Journal of information and communication convergence engineering
http://jicce.org

J. Inf. Commun. Converg. Eng. 18(2): 75-81, Jun. 2020 Regular paper

Ensemble of Convolution Neural Networks for Driver Smartphone Usage Detection Using Multiple Cameras

Ziyi Zhang and Bo-Yeong Kang*

School of Mechanical Engineering, Kyungpook National University, Daegu 41566, Korea

Abstract

Approximately 1.3 million people die from traffic accidents each year, and smartphone usage while driving is one of the main causes of such accidents. Therefore, detection of smartphone usage by drivers has become an important part of distracted driving detection. Previous studies have used single camera-based methods to collect the driver images. However, smartphone usage detection by employing a single camera can be unsuccessful if the driver occludes the phone. In this paper, we present a driver smartphone usage detection system that uses multiple cameras to collect driver images from different perspectives, and then processes these images with ensemble convolutional neural networks. The ensemble method comprises three individual convolutional neural networks with a simple voting system. Each network provides a distinct image perspective and the voting mechanism selects the final classification. Experimental results verified that the proposed method avoided the limitations observed in single camera-based methods, and achieved 98.96% accuracy on our dataset.

Index Terms: Smartphone usage detection, Multi-camera, Convolutional neural networks, Ensemble

I. INTRODUCTION

The World Health Organization states that approximately 1.3 million people die from traffic accidents each year, and distraction while driving contributes to approximately half of all such cases [1]. The National Highway Transportation Safety Administration has defined distracted driving as indulging in any simultaneous activity that can divert a person's attention away from the primary task of driving, and has classified distractions into manual, visual, and cognitive types [2]. Several surveys have confirmed that smartphone conversations cause considerable cognitive distraction, and reduce the brain activity required for driving by as much as 37%. In particular, sending a text message requires simultaneous visual, manual, and cognitive attention, which can cause the driver to remove their eyesight (and attention) from the road for an average of 4.6 s, raising the risk of a traffic accident during such an activity 23 fold over that while driving normally [3]. Therefore, driver smartphone usage detection is an important task in distracted driving detection.

Several previous studies have proposed non-vision- and computer vision-based methods to detect distracted driving, including detection of driver smartphone usage [3-12]. Non-visual methods are generally intrusive and tend to be used for driver fatigue monitoring [10-12], with computer vision methods being more popular and effective in detecting driver inattention [13]. Advanced driver assistance systems can alert drivers to potential problems, thereby helping to prevent driving accidents. These systems generally use camera sensors to monitor the driving behavior by collecting images and then using the driver's visual features to detect distracted driving, including smartphone usage [3-9]. However, these approaches have various limitations. In the majority of previous studies, a single camera was employed to collect the

Received 13 March 2020, **Revised** 09 June 2020, **Accepted** 09 June 2020
***Corresponding Author** Bo-Yeong Kang (E-mail: kby09@knu.ac.kr, Tel: +82-53-950-7542, +82-53-958-7542)
School of Mechanical Engineering, Kyungpook National University, Daegu 41566, Korea.

Open Access https://doi.org/10.6109/jicce.2020.18.2.75 print ISSN: 2234-8255 online ISSN: 2234-8883

driver images. Hence, the detection of driver smartphone usage was poor when the usage was occurring outside the camera monitoring range or the usage behavior was occluded, making smartphone usage detection extremely challenging.

In this paper, we propose the use of multiple cameras to detect driver smartphone usage. Then, by employing three convolutional neural network (CNN) modules that correspond to the driver images from three different perspectives, the output information from each module is combined using ensemble learning.

The remainder of this paper is organized as follows. In Section 2, relevant previous approaches are briefly introduced and discussed. The structural details and several important components of the proposed system are discussed in Section 3. In Section 4, we present and discuss the experimental performance of the proposed method, and finally in Section 5, we summarize and conclude the paper.

