Cooperative AI for Human–Machine Work

Authors: *Saori Matsunaga*, Toshisada Mariyama*, Takuji Morimoto**, Yoshihiro Mitsuka** and *Takumi Sato**

1. Introduction

In the "new normal" society with the need to save labor and increase the separation among workers, humans and machines may more often coexist. Currently, the workspace of humans is usually separated from that of machines to ensure safety. To improve efficiency, however, humans and machines will need to work together in the future. Accordingly, Mitsubishi Electric Corporation has developed, using its AI technology Maisart, cooperative AI for human–machine work that makes it easier for machines to work together with humans by imitating the natural behavior of humans. The technology uses inverse reinforcement learning to attain efficient learning with small data sets. This paper introduces the cooperative AI for human–machine work using the example of applying the technology to a small autonomous mobile system.

2. Outline of the Technology

2.1 Overview

Figure 1 illustrates an overview of the proposed technology. The manipulation data of cooperative actions performed by a human is imitated through inverse reinforcement learning (2.2), which realizes natural cooperative actions. Because the process of trial and error needs to be repeated in this approach, a simulation environment that mimics the actual environment needs to be prepared. The sensor data output from the simulator and actual machine needs to be appropriately processed in advance to convert it to an easy-to-learn format. In addition, to control actual machines, a control cycle and method that are suitable for each machine need to be adopted. Generally, autonomous mobile systems require movement control in a cycle of several milliseconds. However, when determining routes approximately, control in cycles of 100 to several hundred milliseconds is sufficient. For these reasons, functions are divided into several modules: a top-down view generator and feature extraction network that pre-process the input data; a learner and trained model that determine approximate target actions; and a control module that performs fine control. This makes it possible to perform both high-speed smooth control and make difficult judgments in actual environments. Each module is explained below.

