Classification of Images from Construction Sites Using a Deep-Learning Algorithm

Daeyoung Gil, Ghang Lee[*], and Kahyun Jeon

Department of Architecture and Architectural Engineering, Yonsei University, South Korea
E-mail: rlfeodud@gmail.com, glee@yonsei.ac.kr, jeonkh0310@naver.com

Abstract –

Field engineers take and collect several pictures from construction sites every day, and these pictures serve as records of a project. However, many of these images are loaded to and remain on computers in an unorganized manner because tagging, renaming, and organizing them is a time-consuming process. This paper proposes a method for automatically classifying construction photographs by job-type using a deep-learning algorithm. The first goal of this study is to classify construction images according to 27 job-types based on OmniClass Level 2. Google Inception v3—a deep learning algorithm used in this study as an image classifier—was trained using 1,208 construction pictures labeled by job-type. To improve the performance of the classifier, the optimized number of trainings was determined by examining the changes of accuracy and cross-entropy during trainings. The first result shows the incidence of several trainings over 50,000 was not meaningful. The retrained Google Inception as a construction image classifier was validated using a total of 235 images. The validation result shows that the classifier demonstrates an accuracy of 92.6% in classifying inputs properly and an average precision of 58.2% in correct classification. This means that retrained classifier can classify approximately nine out of every ten images correctly and that the deep-learning algorithm has high potential for use in the automatic classification of images from construction sites.

Keywords –

Image classification; Deep learning; Construction site monitoring; Convolutional neural network;

1 Introduction

Pictures taken by field engineers on a construction site contain various information about the site. From a daily report of construction progress to the construction method implemented for each project, pictures have an important role in project documentation, and enormous amounts of pictures are taken to establish a visual record. However, because thousands of pictures are taken for construction projects, the pictures are usually stored in a computer in an unorganized manner after a project has been completed. To properly utilize the information available from these pictures, the pictures must first be classified before they are stored.

This paper proposes the use of a deep-learning algorithm to automatically classify pictures from construction sites according to the job-type that each picture contains using a deep-learning algorithm. After AlexNet [1] won the first prize at ILSVRC 2012 (ImageNet Large Scale Visual Recognition Challenge) using a convolutional neural network (CNN), CNN became the most popular deep-learning method in image classification. Since then, the field of image classification has been developing rapidly. Through the constant development of image classification algorithms, recent studies have reached 96% accuracy in image classification [7].

This study uses Google Inception v3—the latest CNN developed by Google [2]—as an image classifier to automatically classify construction pictures by job-type. To test the performance of the trained classifier, 1,208 pictures were used for training, and 27 images were used for validation.

This paper is divided into five sections. The second section presents a review of previous studies related to this research. The third section describes the overall design of the research and, section 4 reports the experiments and the analytical results. The final section reports the paper's conclusions.

2 Background

2.1 Image Classification on Construction

Several studies have been conducted regarding how

[*] Corresponding author

to extract information from images of construction sites. A recent study related to image processing focused on object detection from pictures of a construction site. Chi and Caldas [9] suggested a way to find objects from images using an artificial neural network (ANN), and Zhu et al. [4] proposed a method of finding concrete area in a picture using parameter optimization. In addition, Wu et al. [15] used image filtering in order to find specific objects from construction images.

Currently, however, meaningful data is generated from images using the previously developed technique. Kim et al. [3] derived information regarding construction progress from construction images by combining project schedule data and filtered images. Also, Kim et al. [16] suggested a method of measuring the construction progress that uses the 4D information of a given building. More recently, adopting deep-learning, Fang et al. [10] proposed a way to detect non-hardhat users in a construction site. Like the above two studies, not only detecting object from images, but also extracting data that can be useful is the main problem of recent studies. Thus, in this study, we extract the job-type information from construction pictures and use it to classify those pictures as a standard using a deep-learning algorithm.

2.2 Deep learning

In 2012, an image classifier constructed with a deep-learning algorithm was introduced at ILSVRC 2012. The name of the algorithm was AlexNet [1], and it was composed of a CNN with eight layers. With a 16.4% error rate, it surpassed its competitors, whose error rates were over 26% on average, and won the first prize. Since then, CNN has become prominent in the field of image classification, and algorithms composed of CNNs show high accuracy nowadays [5][6][7].

