IATSS Research 43 (2019) 244–252

Contents lists available at ScienceDirect

IATSS Research

Overview

Deep learning-based image recognition for autonomous driving

Hironobu Fujiyoshi *, Tsubasa Hirakawa, Takayoshi Yamashita

Chubu University, 1200 Matsumoto-cho, Kasugai, Aichi 487-8501, Japan

ARTICLE INFO

Article history:
Received 4 September 2019
Received in revised form 19 November 2019
Accepted 26 November 2019
Available online 6 December 2019

Keywords:
Image recognition
Computer vision
Deep learning
Convolutional neural network

ABSTRACT

Various image recognition tasks were handled in the image recognition field prior to 2010 by combining image local features manually designed by researchers (called handcrafted features) and machine learning method. After entering the 2010, However, many image recognition methods that use deep learning have been proposed. The image recognition methods using deep learning are far superior to the methods used prior to the appearance of deep learning in general object recognition competitions. Hence, this paper will explain how deep learning is applied to the field of image recognition, and will also explain the latest trends of deep learning-based autonomous driving.

1. Introduction

In the late 1990s, it became possible to process a large amount of data at high speed with the evolution of general-purpose computers. The mainstream method was to extract a feature vector (called the image local features) from the image and apply a machine learning method to perform image recognition. Supervised machine learning requires a large amount of class-labeled training samples, but it does not require researchers to design some rules as in the case of rule-based methods. So, versatile image recognition can be realized. In the 2000 era, handcrafted features such as scale-invariant feature transform (SIFT) and histogram of oriented gradients (HOG) as image local features, designed based on the knowledge of researchers, have been actively researched. By combining the image local features with machine learning, practical applications of image recognition technology have advanced, as represented by face detection. Next, in the late 2010s, deep learning to perform feature extraction process through learning has come under the spotlight. A handcrafted feature is not necessarily optimal because it extracts and expresses feature values using a designed algorithm based on the knowledge of researchers. Deep learning is an approach that can automate the feature extraction process and is effective for image recognition. Deep learning has accomplished impressive results in the general object recognition competitions, and the use of image recognition required for autonomous driving (such as object detection and semantic segmentation) is in progress. This paper explains how deep learning is applied to each task in image recognition and how it is solved, and describes the trend of deep learning-based autonomous driving and related problems.

2. Problem setting in image recognition

In conventional machine learning (here, it is defined as a method prior to the time when deep learning gained attention), it is difficult to directly solve general object recognition tasks from the input image. This problem can be solved by distinguishing the tasks of image identification, image classification, object detection, scene understanding, and specific object recognition, as shown in Fig. 1. Definitions of each task and approaches to each task are described below.

2.1. Image verification

Image verification is a problem to check whether the object in the image is the same as the reference pattern. In image verification, the distance between the feature vector of the reference pattern and the feature vector of the input image is calculated. If the distance value is less than a certain value, the images are determined as identical, and if the value is more, it is determined otherwise. Fingerprint, face, and person identification relates to tasks in which it is required to determine whether an actual person is another person. In deep learning, the

* Corresponding author.
E-mail address: fujiyoshi@isc.chubu.ac.jp (H. Fujiyoshi).

Peer review under responsibility of International Association of Traffic and Safety Sciences.

https://doi.org/10.1016/j.iatssr.2019.11.008

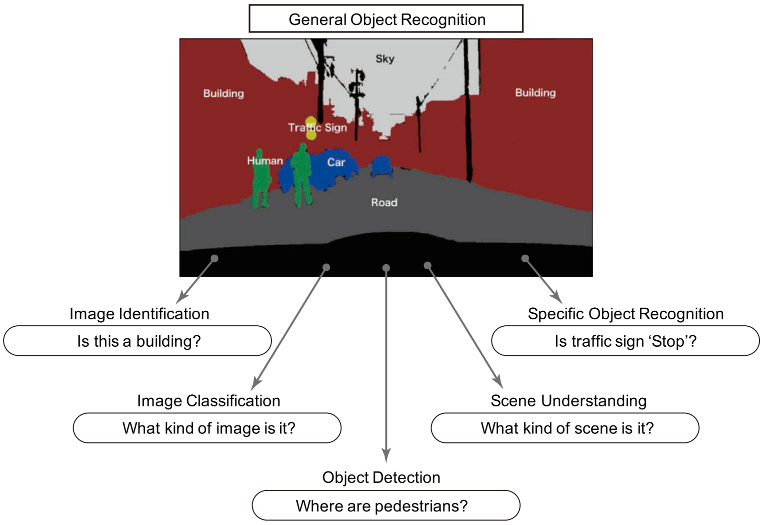


Fig. 1. Segmentation of general object recognition.

problem of person identification is solved by designing a loss function (triplet loss function) that calculates the value of distance between two images of the same person as small, and the value of distance with another person's image as large [1].

2.2. Object detection

Object detection is the problem of finding the location of an object of a certain category in the image. Practical face detection and pedestrian detection are included in this task. Face detection uses a combination of Haar-like features [2] and AdaBoost, and pedestrian detection uses HOG features [3] and support vector machine (SVM). In conventional machine learning, object detection is achieved by training 2-class classifiers corresponding to a certain category and raster scanning in the image. In deep learning-based object detection, multiclass object detection targeting several categories can be achieved with one network.

2.3. Image classification

Image classification is a problem to find out the category to which an object in an image belongs to, among predefined categories. In the conventional machine learning, an approach called bag-of-features (BoF) [4] has been used: a vector quantifies the image local features and expresses the features of the whole image as a histogram. Yet, deep learning is well-suited to the image classification task, and became popular in 2015 by achieving an accuracy exceeding human recognition performance in the 1000-class image classification task.

2.4. Scene understanding (semantic segmentation)

Scene understanding is the problem of understanding the scene structure in an image. Above all, semantic segmentation that finds object categories in each pixel in an image has been considered difficult to solve using conventional machine learning. Therefore, it has been regarded as one of the ultimate problems of computer vision, but it has been shown that it is a problem that can be solved by applying deep learning.

2.5. Specific object recognition

Specific object recognition is the problem of finding a specific object. By giving attributes to objects with proper nouns, specific object recognition is defined as a subtask of the general object recognition problem. Specific object recognition is achieved by detecting feature points using SIFT [5] from images, and a voting process based on the calculation of distance from feature points of reference patterns. Machine learning is not used here directly here, but the learned invariant feature transform (LIFT) [9] proposed in 2016 achieved an improvement in performance by learning and replacing each process in SIFT through deep learning.

3. Deep learning-based image recognition

Image recognition prior to deep learning is not always optimal because image features are extracted and expressed using an algorithm designed based on the knowledge of researchers, which is called a handcrafted feature. Convolutional neural network (CNN) [7]), which is one type of deep learning, is an approach for learning classification and feature extraction from training samples, as shown in Fig. 2. This chapter describes CNN, focuses on object detection and scene understanding (semantic segmentation) introduced in Chapter 2, and describes its application to image recognition and its trends.

