base network, by using hierarchical feature maps, various scale objects can be processed, and the accuracy of detection rate can be high. The reason is that it identifies for each aspect ratio of the object. For these reasons, the

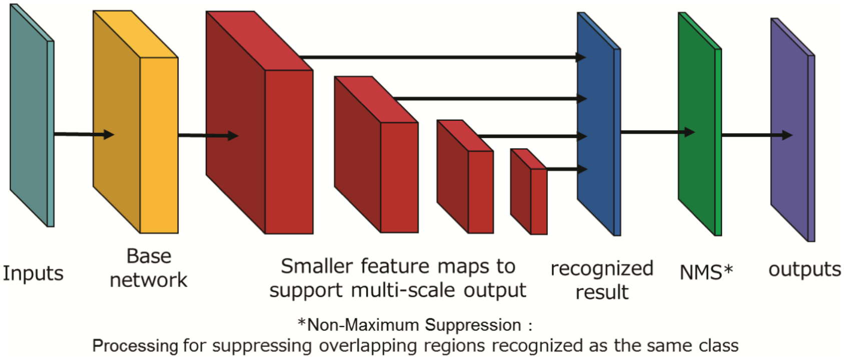
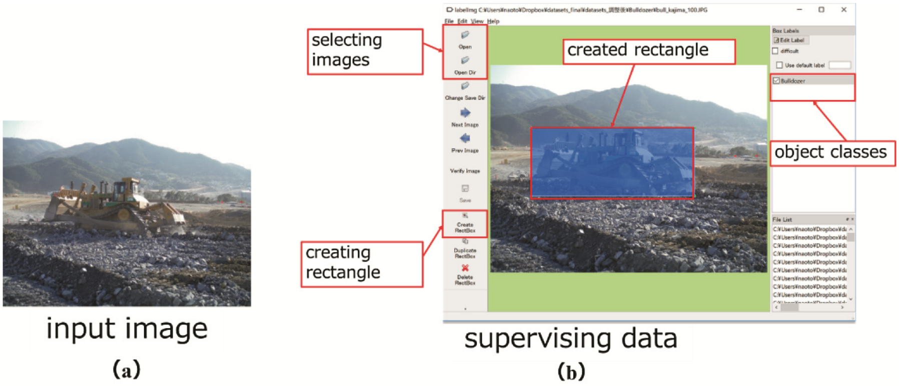

Inputs
Base network
Smaller feature maps to support multi-scale output
recognized result
NMS*
outputs
*Non-Maximum Suppression :
Processing for suppressing overlapping regions recognized as the same class

Fig. 2. Simplified network image model of SSD (drawn by the authors referencing to Fig. 2 of Liu et al.).

SSD can detect a target object even from a relatively low-resolution image. Even if digital images taken at a construction site are high resolution, a low-resolution object exists in the distance of images.

2.2 Detection of Object Type and Position

Creation of Correct Answer Labels for Object Position Detection. In order to detect an object from a digital image with machine learning, it is necessary to know what exists and where it is. Hence, a bounding box and its coordinates are required to create correct answer labels. In addition, correct answer labels must exist with the original digital images. They are created by using LabelImg [21]. This tool was built by Tzutalin to create a correct label called annotation data. The annotation data includes information such as the name of the original image and coordinates of a bounding box (Horizontal and vertical coordinates on the image). First, image data to be learned is opened (Fig. 3(a)). Then, the coordinates of upper left and lower right corners of the object are measured. It is possible to create bounding boxes of a plurality of objects for one image



input image

(a)

supervising data

(b)

```
          <filename>image file name</filename>
          <path>path for image file</path>
          </source>
          <size>
                    <width>2048</width> (width of original image)
                    <height>1536</height> (height original image )
                    <depth>3</depth> (depth of original image)
          </size>
          <object>
                    <name>Bulldozer (object classes) </name>
                    <bndbox> (position of rectangle)
                              <xmin>303</xmin> ( minimum horizontal coordinate of rectangle )
                              <ymin>466</ymin> ( minimum vertical coordinate of rectangle )
                              <xmax>1511</xmax> ( maximum horizontal coordinate of rectangle )
                              <ymax>969</ymax> ( maximum vertical coordinate of rectangle )
                    </bndbox>
          </object>
```

contents of supervising data

(c)

Fig. 3. Sample of supervising data.