II. RELATED WORK

Previous driver smartphone usage detection studies mainly involved driving inattention studies, i.e., inattention behaviors. However, techniques based on capturing the non-visual features of driver inattention are frequently intrusive; they can be further divided into vehicle parameter or driver physiological analyses [10-12]. Vehicle parameter techniques analyze driver attention through vehicle behavior. Wu et al. [11], for example, developed a prototype of a driving behavior-based event data recorder, which provided driving behavior and danger level information. Experimental results proved that their proposed method achieved 95% average detection ratio for behavior recognition. Meanwhile, the physiological sensor approach measures the physiological features such as brain activity, heart rate, or hand moistness [10, 12], which because of being intrusive, is not currently deployed commercially inside vehicles.

In computer vision-based driving inattention detection methods, a camera is placed in front of the driver and the recorded driver behavior and movement is then analyzed. This approach is considered efficient in assessing driver inattention [3, 4, 6, 8, 9]. Zhao et al. [6] proposed feature extraction for driving postures based on the geronimo-hardin-massopust multiwavelet transform, with application of subsequent 3-layer multilayer perceptron classifiers to recognize four predefined driving posture classes. Experimental results indicated 83.01% and 84.04% accuracy in holdout and cross-validation prediction for talking on a smartphone, respectively. Berri et al. [4] proposed a support vector machine driving distraction detection model to detect smartphone usage while driving. They created a dataset with only the frontal image views of the driver face, and a support vector machine with polynomial kernel achieved 91.57% accuracy for the image detection. Craye and Karray [3] classified the Kinect RGB-D data using the adaboost classifier and hidden Markov models. Their method focused on analyzing the color and depth map data for eye behavior, arm position, head orientation, and facial expressions to detect driver distraction, particularly the distraction type including smartphone usage, and achieved 85% accuracy for that distraction type. In a research study using deep learning [14, 15], Hoang et al. [9] developed a multi-scale faster-RCNN model to detect driver smartphone usage, based on detected hand, phone, and steering wheel location and movements and subsequent extraction of the geometric information to determine if the driver was using a smartphone. Their method achieved 93% accuracy in detecting driver smartphone usage. Baheti et al. [8] proposed a CNN-based system that detected distracted drivers, including those using smartphones. They used a modified VGG-16 architecture and regularization technique to improve the model performance. Experimental results proved that their system achieved 96.31% accuracy. Most previous studies have focused on detecting driver inattention; however, few studies have focused on detecting driver smartphone usage specifically. Craye and Karray [3] reported that the best driver smartphone usage detection studies achieved 85% accuracy. Berri et al. [4] and Hoang et al. [9] achieved 91.57% and 93% accuracy in driver smartphone usage detection, respectively.

As the cited systems used a single camera to collect the driver images, they were incapable of correctly capturing the smartphone usage by the driver when it occurred outside the camera monitoring range, or when the usage was occluded.

To avoid the limitations of single camera systems, we developed a convolutional neural network (CNN) ensemble system for driver smartphone usage detection. The method uses a multi-camera setup to collect driver images from various perspectives, and then processes these different perspectives into a single camera-based driver smartphone usage view, ensembled with CNNs. This not only increases the monitoring range but also effectively avoids any self-occlusion observed in single-camera methods.

III. PROPOSED METHOD

We developed a driver smartphone usage detection system based on a multi-camera setup using a CNN ensemble for image processing. This section details the proposed data collection method and the overall system architecture for the CNN ensemble.

A. Data Collection

Fig. 1 shows the virtual driving platform used for driver image collection in the proposed system. We employed a

Hyundai Grandeur 2007 car model simulator to ensure that the collected data were the closest possible in similarity to the real environment. Fig. 2 depicts three GoPro cameras [14] installed on the driving platform to collect the different driver image perspectives. Camera 1 was set near the sun visor in front of the driver to monitor phone usage by capturing the driver head movements (Fig. 2(a)); Camera 2 was set on top of the center panel to monitor phone usage when in the best driving field of view (Fig. 2(b)); and Camera 3 was set on the right side of the driver to monitor in-car driving behavior (Fig. 2(c)). Fig. 2(d)-(f) show the typical captured images from the corresponding cameras.

