https://doi.org/10.2352/ISSN.2470-1173.2021.17.AVM-115

Data Collection Through Translation Network Based on End-to-End Deep Learning for Autonomous Driving

Zelin Zhang
Department of Modern Mechanical Engineering
Waseda University
Tokyo, Japan

Jun Ohya
Department of Modern Mechanical Engineering
Waseda University
Tokyo, Japan

Abstract

To avoid manual collections of a huge amount of labeled image data needed for training autonomous driving models, this paper proposes a novel automatic method for collecting image data with annotation for autonomous driving through a translation network that can transform the simulation CG images to real-world images. The translation network is designed in an end-to-end structure that contains two encoder-decoder networks. The forepart of the translation network is designed to represent the structure of the original simulation CG image with a semantic segmentation. Then the rear part of the network translates the segmentation to a real-world image by applying cGAN. After the training, the translation network can learn a mapping from simulation CG pixels to the real-world image pixels. To confirm the validity of the proposed system, we conducted three experiments under different learning policies by evaluating the MSE of the steering angle and vehicle speed. The first experiment demonstrates that the L1+cGAN performs best above all loss functions in the translation network. As a result of the second experiment conducted under different learning policies, it turns out that the ResNet architecture works best. The third experiment demonstrates that the model trained with the real-world images generated by the translation network can still work great in the real world. All the experimental results demonstrate the validity of our proposed method.

Introduction

During the past few years, autonomous self-driving cars have become more and more popular because of the development of sensor equipment and computer vision technology. Many research groups and car manufactures have joined this industry, such as Google [1] and General Motors [2]. The purpose of autonomous driving is to let the vehicle perceive the surrounding environment and cruise with no human intervention. Therefore, the most important task for the autonomous driving system is to map the surrounding environment to the driving control. Recently, deep convolutional networks have achieved great success in traditional computer vision tasks such as segmentation [3] and object detection [4]. Therefore, the autonomous driving systems using deep learning have become more and more popular in this field. With the trained model, autonomous vehicles can deal with more scenarios over the past.

Some state-of-the-art works divide the autonomous driving problem into several small tasks and fuse the results of each task to a final control decision. The rest of the state-of-the-art works provide an End-to-End solution that allows the autonomous system to learn the mapping from the raw image data to the steering control. Therefore, very many labeled data are required by the training procedure. However, the data collection and annotation for the system training could cost too much time and human labor. Besides, real damage could happen during the data collection under human manual driving in the real world. Some of the state-of-the-art works have published their datasets, but the datasets can be only applied in the research under similar circumstances. Hence, autonomous driving systems that are trained by these datasets can only be utilized in similar scenarios.

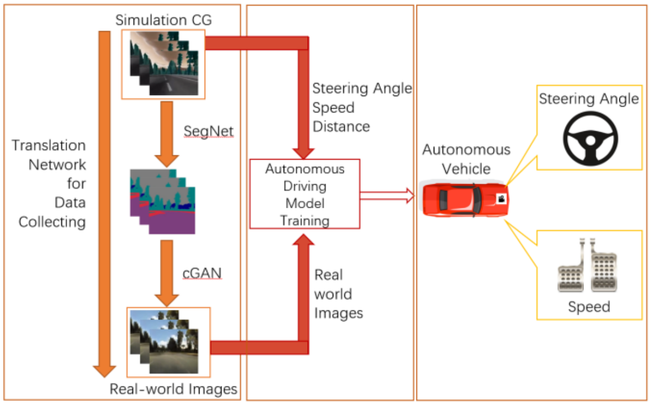

Fig.1. Overview of the system we proposed in this paper. The left part is the translation network for autonomous driving data collection. The middle part is the autonomous driving model training. The final output is shown in the right part that is the steering angle and the speed for the autonomous vehicle.