(Fig. 3(b)). The created annotation data is saved as an XML format file as shown in Fig. 3(c).

Transfer Learning and Object Type/Position Detection. The created correct labels and the original images are learned and stored in SSD. When learning is completed, a new set of weights is acquired. An appropriate set of weights is selected from the newly acquired weights and used for object detection. The selection of the appropriate set of weights is done from the transition of the loss coefficient acquired by transfer learning (Fig. 4). In Fig. 4, since the difference of loss between training data and test data is minimum at Epoch 41, the weight at this point is selected.

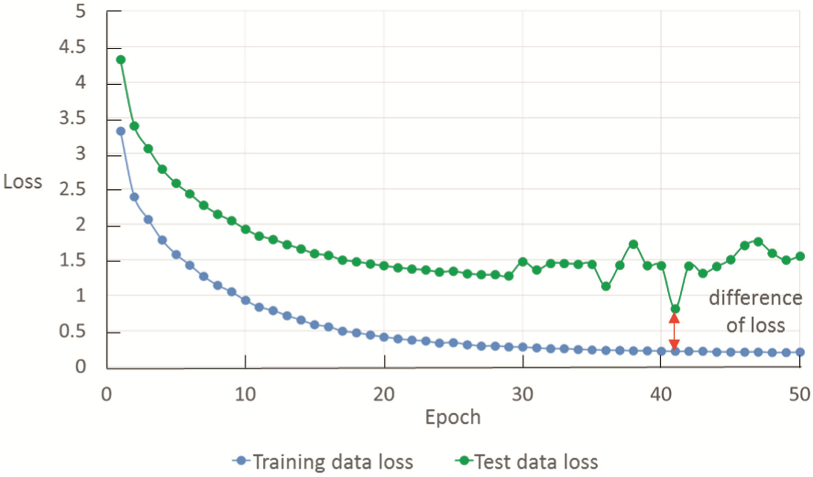


Fig. 4. A sample of transition of training data loss and test data loss.

Classification of Digital Image Files. File folders of which names are detected objects such as bulldozers, backhoes, dump trucks, wheel loaders, workers, signboards, etc. are prepared in a computer. All digital images files that at least one object is detected are put into the designated file folder. If two or more objects are detected, the file is copied and put into the multiple designated folders.

3 Experiments

Two kinds of experiments were executed. Experiment I is to compare the detection result between the proposed method, i.e., deep learning with transfer learning and deep learning without transfer learning. Experiment II is to compare the results among (a) deep learning with transfer learning, (b) deep learning without transfer learning, and (c) conventional machine learning. Construction machinery (backhoes, bulldozers, dump tracks, and wheel loaders), workers, and signboards were selected as objects to be detected, and the accuracy of object position detection was tested. Detection accuracy is evaluated based on the criteria shown in Table 1. System environment for learning and testing is shown in Table 2. The number of images for Experiment I, II, and for testing is shown in Table 3.

Table 1. Evaluation criteria for detection accuracy.

Types of detection	Position detection
Positive detection	Both position and object type are correctly detected
Negative detection	Either (1) position is correctly detected but the object type is incorrect or (2) the object type is correctly detected but the position is incorrect
Not detected	Nothing is detected

Table 2. System environment for learning and testing.

Item	Type
CPU	Intel Core i7-7700 BOX CPU @3.60 GHz
Main memory	64 GB
GPU	GeForce GTX 1080 Ti. Memory: 11 Gbps, 11 GB
OS	Ubuntu 14.04, 64 bit (Linux)
Language	Python 2.0

Table 3. The number of images for Experiment I, II, and testing.

Object	Experiment I	Experiment II	Testing
Backhoe	115	84	202
Bulldozer	113	111	107
Dump truck	100	97	115
Wheel loader	117	117	123
Worker	302	249	64
Signboard	300	217	78

3.1 Experiment I: Verification for Deep Learning with Transfer Learning

Experiment I was executed for comparing the detection accuracy result between deep learning with transfer learning and without it. The model used for deep learning with transfer learning is named I-A while that for without it is named I-B in this research. 90% of the learning dataset was used for training and 10% for testing.