Fig. 3 shows the three employed GoPro cameras and the GoPro remote that was used to wirelessly connect with them, which allowed all camera images to be captured in a time synchronized manner. The GoPro interval setting determines the period of time between each captured frame. The default photo interval was 0.5 s. These GoPro cameras had four selectable field of view settings: wide, medium, linear, and narrow. We selected the wide field of view mode to ensure

Fig. 1. Constructed virtual driving platform for driver image collection. Red squares indicate the camera placements.

Fig. 3. Image capturing system, comprising cameras and remote.

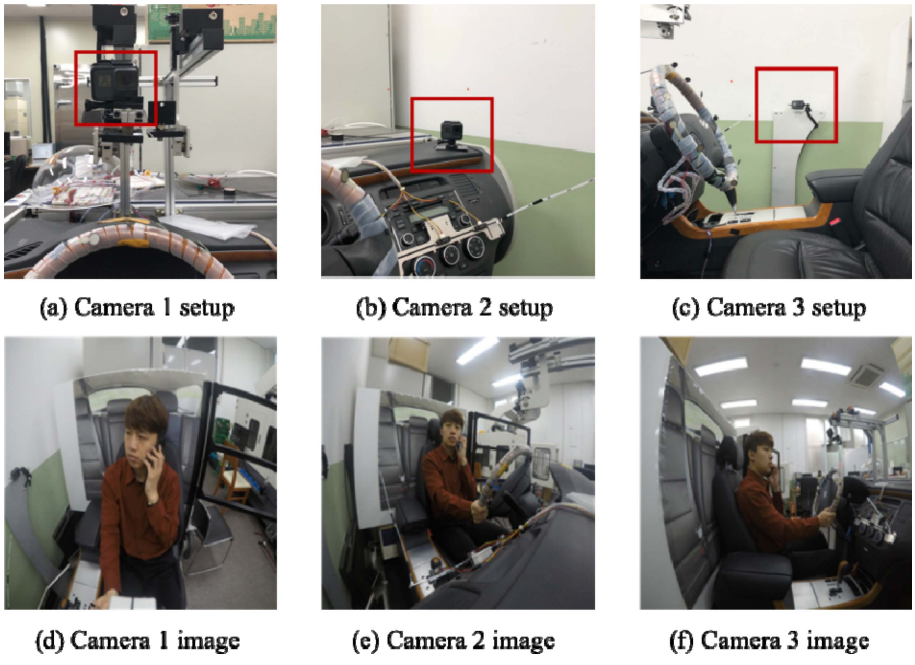


Fig. 2. Camera setups and corresponding camera images.

Fig. 4. Typical driving behavior images captured from perspective 1.

that the images captured the complete upper body of the driver.

Upper body images of three test subjects were collected at 2 frames/s for 60 s under each perspective for capturing eight driving behaviors: (a) normal driving, (b) scratching head, (c) touching ears, (d) rubbing eyes, (e) using smartphone with left hand, (f) using smartphone with right hand, (g) sending message with left hand, and (h) sending message with right hand. Fig. 4 shows the typical driving behavior images acquired from perspective 1. The captured images were then manually divided into the eight behavior categories. The collected image dataset contained 360 images for each behavior type (120 images per camera), for a total of 2,880 images. The initial images, of size 4,000×3,000 pixels, were resized to 227×227, 224×224, and 224×224 pixels to match the AlexNet, GoogLeNet, and ResNet input sizes, respectively, using MATLAB. The collected driving images were manually annotated as normal (driving behaviors (a)-(d)), or phone usage (driving behaviors (e)-(h)), and represented numerically as 0 or 1, respectively. Table 1 summarizes the collected image dataset.