The major difference between deep learning and conventional machine learning based on image processing is not only the high accuracy of image classification but also the ability to conduct feature extraction. The former requires a pre-process of image filtering or feature extraction, whereas a deep learning algorithm can conduct feature extraction automatically by using a huge amount of training data [8]. Although deep learning is being criticized for being a black box approach, its implementation process requires a clear goal first, a deep learning network design, and data collection and analysis based on the goal. Moreover, a considerable amount of programming and optimization efforts is required.

2.3 Google Inception v3

In general, when the structure of a neural network deepens and widens, the performance of the network improves. However, the likelihood that overfitting and vanishing will occur also increases. To prevent overfitting and vanishing, GoogleNet [5] is designed with multiple (22) concatenating layers. Based on its structure, GoogleNet won the first prize at ILSVRC 2014 with a 6.7% error rate. To classify images, we used Google Inception v3 [2], which is developed from GoogleNet. Compared with the previous version, Inception v3 has an improved network structure.

When supervised learning is implemented, both images and label data are needed. The label data indicates what information each image contains. Therefore, in order to use this network in our research, we must retrain the network with labelled pictures that have been prepared for retraining in advance. Using the pictures from the training dataset with label data, the classifier learns each picture and label data in a pair. For example, if an image is given to the classifier after retraining a similar picture with a "Foundation" label, it may determine that the input image is about "Foundation".

3 Research Method

Figure 1 describes the overall research flow. First, pictures from construction sites are gathered for training and validation of the image classifier: Google Inception v3. One of strengths of Google Inception v3 is that it has already pre-trained with a very large set of various types of images from ImageNet [12]. Google Inception v3, however, should be "retrained" with an additional set of images if it is used for a new purpose [13]. Thus, the second step is to retrain Google Inception v3 with construction images labeled by job type so that it can be used as an image classifier for construction pictures [13]. During the retraining process, the classifier learns which image features are associated with which job type.

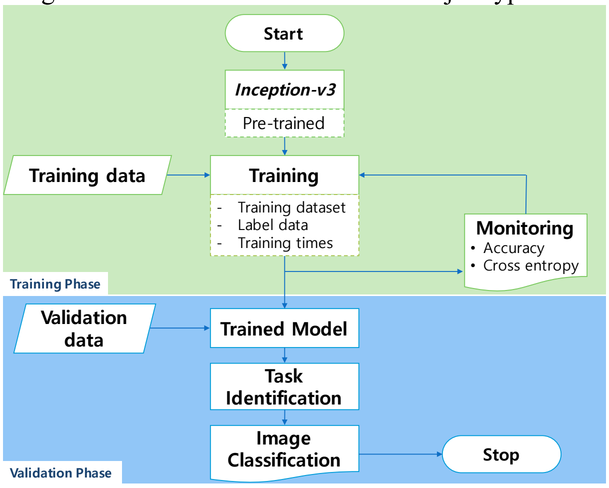

Figure 1. Overall flow of research

At first, we optimized the number of training times with the given training dataset before validating the classifier. Then, after the retraining procedure with the optimized training times was conducted, new images from the validation dataset were inputted into the classifier. Comparing the results, we checked the accuracy and precision of the classifier.

In this study, OmniClass was used as a standard for labeling pictures by job-type. OmniClass is system for classifying the type of construction industry [14]. According to OmniClass, there are 16 tables that can be used to classify the type of industry, and we chose Table 21 – Elements. Originally, there are 29 categories in that table, but we eliminated two categories because they have ambiguous definitions: Integrated Automation and Facility Remediation.

3.1 Training dataset preparation

Because the input image should be classified according to the job-type of construction, we set the 27 standards referring to OmniClass Level 2 categories. After the standard of construction elements was classified according to OmniClass, prepared pictures of the training dataset were labeled by each adequate category. A total of 1,208 pictures in 27 categories were used as construction job-types and the training dataset, and 44 pictures on average were distributed for each category.