Figure 5 shows samples for positive detection and negative detection cases for each object. For example, in the negative detection of backhoe, the machine was detected as a dump truck. Figure 6 shows the result of the object detection experiment using the testing data. Detection accuracy for I-A (deep learning with transfer learning) was 86.6% for backhoes, 61.7% for bulldozers, 80.9% for dump trucks, 86.2% for wheel loaders, 100.0% for signboards, and 79.7% for workers. On the other hand, accuracy for I-B (deep learning without transfer learning) was, 0.0%, 20.6%, 0.0%, 17.1%, 60.9%, and 0.0%, respectively. Obviously, I-A shows much accuracy rate than I-B.

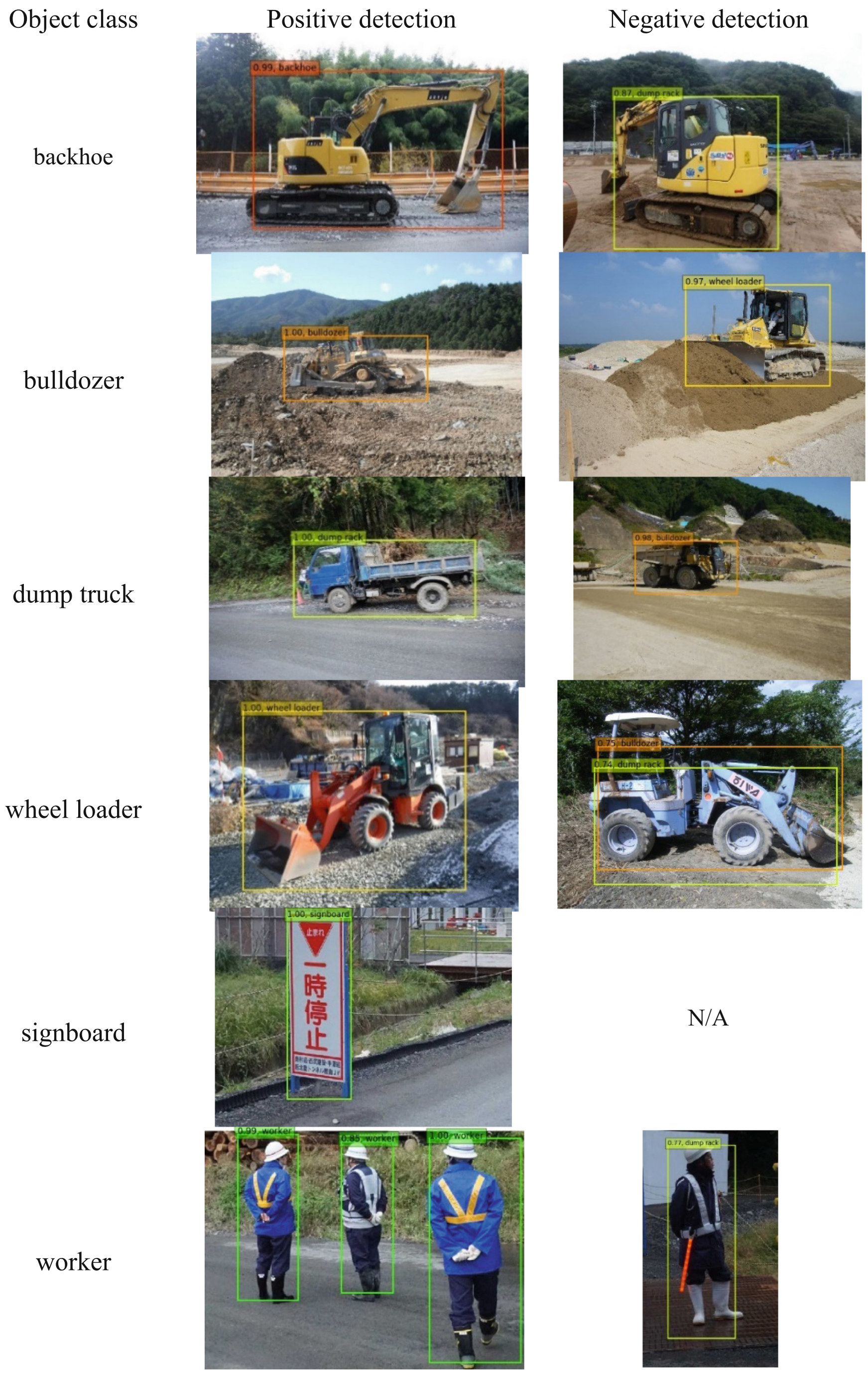
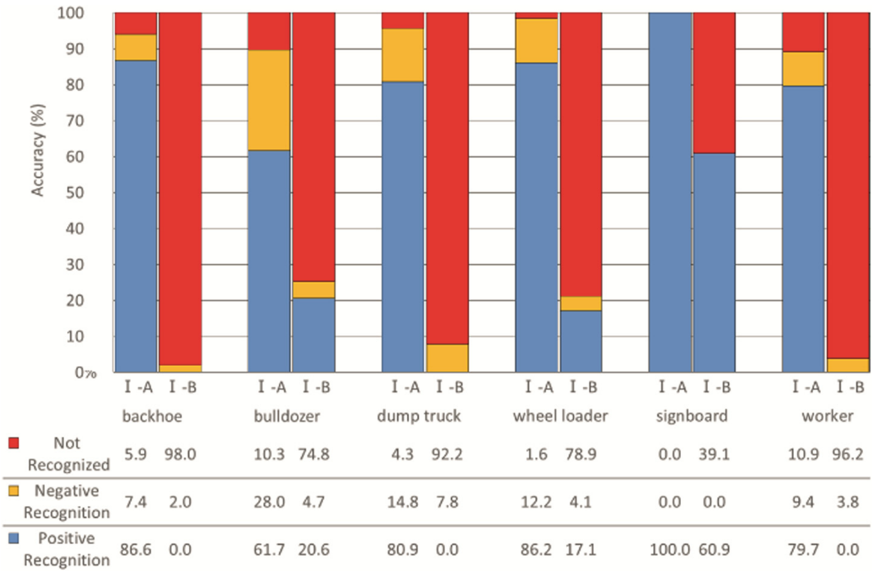