Table 1. Driving behavior image count.

Tagging	Normal				Phone usage			
Data type	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
Camera 1	120	120	120	120	120	120	120	120
Camera 2	120	120	120	120	120	120	120	120
Camera 3	120	120	120	120	120	120	120	120
Total	360	360	360	360	360	360	360	360

B. Multiple Camera-based CNN Ensemble

Fig. 5 shows the proposed multi-camera CNN ensemble, comprising three distinct CNNs, to detect smartphone usage. We used AlexNet [17], GoogLeNet [18], and ResNet [19] networks for each of the three perspective image datasets.

AlexNet comprises five convolutional, three pooling, and three fully connected layers. The 227×227 pixel images were used as input with the last fully connected layer used for classification. We used three AlexNet networks, with modified final fully connected layers corresponding to normal and smartphone usage. All other layers had common structures for each usage case.

GoogLeNet comprises 22 layers with 224×224 pixel image input. Similar to the AlexNet, we modified the last fully connected and final output layers to correspond to normal and smartphone usage, and retained all other layers with common layer architectures.

ResNet-18 accepts 224×224 pixel images as input, and we modified the final connected layer similar to the other two CNN models.

The final layer of each CNN was connected to the prediction layers, with the output being the driver smartphone usage predictions (normal or phone usage) for each image perspective. We applied an ensemble voting layer behind each prediction layer with two possible final predictions: normal (0) and phone usage (1). The ensemble voting layer used majority voting:

Case 1: If a minimum two of the three CNNs produce a negative prediction (0), then the ensemble voting layer output = negative (0).

Case 2: If only one CNN has the negative prediction (0), then the ensemble voting layer output = positive (1).

Different driver image perspectives provide majority voting when driver smartphone usage cannot be extracted from a single perspective, such as when the behavior is outside the camera monitoring range or smartphone usage is occluded. Thus, the majority voting layer increases the confidence for the final prediction. For example, Fig. 5 shows the driver using their left hand to make a phone call. This smartphone usage was captured in Cameras 1 and 2 but occluded in Camera 3 owing to the driver's head.

Hence, the prediction results for the three perspectives were 1 (phone usage) for predictions 1 and 2 and 0 (normal) for prediction 3. The voting system totals the outcomes, and because phone usage was predicted more frequently than normal, the final system prediction = 1 (phone usage).