4 Experiment & Result

To validate the module, we conducted two different experiments. First, we retrained the classifier 400,000 times with 1,208 pictures in order to find the optimized training time. It is ideal to train a very large number of pictures many times, but, in reality, it is challenging to acquire a large number of pictures and a long training time. Thus, observing the training graph, we tried to determine the optimized point of the training times using a given number of pictures. Second, after retraining the classifier again with the optimized point, we put new images that were not used for retraining into the classifier. Through the two experiments, we yielded a retrained classifier that was optimized and how well it classifies images accurately and precisely.

4.1 Optimization in training times

We used the stochastic gradient descent (SGD) algorithm for the optimization process. SGD requires training of the classifier with the same set of data multiple times to progressively fit the classifier to the training dataset. After the trained classifier is fitted well to the training dataset as the training time increases, it can classify new input data based on the optimized parameters inside the classifier [18]. Thus, training should be repeated for fitting with the same dataset. In general, the performance of a classifier improves as the training time increases. However, too much training can cause an overfitting problem, which worsens performance. Training also takes much time. To maximize the efficiency of the training process, we retrained the classifier 400,000 times and looked for the optimized point. During the retraining, the accuracy value when random data are inputted and the cross-entropy data are extracted, and they were used to assess performance.

4.1.1 Accuracy

Giving the classifier random data during the retraining process, we extracted accuracy data (Figure 2). The average accuracy refers to how well the classifier distributes the random input data into correct categories. In this process, the input data is not an image used for retraining but the randomly created image that has label information.

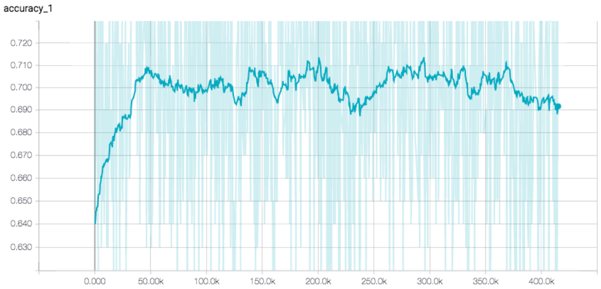

Figure 2. Accuracy (y-axis) changes according to the number of trainings (x-axis). Average accuracy (thick line) first rises dramatically but then hardly changes after the training time exceeds 50,000.

The accuracy value initially increased, but after about 50,000 times of training, the increasing stopped and the value started to fluctuate. With a value of 0.770, the average accuracy is the highest with a training time of 36,700. Figure 2 shows that conducting more than 50,000 trainings is not meaningful in terms of efficiency.

4.1.2 Cross-entropy

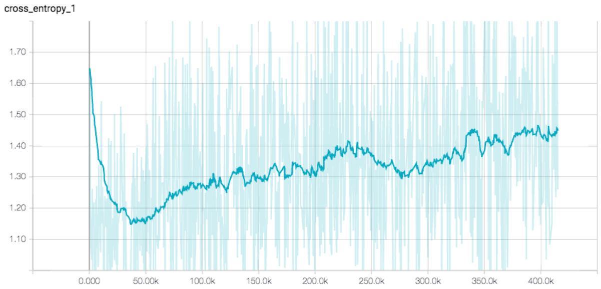

Figure 3. Cross-entropy value (y-axis) initially decreases (x-axis) but gradually increases as the training time exceeds 50,000.

In information theory, cross-entropy is an index that shows the difference between two probability distributions. It is well used for optimization problems and machine learning as a loss function. For one layer, when we suppose p_i represents the probability of the true label and q_i represents the distribution of the predicted value, we can define the cross-entropy loss $H(p,q)$ as follows [17]:

$$H(p,q) = -\sum_{i} p_i \log q_i$$

However, as we have N layers in the classifier algorithm, we need the averaged loss value J so that we can infer the loss of the whole layers. Then, J can be calculated as follows:

$$J = \frac{1}{N}\sum_{n=1}^{N} H(p_n, q_n)$$

In this study, p_i indicates the value distribution of the classifier in the retraining process, and q_i indicates that of the training dataset. Then, the value J represents how different the distribution of the classified output is from that of the original training dataset when random noise data are inputted during the retraining process. Therefore, the low value of cross-entropy indicates a well-trained model, and it is utilized as a loss function in the optimization problem in a direction that minimizes its value [11]. Therefore, a low cross-entropy value indicates that the training was done well.