Moreover, the learning policy plays an important role in the deep learning-based system. Since the most important task for the autonomous driving system is to monitor the surrounding environment and make the corresponding autonomous vehicle movement control decision, different learning policies lead to different trained models which could influence the autonomous driving system massively. Supervised learning requires a large amount of training data with the corresponding annotation. On the other hand, reinforcement learning requires a whole environment for the vehicle to learn how to drive automatically from mistakes and trials. Hence, finding a way to collect the data with the corresponding information is quite important for the deep learning-based autonomous driving system. The dataset could have a strong influence on the final movement control.

In this paper, we propose a novel way for data collection through the translation network. With the translation network, we can generate the dataset under a simulator without hard labor and any risks. More specifically, as shown in Fig. 1, the proposed translation network converts CG images to real images. To demonstrate the proposed method, we test different autonomous driving movement control systems that are trained under different learning policies with the dataset acquired from the translation network. The method we propose has two main contributions as follows:

(1) We propose a translation network for data collection. The translation network can transform a simulation CG image into a real-world image. The translation network provides a novel way to collect data for the autonomous driving training procedure.
(2) We generate a dataset with a translation network. The dataset includes the speed, the steering angle, and the distance information that is used to describe the relative position in the surrounding environment.

The rest of this paper is organized as follows. Section 2 gives an overview of the state-of-the-art related work. The translation network we propose is explained in Section 3. In Section 4, we give the experimental details for data collation through the translation network and the evaluation of the experiments and the corresponding analysis. The conclusion of our work is given in Section 5.

Related Work

With the development of deep learning, research groups and companies have started to attempt a deep learning-based method to solve the autonomous driving problem. We analyze the state-of-the-artwork in the past few years and simply categorize them into two different learning policies: supervised learning and reinforcement learning.

Supervised Learning

Supervised learning policy allows autonomous vehicle learning from the experience in the past.

Rule-based methods divide the autonomous driving problem into several small tasks, such as interaction with cars, lane following [5], pedestrian detection [6], and traffic light recognition [7]. Rule-based methods tend to solve all the small tasks independently and fuse all the results obtained by each task to achieve the final movement control. Although the rule-based methods have achieved great success, each result gained from the sensors could influence the final controls significantly [8]. Even if one of the sensors is unfunctional, it may cause a significant problem. Although the rule-based system sounds reasonable, it is still used for a driver assistant system rather than an autonomous driving system.

Instead of dividing the large task into several small ones, the perception-based method simply learns the mapping from the images to the steering controls. ALVINN et al. [9] proposed an idea first: they used a neural network to make the first attempt. Although the network is very simple and shallow, it can still be used in several certain situations. With the development of convolutional neural networks in recent years, some traditional hardware companies have also joined this field. Recently, Nvidia [10] collected the training datasets with three cameras from the left, right, and center, and trained a deep convolutional neural network to map the pixels to the steering controls. However, all the methods mentioned above aim at processing data properly to achieve better performance.

Reinforcement Learning

In recent years, deep reinforcement learning has drawn many people's attention and has been applied to many felids such as robot automatic system [11], and computer game agents [12]. The goal of deep reinforcement learning is to make agents interact with the surrounding environment. In 2014, Koutník et al. developed the TORCS driving simulator which is applied in a convolutional neural network with 8 CPU cores in parallel. In 2015, Nair et al. proposed an idea of general reinforcement learning architecture. Their system allows asynchronous training of reinforcement learning agents in a distributed setting. Also in 2015, deep Q-learning [13] and policy gradient [14] have been proposed, which drew attention for reinforcement learning. Under deep Q-learning and policy gradient, the agent has to interact with the surrounding environment to learn the movement through the mistakes and trials. However, training an autonomous vehicle under reinforcement learning in the real world could be dangerous, because the interaction could be erroneous and cause damage to the real world.

Translation Network

Although most of the existing datasets have their unique advantages, there are no datasets designed particularly for the autonomous vehicle movement control. Therefore, we try to establish a dataset for the autonomous vehicle movement control.