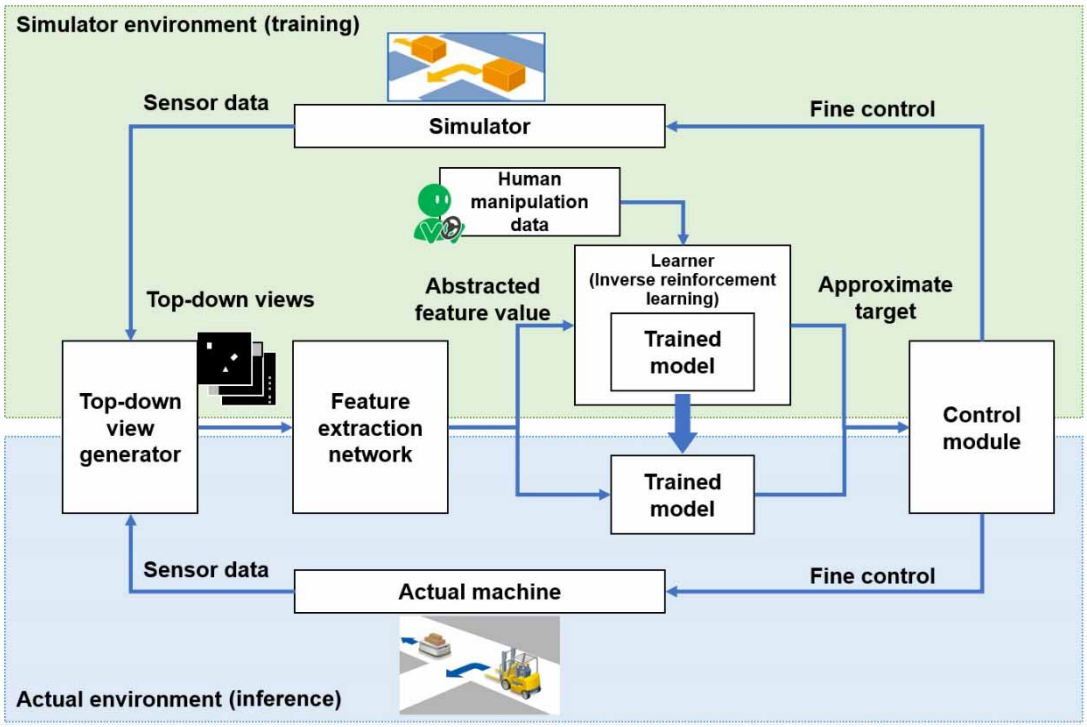


Fig. 1 Overview of the proposed framework

*Information Technology R&D Center **Automotive Electronics Development Center

2.2 Learner (inverse reinforcement learning)

When multiple persons (or machines that humans operate) work in the same environment, operators see the movements of other operators or machines and adjust the speed and sequence of their own operations, ensuring the safety of all operators as well as efficiency. Therefore, in an environment where humans coexist with automated guided vehicles (AGVs) and other machines, the machines should give way to humans or move slowly. Currently, however, because machines move based on predetermined rules such as "advance" and "stop when detecting an obstacle," the operation efficiency of both sides may decrease, for example, when both the machine and operator cannot move. One possible solution would be to give desired action rules for possible events, but it is difficult to list up all rules. Accordingly, we used AI to make machines cooperate with humans. One approach using AI is reinforcement learning for which it is necessary to design a function called a reward function that indicates whether the response to a status is good or bad. However, in autonomous mobile system control in which the surrounding environment changes in a complex manner, it is difficult to design such reward functions. Another technique is called imitation learning, in which behavior is learned such that it becomes similar to a sample action and for which no reward function is designed. There are multiple methods of imitation learning. One method, which uses sample data (demonstration data) to learn behavior in supervised learning, requires a large volume of demonstration data. This is because for statuses that are not included in the training data, appropriate behavior cannot be determined and so errors that may occur during the control need to also be considered and data sets that include these errors need to be prepared. Another method is inverse reinforcement learning. In this method, based on demonstration data, machines estimate reward functions by repeating trial and error through simulation and then use the estimated reward functions to perform reinforcement learning. Generative adversarial imitation learning (GAIL)[1], which is one type of inverse reinforcement learning, learns the optimum behavior in accordance with the procedure of generative adversarial networks (GANs).[2] It has been reported that this method requires fewer demonstration data sets for learning than supervised learning. For this reason, our technology uses GAIL.

2.3 Top-down view generator and feature extraction network

Mobile system control using AI often involves image input. One reason is that images can easily show the positional relationships between multiple objects in an environment where the statuses of a target car and surrounding objects constantly change. However, the obtained images usually contain information that is unrelated to deciding the behavior of the target car. In addition, if a simulator is used for learning, differences between images used in the learning and actual images in actual control cause problems. Therefore, a top-down view generator is used to convert information obtained by the simulator and actual machine into virtual images. This reduces differences between the actual environment and simulation environment while cutting unnecessary information. At present, top-down views are used to simply express the positional relationships between the target car and surrounding objects. Although information on the surroundings can be obtained by top-down views, the number of dimensions of image data is large, which may adversely affect the speed and stability of learning. To solve this problem, a feature extraction network is introduced to compress the obtained top-down views to extract lower-dimensional feature values. In our technology, a variational autoencoder (VAE) is used to convert views into lower-dimensional vectors. By creating a large quantity of various types of artificial top-down views, feature extraction network learning can be completed before inverse reinforcement learning. The final input data is a combination of the feature values extracted from top-down views with information on the speed and other factors that is not included in the top-down views.

3. Application to a Small Mobile System

3.1 Scenarios

We applied this technology to a small autonomous mobile system imitating an AGV in an experiment; the results are shown below. Figure 2 shows the experimental scenario. The target AGV travels in a straight line to the right along the thick solid line in the figure. It aims to reach the right end as fast as possible without colliding with or hindering the forklift. The forklift retreats so as to cross the travel route of the AGV, changes direction, and then travels to the right as shown by the broken lines in the figure. Therefore, if the AGV travels according to the rules of "advance" and "stop when detecting an obstacle," it will collide with the forklift or both the AGV and forklift will stop in front of each other depending on the timing when the forklift retreats (Fig. 3). In this experiment, two patterns were provided as the timing when the forklift edges into the travel route of the AGV: (a) a sudden interrupt that forces the AGV to retreat and (b) a slow interrupt that does not require the AGV to retreat. The two scenarios were mixed for learning and evaluation. The time at which the AGV and forklift started and their start positions were slightly changed every time.