3.1. Convolutional Neural Network (CNN)

As shown in Fig. 3, CNN computes the feature map corresponding to the kernel by convoluting the kernel (weight filter) on the input image. Feature maps corresponding to the kernel types can be computed as there are multiple kernels. Next, the size of the feature map is reduced by the pooling feature map. As a result, it is possible to absorb geometrical variations such as slight translation and rotation of the input image. The convolution process and the pooling process are applied repeatedly to extract the feature map. The extracted feature map is input to fully-connected layers, and the probability of each class is finally output. In this case, the input layer and the output layer have a network structure that has units for the image and the number of classes.

Training of CNN is achieved by updating the parameters of the network by the backpropagation method. The parameters in CNN refer to the kernel of the convolutional layer and the weights of all coupled layers. The process flow of the backpropagation method is shown in Fig. 3. First, training data is input to the network using the current parameters to obtain the predictions (forward propagation). The error is calculated from the predictions and the training label; the update amount of each parameter is obtained from the error, and each parameter in the network is updated from the output layer toward the input layer (back propagation). Training of CNN refers to repeating these processes to acquire good parameters that can recognize the images correctly.

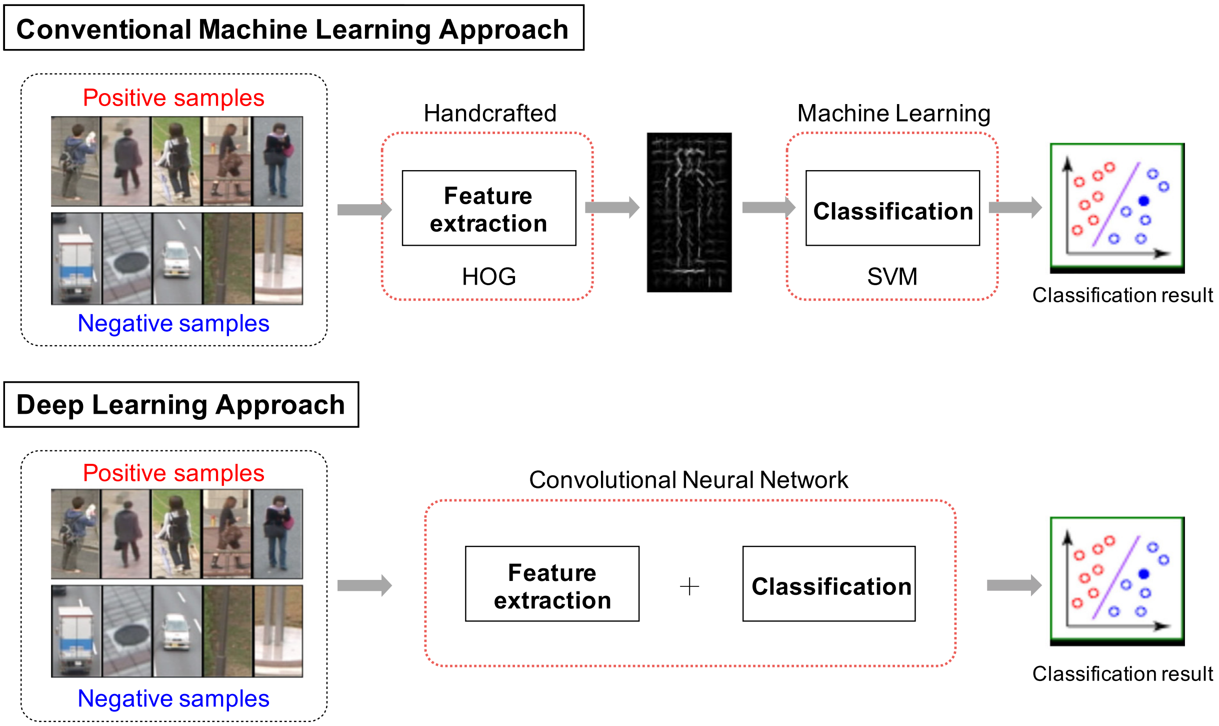


Fig. 2. Conventional machine learning and deep learning.

3.2. Advantages of CNN compared to conventional machine learning

Fig. 4 shows some visualization examples of kernels at the first convolution layer of the AlexNet, which is designed for 1000 object class classification task at ILSVRC(ImageNet Large Scale Visual Recognition Challenge) 2012. AlexNet consists of five convolution layers and three fully-connected layers, whose output layer has 10,000 units corresponding to the number of classes. We see that the AlexNet has automatically acquired various filters that extract edge, texture, and color information with directional components. We investigated the effectiveness of the CNN filter as a local image feature by comparing the HOG in the human detection task. The detection miss rate for CNN filters is 3%, while the HOG is 8%. Although the CNN kernels of the AlexNet not trained for the human detection task, the detection accuracy improved over the HOG feature that is the traditional handcrafted feature.

As shown in Fig. 5, CNN can perform not only image classification but also object detection and semantic segmentation by designing the output layer according to each task of image recognition. For example, if the

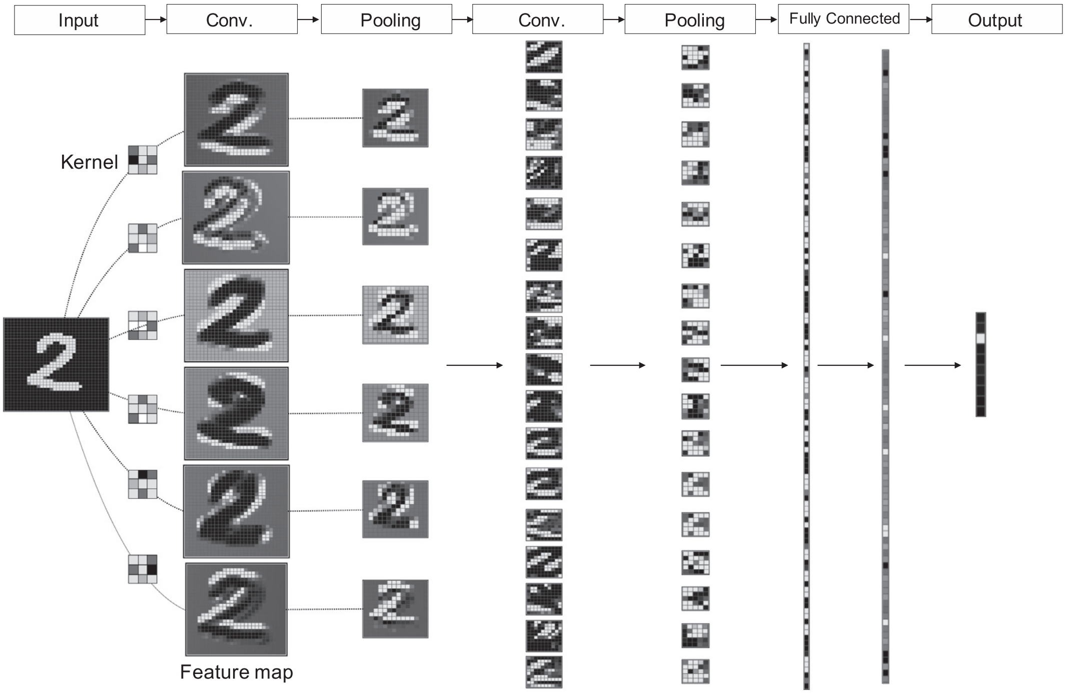


Fig. 3. Basic structure of CNN.