Existing Datasets

The recent state-of-the-art works provide a solution for autonomous driving through supervised learning. With supervised learning, a large scale of data is required to achieve a robust result for autonomous driving. Although several datasets have been published, each of them is collected for its individual goal. Here, we introduce some widely used datasets in this field.

KITTI [15]. KITTI was collected by [15]. It contains 7481 training images. Each image is annotated with the 3D bounding box. The ground truth is measured by a laser scanner and a GPS. The data is collected in both rural areas and highways and reveals up to fifteen vehicles and thirty people. However, the task of interest of the KITTI dataset is 3D object detection, 3D tracking, and visual odometry. Therefore, it is not designed for the autonomous vehicle movement control domain.

Cityscapes [16]. The cityscapes dataset is published by [16]. It contains a sequence of images that are collected in the streets of fifty different cities. By far, it is the dataset with the most variation. However, the cityscapes dataset is designed for scene understanding. It mainly focuses on semantic segmentation. The annotation of each image frame contains dense semantic segmentation results up to thirty classes. Even though it has great diversity, it is still not suitable for training the autonomous vehicle movement control system.

Oxford RobotCar Dataset [17]. The Oxford RobotCar dataset is collected through the central Oxford street by using the Oxford RobotCar platform. The dataset contains 1000km of manual driving data. It is captured by six cameras and has nearly 20 million images. The data is collected in all the weather conditions, including heavy rain, sunlight, and snow. However, the annotation of the movement control is missing, because the Oxford RobotCar dataset is collected for the task of visual odometry.

Overview of the Translation Network

In this section, we establish a dataset for autonomous driving with the indicators of steering angle, speed, and distance data. However, the data collection through manual driving in the real world could cost too much time and human labor. It could save a lot of work if the simulation CG image data can be transformed into real-world image data because the simulation data can be easily obtained by the simulator with the corresponding annotation. We propose a translation network for data collection in this paper. In computer vision, many transformation problems can be defined as translating one input image into a corresponding output image. The goal of image transformation is to map each pixel of the input image to the output image. With enough number of image pairs in a dataset, the problem can be solved by the training procedure. The data collection can be done in the following three steps through the translation network:
1) Data preparation for training translation network.
2) Translation network architecture.
3) Data generation through translation network.

Data Preparation

The translation network learns a mapping from the simulation CG pixels to the real-world image pixels. Therefore, it is important to collect the simulation CG image data and real-world images data for the network training.

The simulation CG images are collected under an open-source game simulator TORCS (The Open Racing Car Simulator) [18]. We collected all the simulation image data through manual driving. All the images are captured as the driver's perspective. All the tracks used in the simulation data collection are one-way street with three lanes without intersection. As a total, we collected 121,624 images during manual driving. However, most simulation images collected in TORCS have similar appearances, because the surrounding environment would not change dramatically in a short time. To avoid the data overlapping, we use the subset of 25000 images which have totally different appearances for the training.

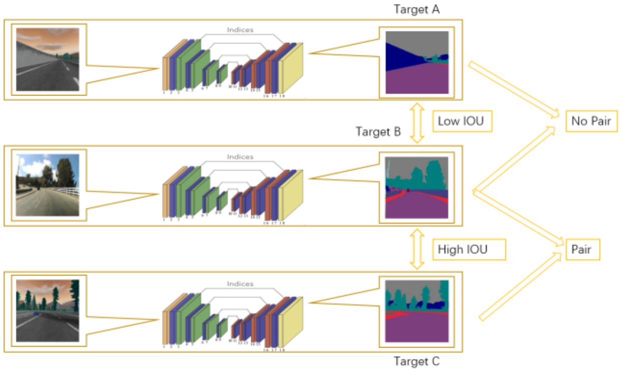

Fig.2. Overview of the data preparation. The simulation CG images and real-world images are segmented by the SegNet. Two images with high IOU are considered to have a similar structure and set as an image pair.