3.2 Learning using a simulator

We used a simulator that we had developed in the

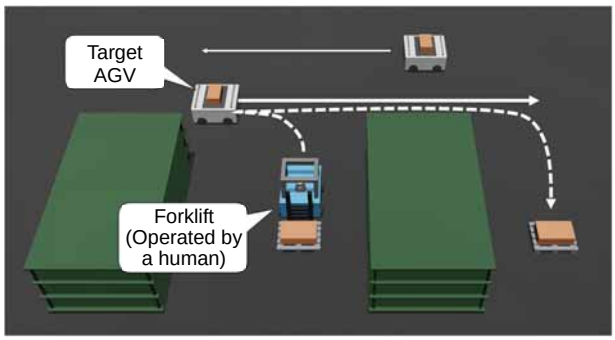


Fig. 2 Experimental scenario

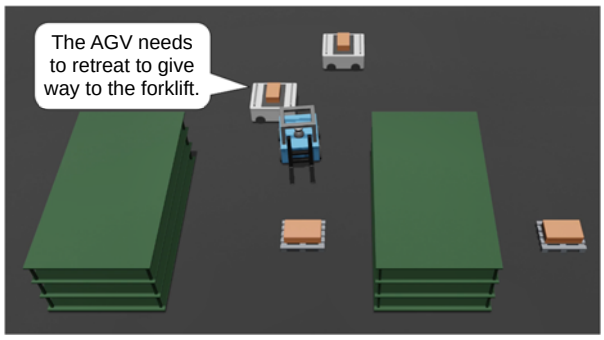
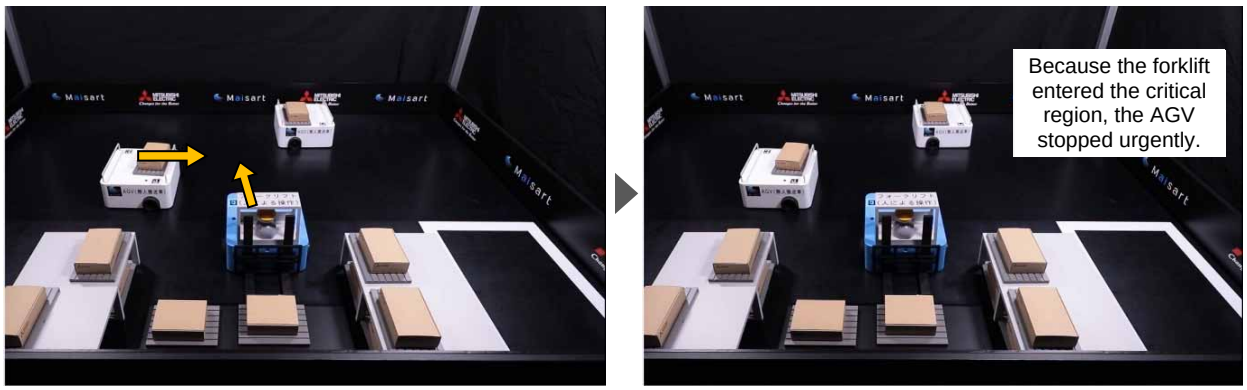
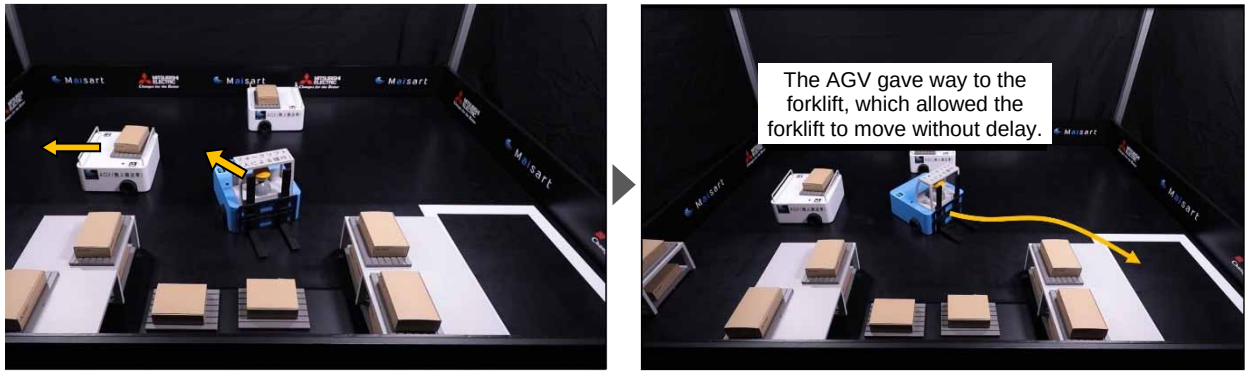