Fig. 5. Samples for positive and negative detection cases for each object class.



	I -A	I -B	I -A	I -B	I -A	I -B	I -A	I -B	I -A	I -B	I -A	I -B
	backhoe		bulldozer		dump truck		wheel loader		signboard		worker	
🟥 Not Recognized	5.9	98.0	10.3	74.8	4.3	92.2	1.6	78.9	0.0	39.1	10.9	96.2
🟨 Negative Recognition	7.4	2.0	28.0	4.7	14.8	7.8	12.2	4.1	0.0	0.0	9.4	3.8
🟦 Positive Recognition	86.6	0.0	61.7	20.6	80.9	0.0	86.2	17.1	100.0	60.9	79.7	0.0

Fig. 6. Result of object detection experiment (Experiment I) using the testing data.

3.2 Experiment II: Comparison with Conventional Machine Learning

Experiment II was executed for comparing the detection accuracy result between deep learning with transfer learning and conventional machine learning (CML). The model used for deep learning with transfer learning is named II-A and that for CML is named II-CML in Experiment II.

Implementation of CML and Numbers of Images. In this research, object detection by HOG features and SVM was executed using Dlib [22], which is a machine learning library. The reason for using Dlib is that objects can be detected with the same creation method of the correct labels as SSD. In object detection using Dlib, if the aspect ratio