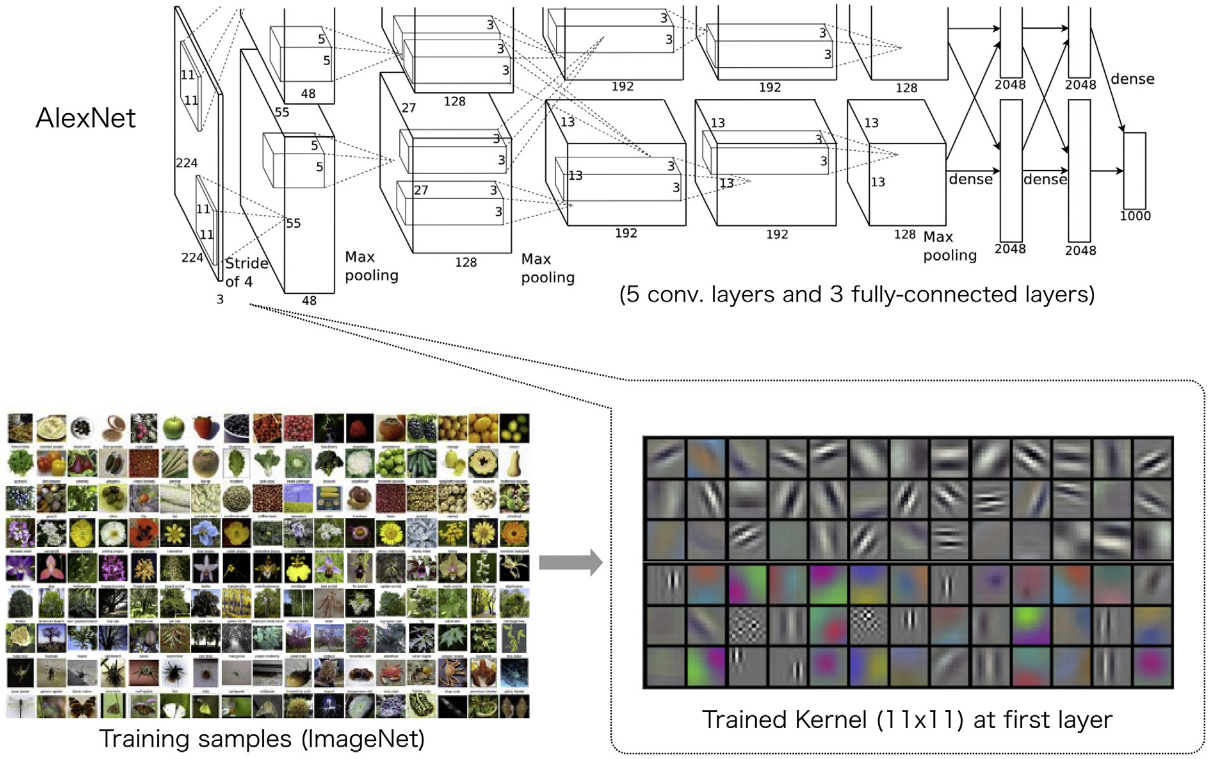
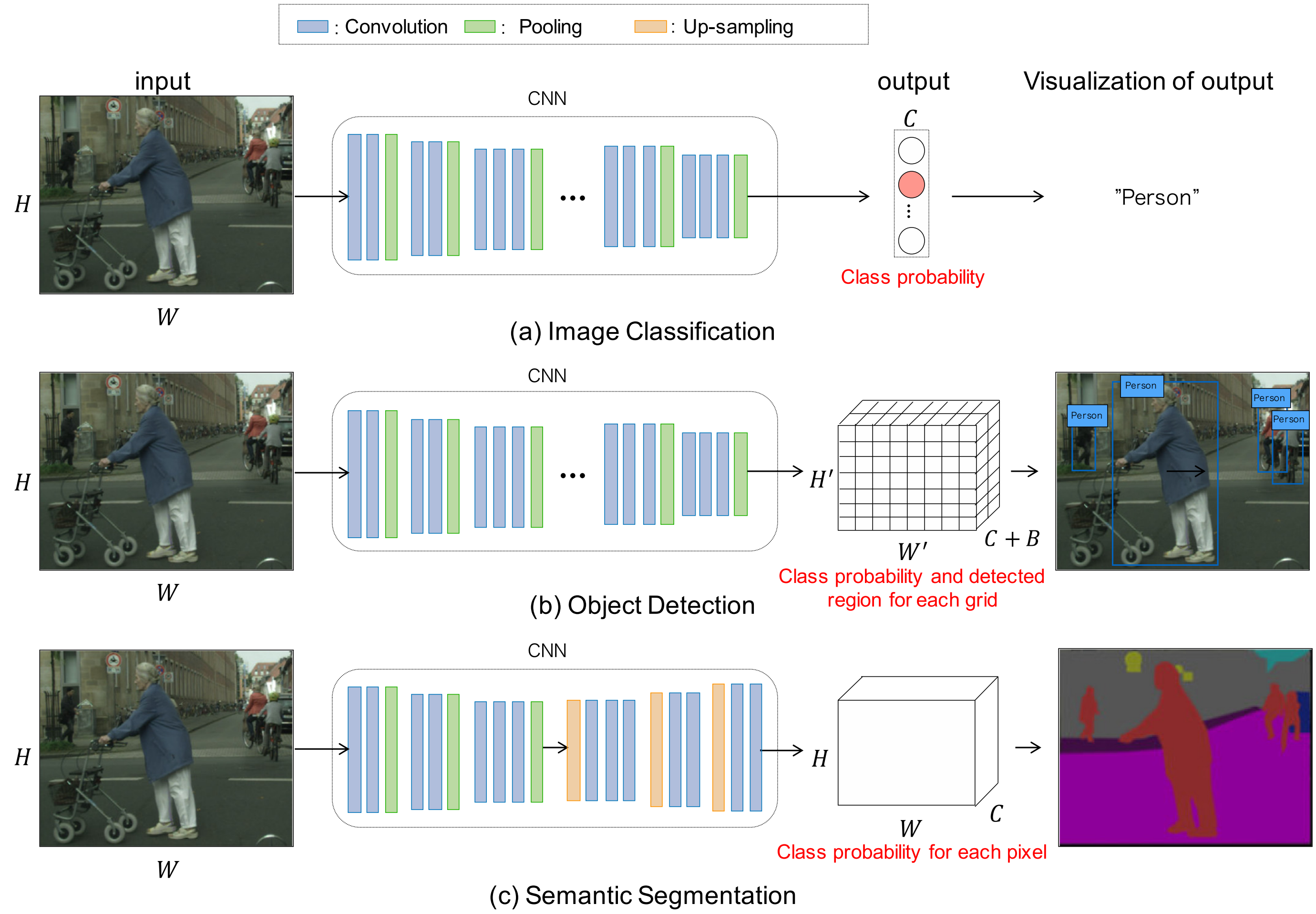


Fig. 4. Network structure of AlexNet and Kernels.