The real-world images come from Pan et al. [19] which was published in 2016. All the image data are collected in sunny daytime so that we do not encounter illumination problems. Although there are nearly 45k images in this dataset, we only select 25k images as the translation network training data to avoid the data overlapping.

After obtaining the simulation CG image data and real-world image data, we set them into pairs for translation network training in order to let the translation network learn a mapping from simulation CG image pixels to real-world image pixels. An image pair of a simulation CG image and real-world image should share a similar structure. This paper uses the semantic segmentation results for the structure. Inspired by the SegNet [20], we utilize the semantic segmentation results to evaluate the similarity between a simulation CG image and real-world image structures. Based on this idea, we first output the semantic segmentation of each simulation CG image and real-world image. Then, we apply the Intersection over Union (IoU) to evaluate the similarity between the semantic segmentation results of the image pair using IoU. Here, IoU is the area of the overlap between the semantic segmentation of the simulation CG images and the semantic segmentation of the real-world images. As defined in Eq. (1), it is decided by the area of the union between the two segmentation results.

$$IoU = \frac{\text{targetA} \wedge \text{targetB}}{\text{targetA} \cup \text{targetB}} \qquad (1)$$

In this paper, the segmentation target A is the simulation CG images collected from the TORCS. The segmentation target B is the real-world images from the dataset [19]. The structure of both target images is represented by the semantic segmentation. Eventually, the simulation CG image and the real-world image will set into a pair with the highest IOU above all. Fig.2 shows the image pair setting procedure. In total, we collect 25k image pairs for the training procedure. With the image pairs, we now design the translation network architecture that can learn the mapping from simulation CG image pixels to real-world image pixels.

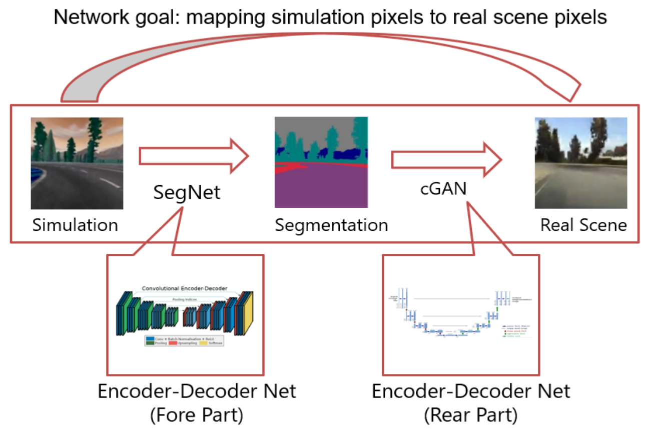

Fig.3. The overview of the image translation network. It is an encoder-decoder network that combines the SegNet and a cGAN.

Translation Network Architecture

Inspired by Badrinarayanan et al. [20], we design an encoder-decoder network to transform simulation CG images to real-world images. As shown in Fig.3, the whole translation network aims at mapping the simulation CG image pixels to real-world image pixels. The translation network is an encoder-decoder network which can be divided into two parts. The fore part extracts the structure of the simulation CG image and outputs the structure as a semantic segmentation result. Inspired by [20], we apply the SegNet structure as the forepart of the transformation network.

The rear part of the translation network can be described as an Image-to-Image translation network. The rear part transforms the semantic segmentation result obtained from the forepart into a real-world image. Inspired by Pan et al. [11], we apply the generator and discriminator to establish the network architectures of the rear part. We implement the convolutional layer and a BatchNorm layer in a cascade manner and put the sequence of the BatchNorm layer and a ReLu layer into a block. This block is used in both generator and discriminator. Besides, we apply a U-Net [13] to link the two separate layers from the encoder and decoder with a simple skip. Previous works usually pass the input data directly to the bottleneck layer. However, this direct architecture could be considered to be time-consuming, because the information has to pass through all the layers to the bottleneck. Besides, the low-level information may be lost during passing through all the layers, whereas low-level information is important and indispensable for high-level information reconstruction. Hence, we apply a skip connection in the network to directly link the two separate layers from both the encoder and decoder sides. We combine the forepart and the rear part to one encoder-decoder network.