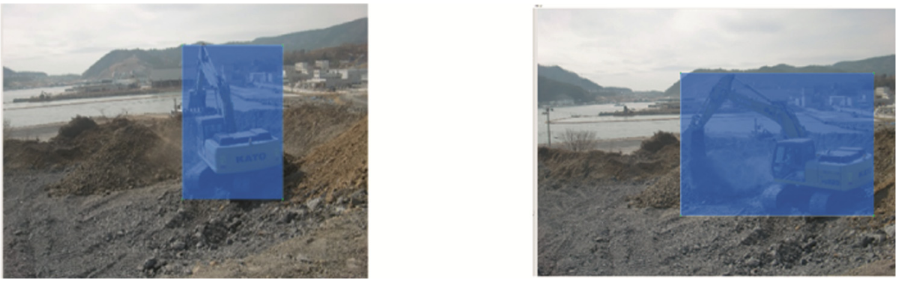


Fig. 7. Sample case of very different aspect ratio of the bounding boxes created for representing correct labels.

of the bounding box created as a correct label is greatly different as shown in Fig. 7, it cannot be used for learning.

Comparison of Detection Results between Deep Learning and CML. Samples of the result of image detection are shown in Fig. 8. In Fig. 8, positive detection by II-A and negative detection by II-CML are shown. In II-CML, Dlib with HOG and SVM was used for detection whereas, in II-A, SSD was used for detection. Figure 9 shows the comparison of the accuracy between II-A and II-CML. The accuracy of the method using deep learning with transfer learning (II-A) is about 90% or higher for all objects while that of II-CML ranges between 30% and 75%. Thus, Experiment shows that the proposed method using deep learning with transfer learning has much higher accuracy compared to CML using HOG and SVM.

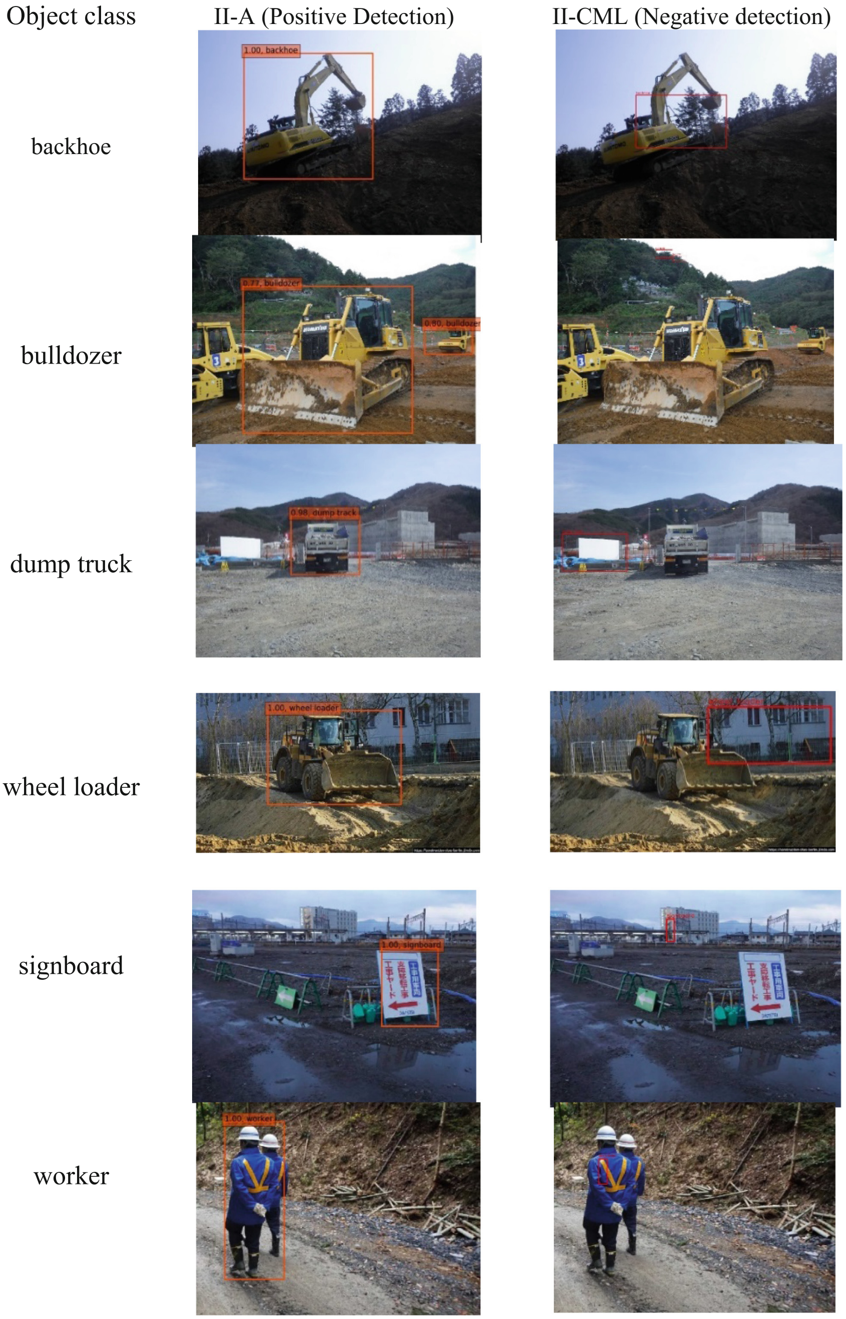
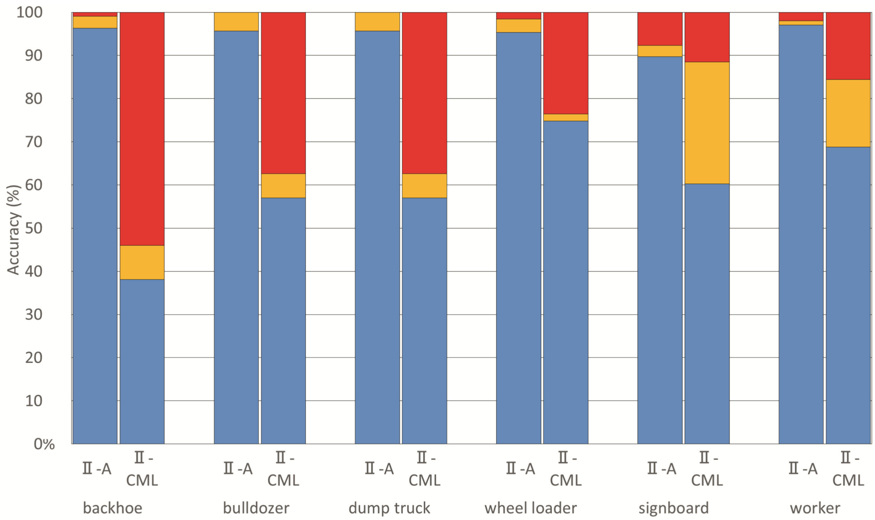