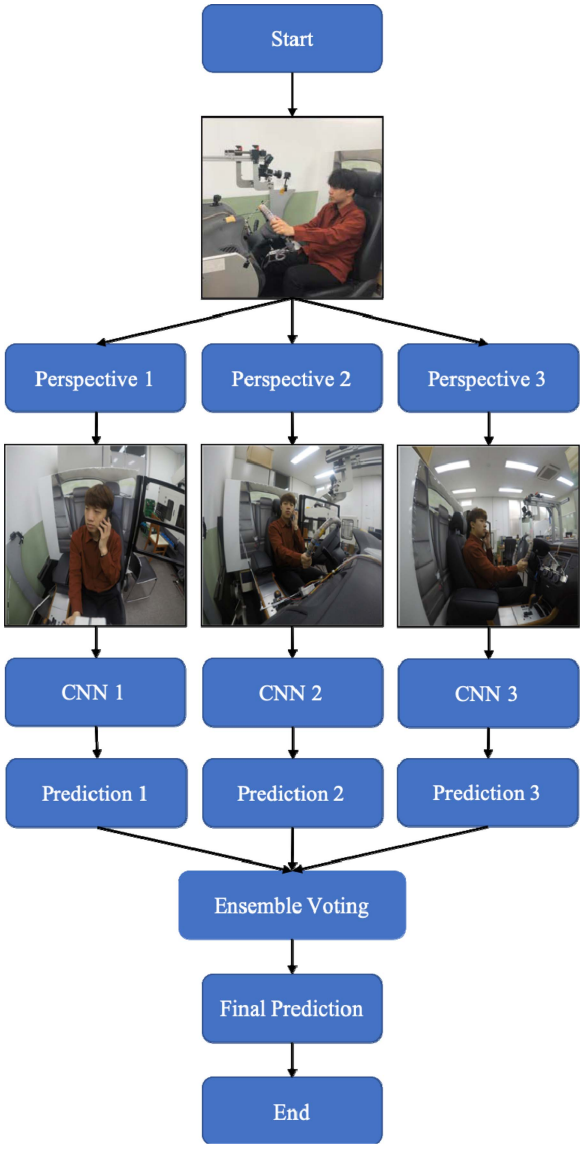


Fig. 5. Proposed CNN ensemble framework for driver smartphone usage detection.

IV. EXPERIMENTAL RESULTS

The proposed CNN ensemble architecture was trained and tested using a computer system with NVIDIA GeForce GTX 745 GPU, 8 GB RAM, and Windows 10. We used the collected image dataset (2,880 images, 960 for each camera) as summarized in Table 1; and applied 10-fold cross-validation, by separating 90% and 10% of the images as the training test sets for each cross-validation stage, respectively. The average results after the cross-validation were used to evaluate the model performance. The models were implemented in MATLAB using the deep learning toolbox. Modified final fully connected layers corresponded to normal and smartphone usage, with the common architecture of all other layers and cross-entropy as the loss function. We trained three networks separately and tested each of the three perspective images for each CNN model simultaneously.

Table 2 shows the confusion matrix for the proposed driver smartphone usage detection system. We used accuracy, recall, and F1 score [20] to evaluate the models. The accuracy was calculated as the overall proportion of the correct predictions of the model, given by,

$$Accuracy = \frac{a+d}{a+b+c+d} \quad (1)$$

Here a is the number of true positives, b is the number of false positives, c is the number of false negatives, and d is the number of true negatives. The recall (also called the true positive rate) is the ratio of the true positive predictions to the total actual positives, given as

$$Recall = \frac{a}{a+c} \quad (2)$$

and F1 score is the weighted average of the precision and recall,

$$F1\ score = 2 * \frac{Recall * Precision}{Recall + \text{Precision}} = 2 * \frac{Recall * \frac{a}{a+b}}{Recall + \frac{a}{a+b}} \quad (3)$$

A. Deep Learning Method Performances for Each Perspective

Table 3 summarizes the accuracy, recall, and F1 score in

Table 2. Confusion matrix for smartphone usage detection system.

System \ Answer	Phone usage	Normal
Phone usage	a	b
Normal	c	d

Table 3. AlexNet performance for each image perspective

Performance	Perspective 1	Perspective 2	Perspective 3
Accuracy	97.81%	**98.54%**	98.15%
Recall	97.14%	**97.22%**	96.97%
F1 score	96.39%	97.50%	**97.52%**

Table 4. GoogLeNet performance for each image perspective

Performance	Perspective 1	Perspective 2	Perspective 3
Accuracy	98.65%	97.60%	**98.75%**
Recall	**99.31%**	96.82%	98.76%
F1 score	96.39%	96.23%	**98.91%**

Table 5. ResNet-18 performance for each image perspective.

Performance	Perspective 1	Perspective 2	Perspective 3
Accuracy	97.92%	**98.34%**	98.22%
Recall	96.31%	**97.74%**	96.08%
F1 score	97.40%	97.50%	**97.81%**

AlexNet for each perspective. The average accuracy was 97.81%, 98.54%, and 98.15% for each perspective, respectively. AlexNet achieved good accuracy for every perspective, with perspective 2 being the most accurate. The average recall was 97.14%, 97.22%, and 96.97% respectively, with perspective 2 displaying the best performance here also. The average F1 score for each perspective was 96.39%, 97.50%, and 97.52% respectively, with perspectives 2 and 3 exhibiting very similar results.