The result graph (Figure 3) shows that the minimum cross-entropy value, 0.9561, occurs with 36,700 training times and starts to increase gradually after about 50,000 training times.

4.1.3 Determining optimized point

When it comes to accuracy results, the performance of the classifier does not change after the training times exceeds 50,000 although there is some fluctuation. However, when the result of cross-entropy is considered with accuracy together, it is recommended to select an optimized training time that is between 35,000 and 50,000. In this study, we retrained the classifier 37,000 times and conducted further research.

4.2 Validation

After retraining the classifier with optimized training times, validation of whether the retrained classifier works well. In this study, the validation process was implemented by inputting new images that were not used for training. Eight images on average for each category were prepared, so a total of 235 images were inputted to the classifier and some example images are shown in Figure 4.

The result is shown in Figure 5. Categories on the x-axis are the labels of input images, and categories on the y-axis are those for labels created automatically. The numbers in the cell indicate the averaged probabilities of the input images having the label of the y-axis when the images labelled with the categories on the x-axis are inputted. The probability numbers are used to indicate precision, and their sum on each column should ideally be 1.00. However, some of them do not reach 1.00 because probabilities below 0.05 are eliminated during the process of being averaged. The yellow cell also indicates the highest number in each column, and the classifier defines the image as the label on the y-axis of the cell at that time.

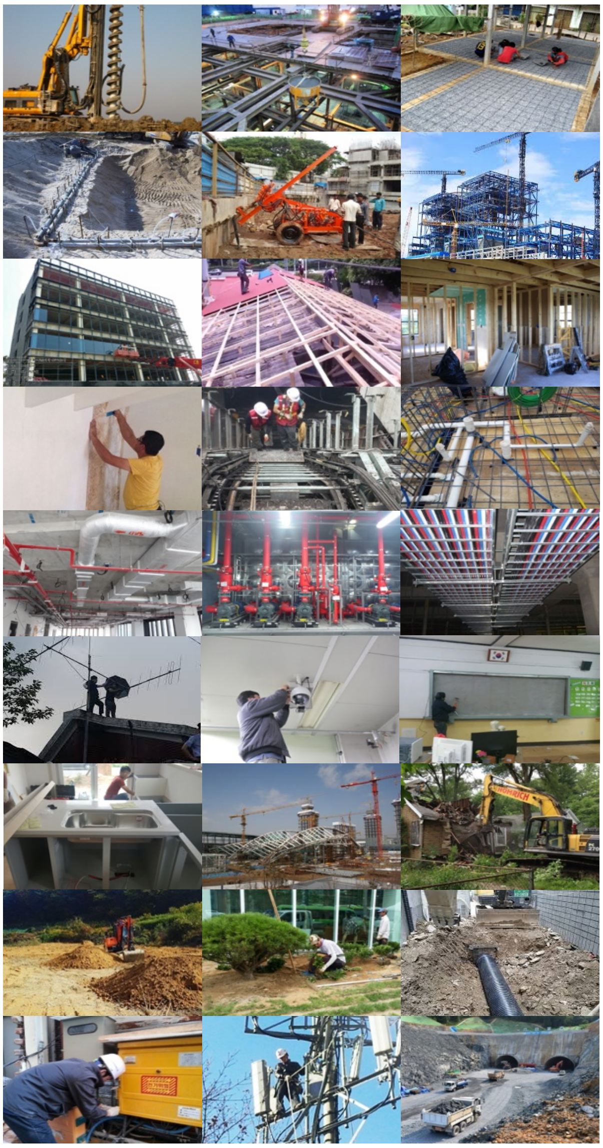

Figure 4. Example images used for validation

For example, if the input images are classified arbitrarily, the numbers of every cell should converge to 3.7%, and the output labels should be determined randomly. However, when the unlabeled images that