Data Generation Through The Translation Network

Now, the translation network is designed as an encoder-decoder architecture. The encoder-decoder consists of the SegNet, and the cGANs [14]. We apply the U-Net structure to cGANs and connect two separate layers from the encoder and decoder with a simple skip. We also place the LeakyReLu layer after each convolutional layer and set the slop as 0.2. Also, ReLu layers are placed after each deconvolutional layer. Adam optimizer is applied in our training. The initial learning rate is set as 0.002 with a momentum of 0.5. After the training procedure, we collect the dataset for the autonomous driving system. We collect all the sequential image data through the simulator TORCS and then transform these simulation CG image data to real-world image data. In total, we use 10 tracks in TORCS and collect nearly 90k images which include nearly 300 times manually driving. The data were collected on two separate weather conditions, where one day was sunny, and the other was overcast. Some examples of image transformation results are shown in Fig.4.

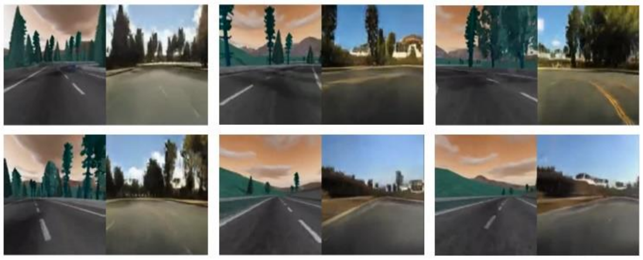

Fig.4 Examples of image transformation results. The left side is the simulation images from TORCS, and the right side is the real-world images generated by the translation network

Experimental Results and Discussion

In this paper, we conducted the following three experiments to evaluate our work.

(1) We conduct experiments for evaluating our translation network with four different loss functions: L1, GAN, cGan, L1+cGAN.

(2) We conduct experiments to testify the authenticity of the dataset generated by the translation network by applying four different learning policies which are DQN, A3C, AlexNet, ResNet.

(3) We conduct experiments to evaluate the model trained under the dataset generated by the translation network by testing it on the real-world driving scenario.

The experiments for autonomous driving use the mean absolute error defined by Eq.(3) for the steering angle and speed for evaluation.

$$MSE = \frac{1}{n}\sum_{k=1}^{n}|p_i - g_i|. \qquad (3)$$

where p stands for the prediction and the g stands for the ground truth. In these experiments, the maximal speed of the autonomous vehicle is 70km/h. The steering angle of the autonomous vehicle ranges [-π/2, π/2]. The range in [-π/2, 0] is defined as to go left, and [0, π/2] is defined as go right. The angle is in degree, and the speed is in km/h.

In the experiments, we try to minimize the difference between the output and the ground truth. However, we do not need the output to be exactly the same as the ground truth. In other words, the system can still perform well if there is only a small error between the prediction and the ground truth. Therefore, we set this small error as a threshold ts to evaluate driving motion. Predictions whose MSE are below the ts should be considered as correct driving motions; otherwise, wrong driving motions. We set the parameter cdm to represent the correct driving motion and wdm to represent the wrong driving motion. Then, we define a system performance score as PeSc = cdm / (cdm + wdm). According to [21], we set the threshold of the steering angle as 10 degrees, and the 8 km/h for the speed.

Evaluation of translation network

With the translation network, we generate a new image dataset for autonomous driving that includes the speed, steering angle, and the distance information with other vehicles. Compared with the existing dataset, our dataset includes the driver's manual driving decisions that can describe the interaction with the surrounding environment. Table 1 shows a comparison with the existing dataset.

Table 1. Comparison of the dataset generated by the translation network and other existing datasets.