Fig. 3 An example of a scene that requires cooperative motion



(a) The cooperative AI for human–machine work was not applied.



(b) The cooperative AI for human–machine work was applied.

Fig. 4 Real-world experiments with cooperative AI

learning of cooperative actions for the scenarios described in the previous section. The target of a top-down view is a square area of side 1 m with the target car at the center; binary images with only another vehicle included were used. The feature extraction network was used to convert a top-down view to 16-dimensional vectors and the speed of the target car was added to them. Therefore, the input data has a total of 17-dimensional vectors. Demonstration data was collected when a human operated the AGV on the simulator for each of scenarios (a) and (b). Inverse reinforcement learning and supervised learning were performed and the scores were calculated based on whether the AGV collided with the forklift and the time required to complete the operation and compared. As a result, although limited to the scenarios used this time, when four or more demonstration data sets were available for each of scenarios (a) and (b) in the inverse reinforcement learning, the obtained scores were the same or higher than those of humans. On the other hand, in the supervised learning, even when the number of demonstration data sets used was ten or more times those used in the inverse reinforcement learning, the scores of the supervised learning were lower than those obtained in the inverse reinforcement learning and they also widely varied. These results confirm that our proposed technology is superior to supervised learning for the number of required data sets, as well as safety, efficiency, and stability of operations.

3.3 Experiment using actual machines

Lastly, the results of an experiment using actual machines are shown below. As shown in Fig. 4, in the experiment, the AGV, forklift, and surrounding objects used are the same as those in the simulation environment shown in Fig. 2. A model trained with 20 demonstration data sets was applied to the target AGV without additional learning and adjustment. When the AGV moved according to the rules of “advance” and “stop when detecting an obstacle,” at the moment when the forklift entered its critical region, the AGV stopped urgently, which resulted in operation time loss (Fig. 4(a)). On the other hand, when our technology was applied, the AGV retreated to give way to the forklift, which allowed the forklift to travel without delay. These results show that our technology contributes to realizing smooth operations (Fig. 4(b)).

4. Conclusion

When the cooperative AI for human–machine work was used, natural behavior was obtained with fewer demonstration data sets thanks to the inverse reinforcement learning. In addition, combining the cooperative AI with a top-down view generator and feature extraction network realized cooperative actions of the actual machine. We will keep working on development toward applying the technology to actual production and distribution sites where humans and machines may coexist and to autonomous driving.

References

(1) Ho, J., et al.: Generative Adversarial Imitation Learning, Advances in Neural Information Processing Systems, 4565–4573 (2016)
(2) Goodfellow, I., et al.: Generative Adversarial Nets, Advances in Neural Information Processing Systems, 2672–2680 (2014)