Table 4 shows the corresponding experimental results for GoogLeNet. The deeper GoogLeNet network structure can extract higher level features from the driver images, and perspective 3 achieved the best accuracy of 98.75% here, which is superior to the best accuracy achieved in AlexNet. Perspective 1 achieved the best recall of 99.31% in GoogLeNet, which is significantly superior to AlexNet, and perspective 3 achieved the best F1 score of 97.91%.

Table 5 shows the corresponding experimental results for ResNet-18. Perspective 2 achieved the best performance for all metrics at 98.34%, 97.74%, and 97.50% respectively.

Average accuracy, recall, and F1 score for the datasets differs for each perspective even after using the same CNN, which indicates that the perspective setting has an impact on the driver behavior monitoring. Thus, the multi-camera approach effectively avoided single camera problems, particularly occlusion, and all the included deep learning methods exhibited good results, although there is still room for improvement.

B. CNN Ensemble Performance

Table 6 shows the average accuracy, recall, and F1 scores for each proposed CNN ensemble method. CNN ensemble outcomes outperform each perspective's original outcomes (after comparing Tables 3-5). The GoogLeNet ensemble exhibits the best performance for all three metrics. Thus, the proposed ensemble voting mechanism effectively avoids situations where driver smartphone usage behavior cannot be detected due to occlusion or being outside the individual camera range.

Table 6. Mean CNN ensemble performance.

Performance	Perspective 1	Perspective 2	Perspective 3
Accuracy	98.75%	**98.96%**	98.54%
Recall	97.77%	**99.05%**	97.78%
F1 score	97.79%	**98.33%**	98.23%

The trade-off for achieving this improved accuracy was the significantly higher training time for each deep learning method. However, the extra overhead only applied to training, which can be alleviated by employing a more powerful GPU. The application processing time remained comparable to non-ensemble cases.

V. CONCLUSION

Distracted driving is a major problem that causes a large number of traffic accidents worldwide every year. An advanced driver assistance system can effectively reduce such accidents. This paper presented a CNN ensemble framework, with a simple majority voting mechanism, to identify driver smartphone usage while driving. We collected images that captured driver distraction behavior from multiple perspectives to develop and test the driver smartphone usage detection system. The proposed CNN ensemble framework avoids self-occlusion and other single-camera problems for achieving smartphone usage detection.

In future studies, we plan to develop an effective CNN ensemble framework that incorporates additional image perspectives and/or camera settings, with preprocessing to recognize phone usage gestures to improve the smartphone usage detection performance.

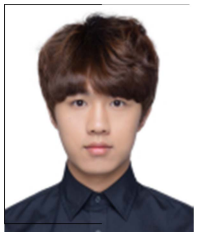
ACKNOWLEDGMENTS

This work was supported by the National Research Foundation of Korea funded by the Korean Government under grant NRF-2019R1A2C1011270.