Dataset	Setting	Type	Diversity	Move-ments
KITTI	City, highway	Real	Road	NO
City-scape	City	Real	Weather	NO
Oxford	City	Real	One city	NO
Ours	City, highway	synthesis	Road	Yes

In order to minimize the difference between the simulation CG image and generated real-world images, we apply four loss-functions to find the best combination for the translation network. We iterate each network for 10k times to make the network convergence. IoU is applied to evaluate each loss function. The results are shown in Table 2.

Table 2. IoU performance of different loss functions for the translation network.

Loss	IoU
L1	0.56
GAN	0.48
cGAN	0.75
L1+cGAN	0.83

Apparently, L1+cGAN performs the best above all four loss-functions. Therefore, we apply the L1+cGAN as the loss function for the translation network. Fig.5 shows an example of image translation under different loss functions. Compared to the original simulation CG images and its segmentation result, network with L1 loss is quite blurred. Results with GAN reasonable but still blurred. cGAN turned out an acceptable result, but still not as good as cGAN + L1.

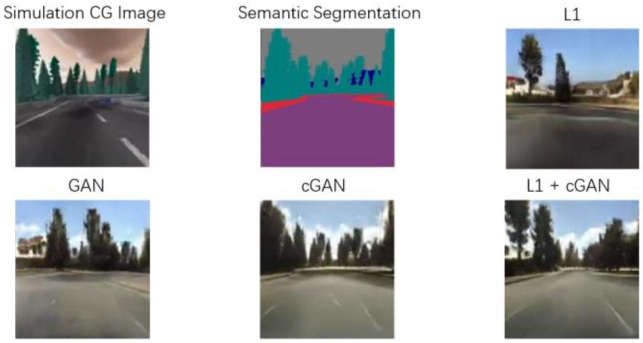

Fig.5. The image translation results with different loss functions. Compared with the original simulation CG image, the L1 + cGAN performs best among all.

Evaluation of Different Learning Policy

After obtaining the dataset from the translation network, we conduct experiments with four different learning policies for autonomous driving: DQN, A3C, AlexNet, and ResNet. Table 3 compares the results, where MSE and system performance score PeSc for the angle and speed for the four architectures are listed.

Table 3. Performance of different learning policies.

Learning Policy	Item	MSE	PeSc
DQN	Angle	15.13	63.5%
	Speed	12.40	66.9%
A3C	Angle	13.71	70.1%
	Speed	11.28	72.3%
AlexNet	Angle	11.56	78.8%
	Speed	10.37	76.4%
ResNet	Angle	**9.68**	**81.1%**
	Speed	**8.53**	**78.7%**

Obviously, supervised learning with ResNet architecture performs the best among all the four learning policies which demonstrate that supervised learning would perform better with applicable annotation. However, consider that the system tolerance, we think that the dataset acquired from the translation network can be applied for the autonomous driving model training procedure.

Evaluation of Real-world Diving Scenario

The above-mentioned experiments prove that supervised learning with ResNet is the best learning policy for autonomous driving model training. Now we train a model under ResNet and test the trained model on the existing dataset. We apply the trained model to KITTI dataset to demonstrate that the dataset acquired from the translation network can achieve great performance in a real-world driving scenario. We apply two different training set to generate the model and test the two models on the same KITTI test set. Table 4 shows the results of both datasets.

Table 4. Performance of different datasets.

Training Set	Item	MSE	PeSc
KITTI	Angle	8.77	83.9%
	Speed	9.31	75.2%
Ours	Angle	9.68	81.1%
	Speed	8.53	78.7%

Table 4 shows that the model trained on KITTI dataset gives better performance on the angle. However, the dataset acquired from the translation network also achieves better results on speed. Considering the acceptable system tolerance, we think the dataset acquired from the translation network can achieve a great performance in the real-world environment.