Fig. 8. Sample cases of positive detection by deep learning with transfer learning (II-A) and negative detection by conventional machine learning (II-CML).



	backhoe		bulldozer		dump truck		wheel loader		signboard		worker	
	Ⅱ-A	Ⅱ-CML	Ⅱ-A	Ⅱ-CML	Ⅱ-A	Ⅱ-CML	Ⅱ-A	Ⅱ-CML	Ⅱ-A	Ⅱ-CML	Ⅱ-A	Ⅱ-CML
🟥 Not Recognized	0.9	54.0	0.0	37.4	0.0	37.4	1.6	23.6	7.7	11.5	2.0	15.6
🟨 Negative Recognition	2.8	7.9	4.3	5.6	4.3	5.6	3.1	1.6	2.6	28.2	1.0	15.6
🟦 Positive Recognition	96.3	38.1	95.7	57.0	95.7	57.0	95.3	74.8	89.7	60.3	97.0	68.8

Fig. 9. Testing result of object position detection comparing among deep learning with transfer learning (II-A) and conventional machine learning (II-CML).

4 Conclusions

In this research, a problem was identified in tasks such as collecting, sorting, annotating, storing, deleting, distributing enormous number of digital photographs taken by engineers at construction sites and disaster areas. It was the challenge to manually classify a huge number of photographs. In order to automate these error-prone and cumbersome tasks, an object detection method was proposed, which can detect not just the object class but the position as well as a bounding box. In the proposed method, deep learning (CNNs) with transfer learning was employed because construction-specific objects such as construction machinery, workers, signboards, are not available in the large amount of datasets provided for AI researchers. Most of the datasets usually include ordinary things. The reason for adopting deep learning with transfer learning is that deep learning can automatically determine features from digital images while conventional machine learning requires manual feature selection. The second reason is while deep learning has this advantage, it requires a huge amount of training data, which is difficult to obtain for construction-specific objects that are not available in popular, open datasets. Thus, transfer learning, where a knowledge obtained for one problem can be applied to different but related problems. After the object position detection is done, digital photo files can be classified, copied and put into the designated folders automatically.