(a) Image Classification

(b) Object Detection

(c) Semantic Segmentation

Fig. 5. Application of CNN to each image recognition task.

output layer is designed to output class probability and detection region for each grid, it will become a network structure that can perform object detection. In semantic segmentation, the output layer should be designed to output the class probability for each pixel. Convolution and pooling layers can be used as common modules for these tasks. On the other hand, in the conventional machine learning method, it was necessary to design image local features for each task and combine it with machine learning. CNN has the flexibility to be applied to various tasks by changing the network structure, and this property is a great advantage in achieving image recognition.

3.3. Application of CNN to object detection task

Conventional machine learning-based object detection is an approach that raster scans two class classifiers. In this case, because the aspect ratio of the object to be detected is constant, it will be object detection of only a certain category learned as a positive sample. On the other hand, in object detection using CNN, object proposal regions with different aspect are detected by CNN, and multiclass object detection is possible using the Region Proposal approach that performs multiclass classification with CNN for each detected region. Faster R-CNN [8] introduces Region Proposal Network (RPN) as shown in Fig. 6, and simultaneously detects object candidate regions and recognizes object classes in those regions. First, convolution processing is performed on the entire input image to obtain a feature map. In RPN, an object is detected by raster scanning the detection window on the obtained feature map. In raster scanning, detection windows in the form of k number of shapes are applied centered on focused areas known as anchor. The region specified by the anchor is input to RPN, and the score of object likeness and the detected coordinates on the input image are output. In addition, the region specified by the anchor is also input to another all-connected network, and object recognition is performed when it is determined to be an object by RPN. Therefore, the unit of the output layer is the number obtained by adding the number of classes and ((x, y, w, h) × number of classes) to one rectangle. These Region Proposal methods have made it possible to detect multiple classes of objects with different aspect ratios.

In 2016, the single-shot method was proposed as a new multiclass object detection approach. This is a method to detect multiple objects only by giving the whole image to CNN without raster scanning the image. YOLO (You Only Look Once) is a representative method [9] in which an object rectangle and an object category is output for each local region divided by a 7 × 7 grid, as shown in Fig. 6. First, feature maps are generated through convolution and pooling of input images. The position (i, j) of each channel of the obtained feature map (7 × 7 × 1024) is a structure that becomes a region feature corresponding to the grid (i, j) of the input image, and this feature map is input to fully connected layers. The output values obtained through fully connected layers are the score (20 categories) of the object category at each grid position and the position, size, and reliability of the two object rectangles. Therefore, the unit of the output layer is the number (1470) in which the position, size, and reliability ((x, y, w, h, reliability) × 2) of two object rectangles is added to the number of categories (20 categories) and multiplied with the number of grids (7 × 7). In YOLO, it is not necessary to detect object region candidates such as Faster R-CNN; therefore, object detection can be performed in real time. Fig. 7 shows an example of YOLO-based multiclass object detection.

3.4. Application of CNN to semantic segmentation

Semantic segmentation is a difficult task and has been studied for many years in the field of computer vision. However, as with other tasks, deep learning-based methods have been proposed and achieved much higher performance than conventional machine learning methods. Fully convolutional network (FCN) [10] is a method that enables end-to-end learning and can obtain segmentation results using only CNN. The structure of FCN is shown in Fig. 8. The FCN has a network structure that does not have a fully-connected layer. The size of the generated feature map is reduced by repeatedly performing the convolutional layer and the pooling layer on the input image. To make it the same size as the original image, the feature map is enlarged 32 times in the final layer, and convolution processing is performed. This is called deconvolution. The final layer outputs the probability map of each class. The probability map is trained so that the probability of the class in each pixel is obtained, and the unit of output of the end-to-end segmentation model is (w × h × number of classes). Generally, the feature map of the middle layer of CNN captures more detailed information as it is closer to the input layer, and the pooling process integrates these pieces of information, resulting in the loss of detailed information. When this feature map is expanded, coarse segmentation results are obtained. Therefore, high accuracy is achieved by integrating and using the feature map of the middle layer. Additionally, FCN performs processing to integrate feature maps in the middle of the network. Convolution process is performed by connecting mid-feature maps in the channel direction, and segmentation results of the same size as the original image are output.

When expanding the feature map obtained on the encoder side, PSPNet [11] can capture information of different scales by using the Pyramid Pooling Module, which expands at multiple scales. The Pyramid Pooling Module is used to pool feature maps with 1 × 1, 2 × 2, 3 × 3, 3 × 6 × 6 in which the vertical and horizontal sizes of the original image are reduced to 1/8, respectively, on the encoder side. Then, convolution process is performed on each feature map. Next, the convolution process is performed and probability maps of each class are output after expanding and linking feature maps to the same size. PSPNet is the method that won in the “Scene parsing” category of ILSVRC held in 2016. Also, high accuracy has been achieved with the Cityscapes Dataset [12] taken with a dashboard camera. Fig. 9 shows the result of PSPNet-based semantic segmentation.

3.5. CNN for ADAS application

The machine learning technique is applicable to use for system intelligence implementation in ADAS(Advanced Driving Assistance System)

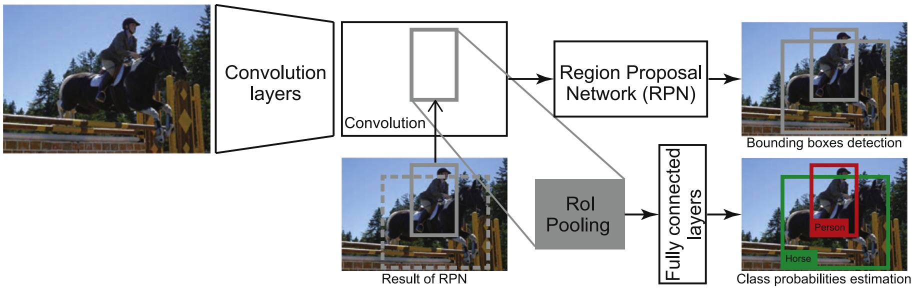


Fig. 6. Faster R-CNN structure.