Conclusion

In this paper, we have proposed a novel way to collect data for autonomous driving using the translation network. The translation network can learn a mapping from the simulation CG pixels to the real-world image pixels. Through the translation network, we generate a dataset with the corresponding annotation for autonomous driving. To confirm the validity of the proposed system, we conducted three experiments for evaluating the MSE and accuracy of the steering angle and speed. The first experiment demonstrates that the L1+cGAN performs best among all the loss functions for the translation network. The second experiment demonstrates the real-world images generates by the translation network can work under different learning policies, and supervised learning with ResNet performs best by far. The third experiment demonstrates that the autonomous driving model trained by real-world images can work great in a real-world driving scenario. In the future, we may expand the working scenarios. Meanwhile, we would like to improve the network learning ability and accuracy and try to make the system work more stable.

References

[1] Gibbs, Samuel. "Google sibling waymo launches fully autonomous ride-hailing service." The Guardian 7 (2017).

[2] Wehner, Michael, et al. "An integrated design and fabrication strategy for entirely soft, autonomous robots." Nature 536.7617 (2016): 451-455.

[3] Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

[4] Lin, Tsung-Yi, et al. "Feature pyramid networks for object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[5] Kawazoe, Hiroshi, On Sadano, and Masayasu Shimakage. "Lane following vehicle control." U.S. Patent No. 6,542,800. 1 Apr. 2003.

[6] Zhang, Liliang, et al. "Is faster r-cnn doing well for pedestrian detection?." European conference on computer vision. Springer, Cham, 2016.

[7] Li, Xi, et al. "Traffic light recognition for complex scene with fusion detections." IEEE Transactions on Intelligent Transportation Systems 19.1 (2017): 199-208.

[8] Cho, Hyunggi, et al. "A multi-sensor fusion system for moving object detection and tracking in urban driving environments." *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014.

[9] Pomerleau, Dean A. "Alvinn: An autonomous land vehicle in a neural network." Advances in neural information processing systems. 1989.

[10] Bojarski, Mariusz, et al. "End to end learning for self- driving cars." arXiv preprint arXiv:1604.07316 (2016).

[11] Mosavi, Amir, and Annamaria R. Varkonyi-Koczy. "Integration of machine learning and optimization for robot learning." Recent Global Research and Education: Technological Challenges. Springer, Cham, 2017. 349-355.

[12] Vinyals, Oriol, et al. "Starcraft ii: A new challenge for reinforcement learning." arXiv preprint arXiv:1708.04782 (2017).

[13] Gu, Shixiang, et al. "Continuous deep q-learning with model-based acceleration." International Conference on Machine Learning. 2016.

[14] Yu, Lantao, et al. "Seqgan: Sequence generative adversarial nets with policy gradient." Thirty-First AAAI Conference on Artificial Intelligence. 2017.

[15] Geiger, Andreas, et al. "Vision meets robotics: The kitti dataset." *The International Journal of Robotics Research* 32.11 (2013): 1231-1237.

[16] Cordts, Marius, et al. "The cityscapes dataset for semantic urban scene understanding." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[17] Maddern, Will, et al. "1 year, 1000 km: The Oxford RobotCar dataset." *The International Journal of Robotics Research* 36.1 (2017): 3-15.

[18] Salem, Mohammed, et al. "Driving in TORCS using modular fuzzy controllers." *European Conference on the Applications of Evolutionary Computation*. Springer, Cham, 2017.

[19] Pan, Xinlei, et al. "Virtual to real reinforcement learning for autonomous driving." arXiv preprint arXiv:1704.03952 (2017).

[20] Badrinarayanan, Vijay, Alex Kendall, and Roberto Cipolla. "Segnet: A deep convolutional encoder-decoder architecture for image segmentation." IEEE transactions on pattern analysis and machine intelligence 39.12 (2017): 2481-2495.

[21] Wei, Junqing, et al. "Towards a viable autonomous driving research platform." 2013 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2013.

Author Biography

Zelin Zhang received his BS in Engineering from the Wuhan University of Science and Technology and his PhD in applied Engineering from Waseda University. His work has focused on the development of an autonomous driving system.