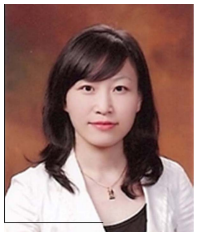
REFERENCES

[1] M. A. Regan, J. D. Lee, and K. Young, *Driver distraction: Theory,*

effects, and mitigation, CRC Press, 2008.
[2] National highway traffic safety administration traffic safety facts [Internet]. Available: https://www.nhtsa.gov/risky-driving/distracted-driving.
[3] C. Craye and F. Karray, Driver distraction detection and recognition using RGB-D sensor, 2015, [online] Available: https://arxiv.org/abs/1502.00250.
[4] R. A. Berri, A. G. Silva, R. S. Parpinelli, E. Girardi, and R. Arthur, "A pattern recognition system for detecting use of mobile phones while driving," *International Conference on Computer Vision Theory and Applications (VISAPP),* IEEE, vol. 2, pp. 411-418, 2014. DOI: 10.5220/0004684504110418.
[5] X. Zhang, N. Zheng, F. Wang, and Y. He, "Visual recognition of driver hand-held cell phone use based on hidden CRF," in *Proceedings of IEEE International Conference on Vehicular Electronics and Safety*, IEEE, pp. 248-251, 2011.
[6] C. Zhao, Y. Gao, J. He, and J. Lian, "Recognition of driving postures by multiwavelet transform and multilayer perceptron classifier," *Engineering Applications of Artificial Intelligenc*e, vol. 25, no. 8, pp. 1677-1686, 2012. DOI: 10.1016/j.engappai.2012.09.018.
[7] D. Wang, M. Pei, and L. Zhu, "Detecting driver use of mobile phone based on in-car camera," in *IEEE Tenth International Conference on Computational Intelligence and Security,* pp. 148-151, 2014.
[8] B. Baheti, S. Gajre, and S. Talbar, "Detection of distracted driver using convolutional neural network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1032-1038, 2018.
[9] T. H. N. Le, Y. Zheng, C. Zhu, K. Luu, and M. Savvides, "Multiple scale faster-rcnn approach to driver's cell-phone usage and hands on steering wheel detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 46-53, 2016.
[10] L. Shiwu, W. Linhong, Y. Zhifa, J. Bingkui, Q. Feiyan, and Y. Zhongkai, "An active driver fatigue identification technique using multiple physiological features," in *IEEE International Conference on Mechatronic Science, Electric Engineering and Computer (MEC)*, pp. 733-737, 2011.
[11] B. F. Wu, Y. H. Chen, and C. H. Yeh, "Driving behaviour-based event data recorder," *IET Intelligent Transport Systems*, vol. 8, no. 4, pp. 361-367, 2013.
[12] N. S. Karuppusamy and B. Y. Kang, "Driver fatigue prediction using eeg for autonomous vehicle," *Advanced Science Letters*, vol. 23, no. 10, pp. 9561-9564, 2017. DOI: 10.1166/asl.2017.9747.
[13] C. Yan, "Vision-based Driver Behaviour Analysis," PhD thesis, University of Liverpool, 2016. [Internet] Available: https://core.ac.uk/reader/80777305.
[14] H. Kim, J. Kim, and H. Jung, "Convolutional Neural Network Based Image Processing System," *Journal of Information and Communication Convergence Engineering*, vol. 16, no. 3, pp. 160-165, 2018. DOI: 10.6109/jicce.2018.16.3.160
[15] V. H. Phung and E. J. Rhee, "A Deep Learning Approach for Classification of Cloud Image Patches on Small Datasets," *Journal of Information and Communication Convergence Engineering*, vol. 16, no. 3, pp. 173-178, 2018. DOI: 10.6109/jicce.2018.16.3.173
[16] GoPro Customer Support. [Internet], Available: https://gopro.com/help/hero5-black.
[17] 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.
[18] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1-9, 2015.
[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778, 2016.
[20] D.M. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," *Journal of Machine Learning Technologies*, vol. 2, issue 1, pp. 37-63, 2011, [online] Available: http://www.bioinfo.in/contents.php?id=51



Ziyi Zhang
is currently studying Master in the School of Mechanical Engineering at Kyungpook National University. He has completed his Bachelor in Mechanical Design Manufacture and Automation at Jilin Institute of Chemical Technology in China. He is currently a researcher in artificial intelligence implementation for social / intelligent robots and farming and industry Robots.



Bo-Yeong Kang
is currently a professor in the School of Mechanical Engineering and the Department of Robot and Smart System Engineering at Kyungpook National University, in Daegu Korea, which she joined in 2009. She received her Ph.D., M.S., M.A. and B.S. from Kyungpook National University. Prior to coming to KNU, she worked at Seoul National University as research professor and KAIST as postdoc. She is currently interested in artificial intelligence implementation for social / intelligent robots.