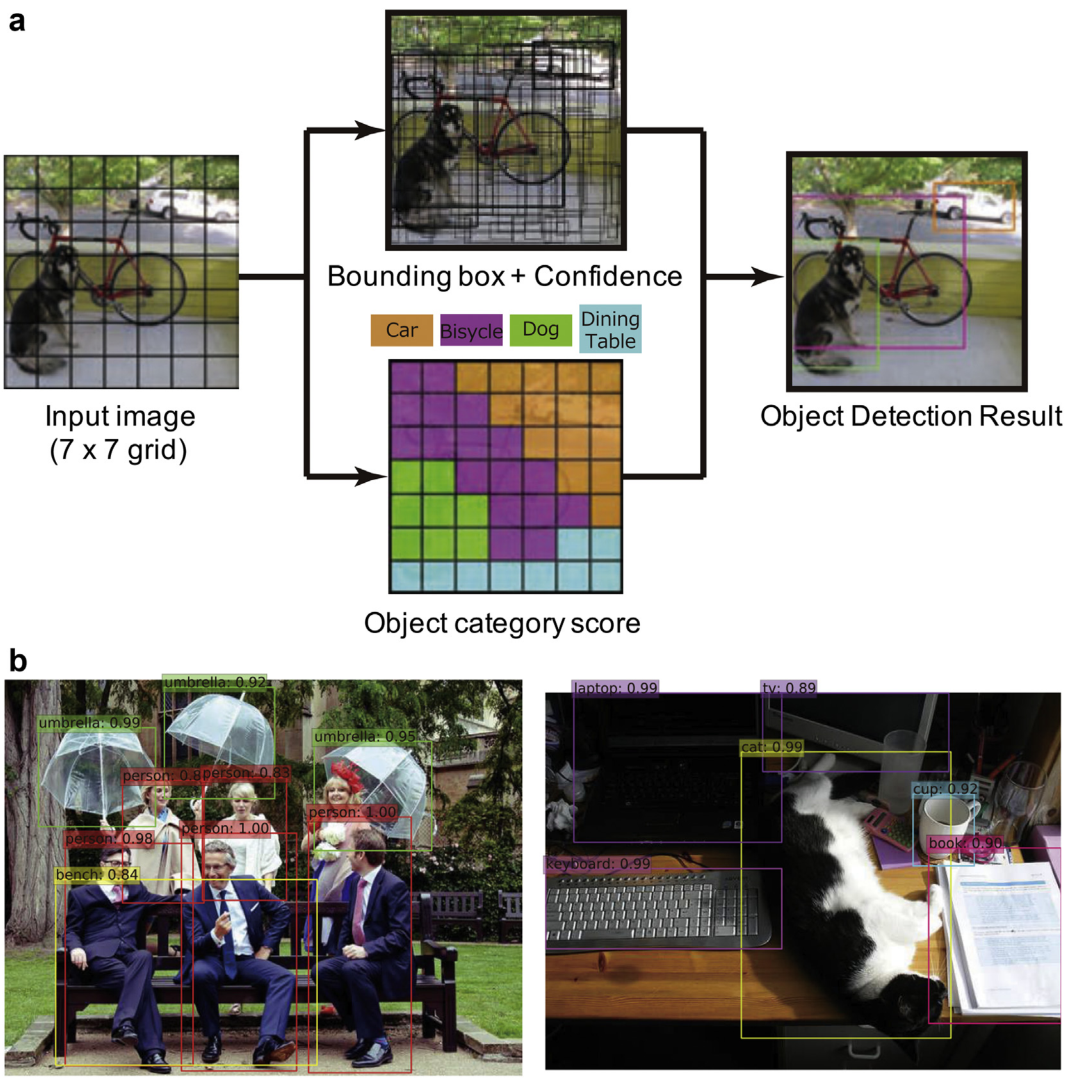


Fig. 7. YOLO structure and examples of multiclass object detection.

[13]. In ADAS, it is to facilitate the driver with the latest surrounding information obtained by sonar, radar, and cameras. Although ADAS typically utilizes radar and sonar for long-range detection, CNN-based systems can recently play a significant role in pedestrian detection, lane detection, and redundant object detection at moderate distances.

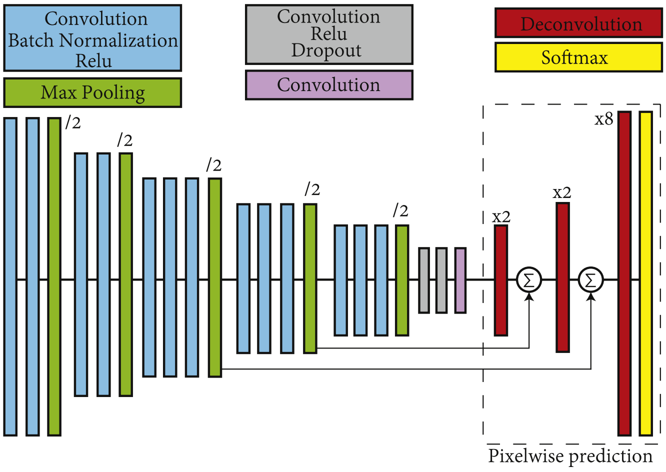


Fig. 8. Fully Convolutional Network (FCN) Structure.

For autonomous driving, the core component can be categorized into three categories, namely perception, planning, and control [14]. Perception refers to the understanding of the environment, such as where obstacles located, detection of road signs/marking, and categorizing objects by their semantic labels such as pedestrians, bikes, and vehicles. Localization refers to the ability of the autonomous vehicle to determine its position in the environment. Planning refers to the process of making decisions in order to achieve the vehicle's goals, typically to bring the vehicle from a start location to a goal location while avoiding obstacles and optimizing the trajectory. Finally, the control refers to the vehicle's ability to execute the planned actions. CNN-based object detection is suitable for the perception because it can handle the multi-class objects. Also, semantic segmentation is useful information for making decisions in planning to avoid the obstacles by referring to pixels categorized as road.