Foundation	0.31		0.06		0.03						0.01													0.20			
Subgrade_enc		0.26	0.16		0.05	0.12			0.09		0.04				0.03			0.03									0.03
Slab on grade		0.07	0.11	0.02				0.03				0.03						0.02							0.06		
Water and gas mitigation		0.02	0.14	0.66	0.03																		0.02	0.02			0.15
Substructure related activity	0.05	0.15	0.03	0.02	0.55			0.05													0.01	0.03					0.03
Superstructure		0.19	0.12		0.01	0.68	0.04				0.17				0.06			0.11		0.02							
Exterior vertical enclosure	0.09	0.01			0.01		0.45	0.12	0.08	0.02	0.02	0.09	0.07														
Exterior horizontal enclosure		0.03	0.04				0.07	0.50	0.08							0.28				0.01			0.07				
Interior construction						0.01	0.09	0.02	0.48	0.12	0.04		0.04		0.06		0.06										
Interior finishes									0.02	0.73																	
Conveying	0.27	0.06						0.01	0.02		0.17				0.08					0.03							
Plumbing			0.07	0.14	0.03			0.06	0.02		0.03	0.38	0.02	0.08	0.10								0.01				
HVAC						0.09				0.02	0.09	0.03	0.56	0.09	0.21			0.02			0.01						0.05
Fire protection												0.17		0.68				0.08									
Electrical						0.01	0.13	0.14	0.01	0.02	0.10	0.07	0.07		0.25												
Communication																0.34											
Electronic safety and security												0.03					0.73	0.04							0.01		
Equipment									0.06		0.05		0.03				0.03	0.48	0.02								
Furnishings																	0.02	0.02	0.94	0.12							
Special construction							0.03				0.01	0.01	0.09							0.66							
Demolition	0.16				0.01						0.05							0.02		0.01	0.77	0.07			0.09		
Site preparation				0.04	0.14																0.07	0.81	0.01		0.01		0.07
Site improvement												0.03		0.03							0.01		0.73		0.01		0.01
Liquid and gas site utilities			0.01	0.03																			0.07	0.75			0.03
Electrical site improvement	0.08										0.03	0.02				0.35				0.03	0.04				0.67	0.09	
Site communication							0.03							0.03												0.86	
Miscellaneous site construction											0.04				0.02			0.08									0.50
Resulted Categories / **Label of input images**	**Foundation**	**Subgrade enclosure**	**Slab on grade**	**Water and gas mitigation**	**Substructure related activity**	**Superstructure**	**Exterior vertical enclosure**	**Exterior horizontal enclosure**	**Interior construction**	**Interior finishes**	**Conveying**	**Plumbing**	**HVAC**	**Fire protection**	**Electrical**	**Communication**	**Electronic safety and security**	**Equipment**	**Furnishings**	**Special construction**	**Demolition**	**Site preparation**	**Site improvement**	**Liquid and gas site utilities**	**Electrical site improvement**	**Site communication**	**Miscellaneous site construction**

Figure 5. Distribution of averaged precision (numbers) and resulted labels of job-type (y-axis) when the original label of input images (x-axis) is given. The numbers in yellow cells indicate the maximum precision of each column

should be classified under "Foundation" are inputted into the classifier, they are classified under the correct category with 31% precision. They could be classified under the "Conveying" category with 27% precision, but it is not considered because job-type inferring only follows the category with the maximum precision.

The classifier shows 92.6% accuracy, and the average precision on correct data is 58.2%. Among the 27 categories, 25 categories were classified properly and two categories were classified incorrectly. While the result of "Communication" is close to those of the adequate categories, that of "Slab on grade" has less precision than 12%. Also, the precision of some job-types is distributed to other job-types that have similar visual characteristics. This indicates the confusion that can occur due to human error can be similarly occurred by deep-learning, and it can be fixed to some degree by complementing more featured images or combining those categories.

5 Conclusion

Although there are lots of pictures taken from construction sites, not enough studies to manage them are conducted yet. We propose a method of classifying pictures taken from construction sites using deep-learning algorithm in order to automate the process.