Based on the proposed methodology, a system was developed employing SSD and VGG-16. A number of digital images of backhoes, bulldozers, dump trucks, wheel loaders, construction workers, and construction signboards were collected and used for learning. To verify the proposed method, Experiment I, which compares the object detection accuracy between deep learning with and without transfer learning, was executed. As a result, proposed method showed about 80% accuracy while the latter showed very poor performance. Next, Experiment II, which compares the deep learning with transfer learning and conventional machine learning (CML) using HOG features with SVM. The result showed that the proposed method showed about 90% accuracy while that of CML ranged from 30 to 75%.

In conclusion, the proposed method of deep learning with transfer learning can be a useful and effective way to detect object positions from digital images, especially in the area where is domain is specific such as construction sites and huge datasets are not available.

Future work includes developing an object shape detection methodology and apply it to construction-specific objects and improving the object detection accuracy by enhancing the quality of data and increasing the amount of data. As the potential applications of this research in construction and disaster management are more widespread, the authors are planning to apply the future research outcomes to various applications.

References

1. About people in photos on your iPhone, iPad, or iPod touch. https://support.apple.com/en-us/HT207103. Accessed 19 Jan 2018
2. Google products. https://www.google.com/about/products/. Accessed 19 Jan 2018
3. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Kauai, pp. 511–518. IEEE (2001)
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, vol. 1, pp. 511–518. IEEE (2005)
5. Levi, K., Weiss, Y.: Learning object detection from a small number of examples: the importance of good features. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington D.C., vol. 2, pp. 53–60. IEEE (2004)
6. Wu, B., Nevatia, R.: Detection of multiple, partially occluded humans in a single image by Bayesian combination of edgelet part detectors. In: Tenth IEEE International Conference on Computer Vision, Beijing, vol. 1, pp. 90–97. IEEE (2005)
7. Mitsui, T., Fujiyoshi, H.: Object detection by joint features based on two-stage boosting. In: Proceedings of 2009 IEEE 12th International Conference on Computer Vision Workshops, Kyoto, pp. 1169–1176. IEEE (2009)
8. Simonyan, K., Andrew, Z.: Very deep convolutional networks for large-scale image recognition. In: Proceedings of International Conference on Learning Representations 2015, San Diego (2015). Preprint CoRR: arXiv:1409.1556
9. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Ohio, pp. 580–587. IEEE (2014)

10. Girshick, R.: Fast R-CNN. In: Proceedings of 15th IEEE International Conference on Computer Vision, Santiago, pp. 1440–1448. IEEE (2015)
11. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Proceedings of Advances in Neural Information Processing Systems, Montreal, pp. 91–99. NIPS (2015)
12. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C.: SSD: single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
13. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, pp. 779–788. IEEE (2016)
14. The PASCAL Visual Object Classes Homepage. http://host.robots.ox.ac.uk/pascal/VOC/. Accessed 19 Jan 2018
15. Caltech 101. http://www.vision.caltech.edu/Image_Datasets/Caltech101/. Accessed 19 Jan 2018
16. COCO dataset. http://cocodataset.org/. Accessed 19 Jan 2018
17. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common Objects in Context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
18. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml/about.html. Accessed 19 Jan 2018
19. Transfer Learning – Machine Learning's Next Frontier. http://ruder.io/transfer-learning/index.html. Accessed 19 Jan 2018
20. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**(10), 1345–1359 (2010)
21. LabelImg. https://github.com/tzutalin/labelImg. Accessed 1 Sept 2017
22. Dlib C++ Library Python API. http://dlib.net/python/index.html. Accessed 1 Sept 2017