4. Deep learning-based autonomous driving

This chapter introduces end-to-end learning that can infer the control value of the vehicle directly from the input image as the use of deep learning for autonomous driving, and describes visual explanation

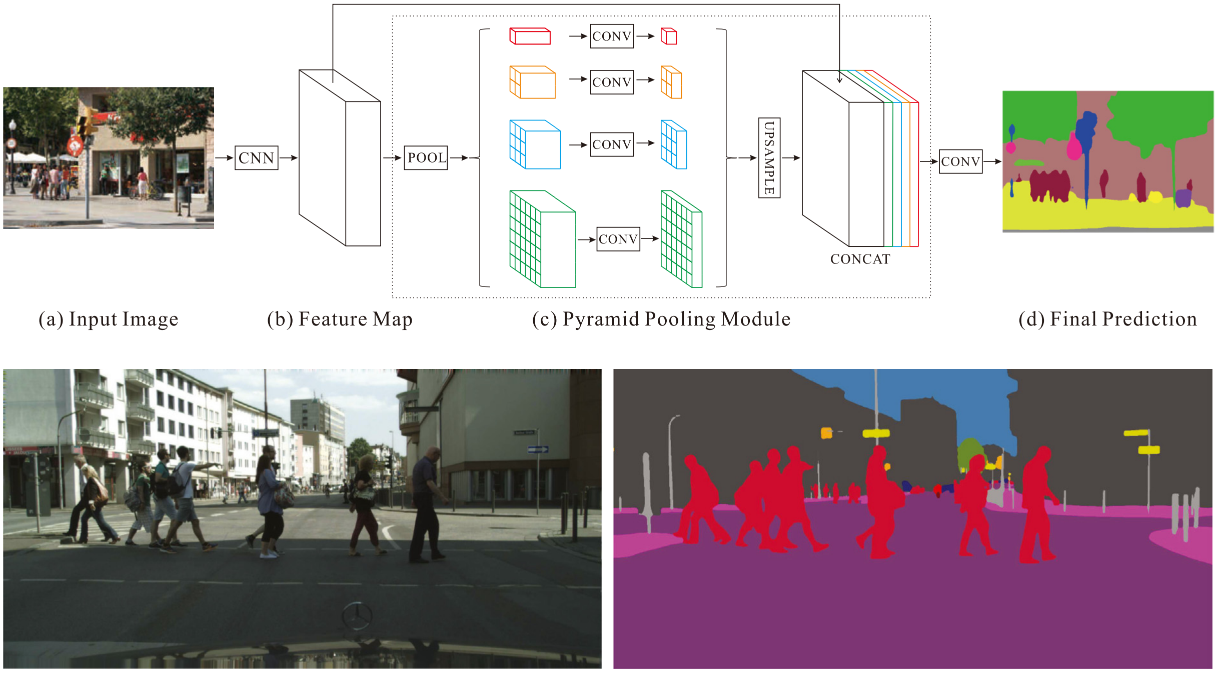


Fig. 9. Example of PSPNet-based Semantic Segmentation Results (cited from Reference [11]).

of judgment grounds that is the problem of deep learning models and future challenges.

4.1. End-to-end learning-based autonomous driving

In most of the research on autonomous driving, the environment around the vehicle is understood using a dashboard camera and Light Detection and Ranging (LiDAR), appropriate traveling position is determined by motion planning, and the control value of the vehicle is determined [18,19]. Autonomous driving based on these three processes is common, and deep learning-based object detection and semantic segmentation introduced in Chapter 3 are beginning to be used to understand the surrounding environment. On the other hand, with the progress in CNN research, end-to-end learning-based method has been proposed that can infer the control value of the vehicle directly from the input image [15,16,20]. In these methods, network is trained by using the images of the dashboard camera when driven by a person, and the vehicle control value corresponding to each frame as learning data. End-to-end learning-based autonomous driving control has the advantage that the system configuration is simplified because CNN learns automatically and consistently without explicit understanding of the surrounding environment and motion planning.

To this end, Bojarski et al. proposed an end-to-end learning method for autonomous driving, which input dash-board camera images into a CNN and outputs steering angle directory [15]. Started by this work, several works have been conducted: a method considering temporal structure of a dash-board camera video [20] or a method to train CNN by using a driving simulator and use the trained network to control vehicle under real environment [16]. These methods basically control only steering angle and throttle (i.e., accelerator and brake) is controlled by human. According to autonomous driving model in Reference [17], it infers not only the steering but also the throttle as the control value of the vehicle. The network structure is composed of five layers of convolutional layers through pooling process and three layers of fully-connected layers. In addition, the inference is made in consideration of one's own state by giving the vehicle speed to the fully-connected layer in addition to the dashboard images, since it is necessary to infer the change of speed in one's own vehicle for throttle control. In this manner, high-precision control of steering and throttle can be achieved in various driving scenarios.

4.2. Visual explanation of end-to-end learning

CNN-based end-to-end learning has a problem where the basis of output control value is not known. To address this problem, research is being conducted on an approach on the judgment grounds (such as turning steering wheel to the left or right and stepping on brakes) that can be understood by humans.

The common approach to clarify the reason of the network decision-making is a visual explanation [17,21,22]. Visual explanation method outputs an attention map that visualizes the region in which the network focused as a heat map. Based on the obtained attention map, we can analyze and understand the reason of the decision-making. To obtain more explainable and clearer attention map for efficient visual explanation, a number of methods have been proposed in the computer vision field. Class activation mapping (CAM) [21] generates attention maps by weighting the feature maps obtained from the last convolutional layer in a network. A gradient-weighted class activation mapping (Grad-CAM) [22] is another common method, which generates an attention map by using gradient values calculated at backpropagation process. This method is widely used for a general analysis of CNNs because it can be applied to any networks. Fig. 10 shows example attention maps of CAM and Grad-CAM.

Visual explanation methods have been developed for general image recognition tasks while visual explanation for autonomous driving has been also proposed [17,23,24]. Visualbackprop [23] is developed for visualize the intermediate values in a CNN, which accumulates feature maps for each convolutional layer to a single map. This enables us to understand where the network highly responds to the input image. Reference [17] proposes a Regression-type Attention Branch Network in which a CNN is divided into a feature extractor and a regression branch, as shown in Fig. 11, with an attention branch inserted that outputs an attention map that serves as a visual explanation. By providing vehicle speed in fully connected layers and through end-to-end learning of each branch of Regression-type Attention Branch Network, control values for steering and throttle for various scenes can be output, and

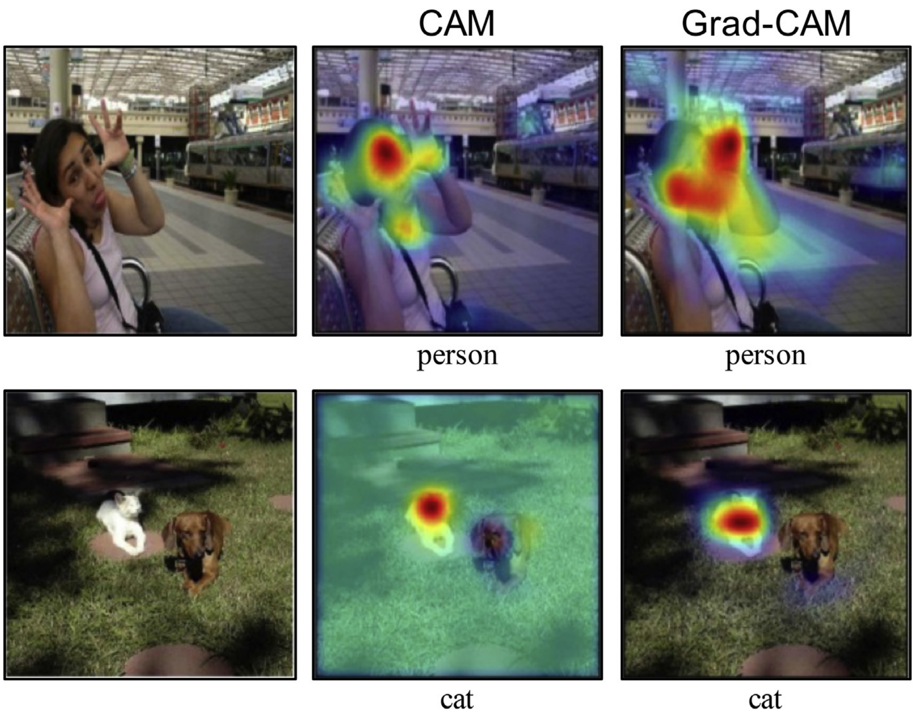


Fig. 10. Example attention maps of CAM and Grad-CAM. (cite from reference [22]).