The performance test shows that a deep-learning algorithm (Google Inception v3) can automatically classify construction pictures into OmniClass Level 2 with more than 90% of accuracy when it was trained with 1,208 images. The major contribution of this study is on suggesting a method of automation in documentation. The result shows that the classifier with deep-learning can do some of documentation task instead of human only after retrained with pictures taken in advance.

In fact, recognizing the process of optimization when a deep learning algorithm is used is difficult. However, it is only based on complex mathematical calculation, not on the black box totally [19][20]. For the appropriate result from deep learning to be obtained, the algorithm should be designed first, and a large volume of data for training should be arranged well. This study used over 1,400 images for training and validation, but more reliable results can be derived if a larger number of images were used during the training and validation. Moreover, this study shows that an automated construction-picture classification method solely based on CNN has room for improvement. This paper reports the first step toward developing a reliable classifier for construction work by using deep learning. We are developing it further to improve its reliability. We are planning to develop a new algorithm based on a semantic approach and deep learning (CNN) to improve performance in terms of precision and accuracy with the use of a larger number of construction pictures.

Acknowledgements

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (2017-0-00701; the 1st year).

References

[1] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks.". *Advances in neural information processing systems*, 2012.

[2] Szegedy, Christian, et al. "Rethinking the inception architecture for computer vision.". *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[3] Kim, Changyoon, Byoungil Kim, and Hyoungkwan Kim. "4D CAD model updating using image processing-based construction progress monitoring.". Automation in Construction, 35: 44-52, 2013.

[4] Zhu, Zhenhua, and Ioannis Brilakis. "Parameter optimization for automated concrete detection in image data." Automation in Construction, 19.7: 944-953, 2010

[5] Szegedy, Christian, et al. "Going deeper with convolutions." Proceedings of the IEEE conference on computer vision and pattern recognition, 2015.

[6] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition.". arXiv preprint arXiv:1409.1556, 2014.

[7] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[8] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." Nature, 521.7553: 436-444, 2015.

[9] Chi, S. and C. H. Caldas. "Automated object identification using optical video cameras on construction sites." Computer-Aided Civil and Infrastructure Engineering, 26(5): 368-380, 2011

[10] Fang, Q., et al. "Detecting non-hardhat-use by a deep learning method from far-field surveillance videos." Automation in Construction, 85(Supplement C): 1-9, 2018

[11] Kern-Isberner, Gabriele. "Characterizing the principle of minimum cross-entropy within a conditional-logical framework.". Artificial Intelligence 98.1-2: 169-208, 1998

[12] Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database.". Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009

[13] Sharif Razavian, Ali, et al. "CNN features off-the-shelf: an astounding baseline for recognition.". Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2014

[14] Dikbas, A., and K. Ercoskun. "Construction information classification: an object oriented paradigm.". eWork and eBusiness in Architecture, Engineering and Construction. ECPPM 2006: European Conference on Product and Process Modelling 2006 (ECPPM 2006), Valencia, Spain, 13-15 September 2006. CRC Press, 2006.

[15] Wu, Yuhong, et al. "Object recognition in construction-site images using 3D CAD-based filtering.". Journal of Computing in Civil Engineering 24.1: 56-64, 2009

[16] Kim, Changmin, Hyojoo Son, and Changwan Kim. "Automated construction progress measurement using a 4D building information model and 3D data.". Automation in Construction 31: 75-82, 2013

[17] Goodfellow, Ian, et al. Deep learning. Vol. 1. Cambridge: MIT press, 2016.

[18] Bottou, Léon. "Large-scale machine learning with stochastic gradient descent." Proceedings of COMPSTAT'2010. Physica-Verlag HD. 177-186, 2010.

[19] Alain, Guillaume, and Yoshua Bengio. "Understanding intermediate layers using linear classifier probes." *arXiv preprint arXiv:1610.01644*, 2016

[20] Shwartz-Ziv, Ravid, and Naftali Tishby. "Opening the black box of deep neural networks via information." arXiv preprint arXiv:1703.00810, 2017