also output the attention map that describes the location in which the control value was output on the input image. Fig. 12 shows an example of visualization of attention map during Regression-type Attention Branch Network-based autonomous driving. S and T in the figure is the steering value and throttle value, respectively. Fig. 12 (a) shows a scene where the road curves to the right where there is a strong response to the center line of the road, and the steering output value is a positive value indicating the right direction. On the other hand, Fig. 12 (b) is a scene where the road curves to the left, the steering output value is a negative value indicating the left direction, and the attention map responds strongly to the white line on the right. By visualizing the attention map in this way, it can be said that the center line of the road and the position of the lane are observed for estimation of the steering value. Also, in the scene where the car stops as shown in Fig. 12 (c), the attention map strongly responds to the brake lamp of the vehicle ahead. The throttle output is 0, which indicates that the accelerator and the brake are not pressed. Therefore, it is understood that the condition of the vehicle ahead is closely watched in the determination of the throttle. In addition, the night travel scenario in Fig. 12 (d) shows a scene of following a car ahead, and it can be seen that the attention map strongly responds to the car ahead because the road shape ahead is unknown. It is possible to visually explain the judgment grounds through output of attention map in this way.

4.3. Future challenges

The visual explanations enable us to analyze and understand the internal state of deep neural networks, which is efficient for engineers and researchers. One of the future challenges is explanation for end users, i.e., passengers on a self-driving vehicle. In case of fully autonomous driving, for instance, when lanes are suddenly changed even when there are no vehicles ahead or on the side, the passenger in the car may be concerned as to why the lanes were changed. In such cases, the attention map visualization technology introduced in Section 4.2 enables people to understand the reason for changing lanes. However, visualizing the attention map in a fully automated vehicle does not make sense unless a person on the autonomous vehicle always sees it. A person in an autonomous car, that is, a person who receives the full

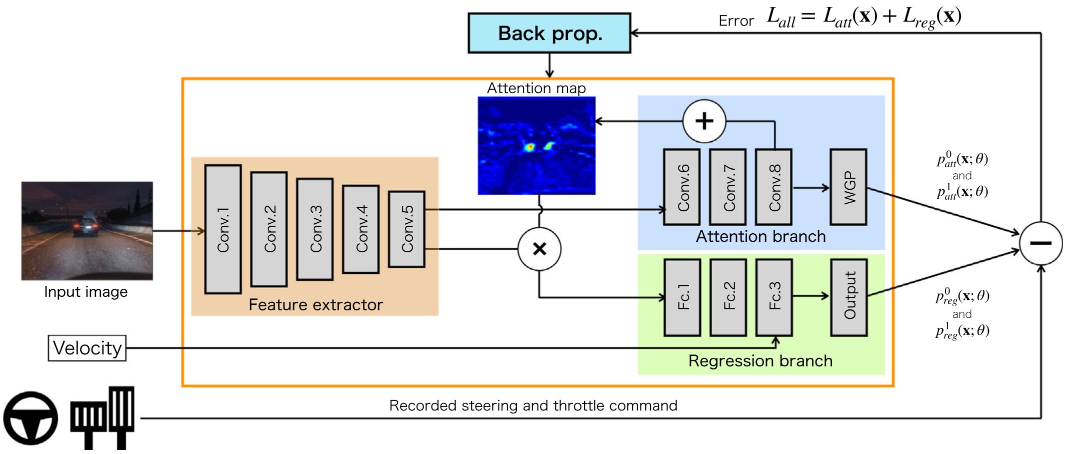

Back prop.
Error $L_{all} = L_{att}(\mathbf{x}) + L_{reg}(\mathbf{x})$
Attention map
Input image
Conv.1
Conv.2
Conv.3
Conv.4
Conv.5
Feature extractor
+
×
Conv.6
Conv.7
Conv.8
WGP
Attention branch
Fc.1
Fc.2
Fc.3
Output
Regression branch
$p^0_{att}(\mathbf{x};\theta)$
and
$p^1_{att}(\mathbf{x};\theta)$
$p^0_{reg}(\mathbf{x};\theta)$
and
$p^1_{reg}(\mathbf{x};\theta)$
−
Velocity
Recorded steering and throttle command

Fig. 11. Regression-type Attention Branch Network. (cite from reference [17].

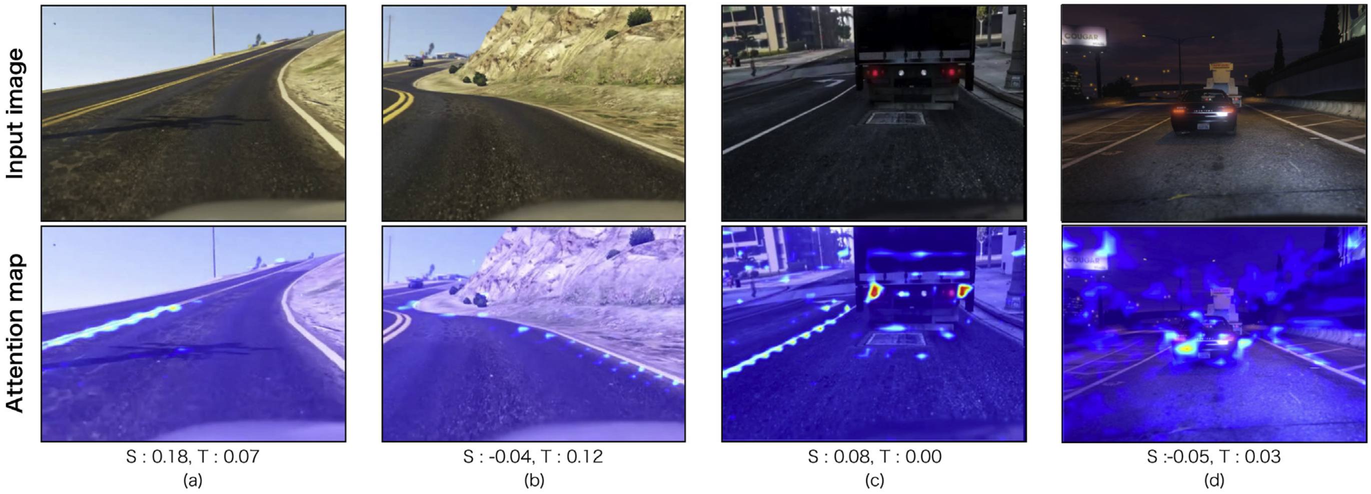


Fig. 12. Attention map-based visual explanation for self-driving.

benefit of AI, needs to be informed of the judgment grounds in the form of text or voice stating, “Changing to left lane as a vehicle from the rear is approaching with speed.” Transitioning from recognition results and visual explanation to verbal explanation will be the challenges to confront in the future. In spite of the fact that several attempts have been conducted for this purpose [17,25,26], it does not still achieve sufficient accuracy and flexible verbal explanations.

Also, in the more distant future, such verbal explanation functions will eventually not be used. At first, people who receive the full benefit of autonomous driving find it difficult to accept, but a sense of trust will be gradually created by repeating the verbal explanations. Thus, if confidence is established between autonomous driving AI and the person, the verbal explanation functions will not be required, and it can be expected that AI-based autonomous driving will be widely and generally accepted.

5. Conclusion

This paper explains how deep learning is applied in image recognition tasks and introduces the latest image recognition technology using deep learning. Image recognition technology using deep learning is the problem of finding an appropriate mapping function from a large amount of data and teacher labels. Further, it is possible to solve several problems simultaneously by using multitask learning. Future prospects not only include “recognition” for input images, but also high expectations for the development of end-to-end learning and deep reinforcement learning technologies for “judgment” and “control” of autonomous vehicles. Moreover, citing judgment grounds for output of deep learning and deep reinforcement learning is a major challenge in practical application, and it is desirable to expand from visual explanation to verbal explanation through integration with natural language processing.

References

[1] D. Cheng, Y. Gong, S. Zhou, J. Wang, N. Zheng, Person re-identification by multi-channel parts-based cnn with improved triplet loss function, Proc. of IEEE Conference on Computer Vision and Pattern Recognition (27-30 June 2016) https://doi.org/10.1109/CVPR.2016.149.

[2] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, Proc. of IEEE Conference on Computer Vision and Pattern Recognition (8-14 Dec. 2001) https://doi.org/10.1109/CVPR.2001.990517.

[3] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, Proc. of IEEE Conference on Computer Vision and Pattern Recognition (20-25 June 2005) https://doi.org/10.1109/CVPR.2005.177.

[4] G. Csurka, C. Dance, L. Fan, J. Willamowski, C. Bray, Visual categorization with bags of keypoints, Proc. of ECCV Workshop on Statistical Learning in Computer Vision, 2004.

[5] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2004) 91–110.

[7] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (1998) 2278–2324.

[8] S. Ren, K. He, R. Girshick, J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” in IEEE Transactions on Pattern Analysis and Machine Intelligence, 39 (6) (1 June 2017) 1137–1149, https://doi.org/10.1109/TPAMI.2016.2577031.

[9] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, Proc. of IEEE Conference on Computer Vision and Pattern Recognition (27-30 June 2016) https://doi.org/10.1109/CVPR.2016.91.

[10] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, Proc. of IEEE Conference on Computer Vision and Pattern Recognition (7-12 June 2015) https://doi.org/10.1109/CVPR.2015.7298965.

[11] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, Proc. of IEEE Conference on Computer Vision and Pattern Recognition (2017).

[12] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The cityscapes dataset for semantic urban scene understanding, Proc. of IEEE Conference on Computer Vision and Pattern Recognition (27-30 June 2016) https://doi.org/10.1109/CVPR.2016.350.

[13] A. Moujahid, M.E. Tantaoui, M.D. Hina, A. Soukane, A. Ortalda, A. ElKhadimi, A. Ramdane-Cherif, Machine learning techniques in ADAS: a review, Proc. of 2018 International Conference on Advances in Computing and Communication Engineering (22-23 June 2018) https://doi.org/10.1109/ICACCE.2018.8441758.

[14] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J.Z. Kolter, D. Langer, O. Pink, V.R. Pratt, M. Sokolsky, G. Stanek, D.M. Stavens, A. Teichman, M. Werling, S. Thrun, Towards fully autonomous driving: systems and algorithms, Intelligent Vehicles Symposium (2011) 163–168.

[15] M. Bojarski, D. Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, K. Zieba, End to End Learning for Self-Driving Cars, arXiv preprint, arXiv:abs/1604.07316 2016.

[16] Y. Luona, L. Xiaodan, W. Tairui, X. Eric, Real-to-virtual domain unification for end-to-end autonomous driving, Proc. of European Conference on Computer Vision (2018) 553–570.

[17] K. Mori, H. Fukui, T. Murase, T. Hirakawa, T. Yamashita, H. Fujiyoshi, Visual explanation by attention branch network for end-to-end learning-based self-driving, Proc. of IEEE Intelligent Vehicles Symposium (9-12 June 2019) https://doi.org/10.1109/IVS.2019.8813900.

[18] Q. Li, L. Chen, M. Li, S.L. Shaw, A. Nüchter, A sensor-fusion drivable-region and lane-detection system for autonomous vehicle navigation in challenging road scenarios, IEEE Transactions on Vehicular Technology 63 (2) (2013) 540–555.

[19] U. Lee, J. Jung, S. Jung, D.H. Shim, Development of a self-driving car that can handle the adverse weather, Int. J. Automot. Technol. 19 (1) (2018) 191–197.

[20] H. Xu, Y. Gao, F. Yu, T. Darrell, End-to-end learning of driving models from large-scale video datasets, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (21-26 July 2017) https://doi.org/10.1109/CVPR.2017.376.

[21] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, Computer Vision and Pattern Recognition 2016, pp. 2921–2929.

[22] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-CAM: visual explanations from deep networks via gradient-based localization, International Conference on Computer Vision 2017, pp. 618–626.

[23] Mariusz Bojarski, Anna Choromanska, Krzysztof Choromanski, Bernhard Firner, Larry Jackel, Urs Muller, Karol Zieba, VisualBackProp: Visualizing CNNs for Autonomous Driving, arXiv preprint, arXiv:abs/1611.05418 2016.

[24] J. Kim, J. Canny, Interpretable learning for self-driving cars by visualizing causal attention, International Conference on Computer Vision 2017, pp. 2942–2950.

[25] J. Kim, A. Rohrbach, T. Darrell, J. Canny, Z. Akata, Textual explanations for self-driving vehicles, European Conference on Computer Vision 2018, pp. 563–578.

[26] Y. Mori, H. Fukui, T. Hirakawa, N. Jo, T. Yamashita, H. Fujiyoshi, Attention neural baby talk: captioning of risk factors while driving, IEEE International Conference on Intelligent Transportation Systems, 2019.