Mitigate Catastrophic Forgetting by Varying Goals

Lu Chen[1] and Murata Masayuki[2]
[1]*Kyoto Institute of Technology, Kyoto, Japan*
[2]*Osaka University, Osaka, Japan*

Keywords: Modular Network, Catastrophic Forgetting, Neural Network.

Abstract: Catastrophic forgetting occurs because neural network learning algorithms change connections to learn a new skill which encodes previously acquired skills. Recent research suggests that encouraging modularity in neural networks may overcome catastrophic forgetting because it should reduce learning interference. However, manually constructing modular topology is hard in practice since it involves expert design and trial and error. Therefore, an automatic approach is needed. Kashtan et al. find that evolution under an environment that changes in a modular fashion can lead to the spontaneous evolution of modular network structure. However, goals in their research are made of a different combination of subgoals, while real-world data is rarely perfectly separable. Therefore, in this paper, we explore the application of such approach to mitigate catastrophic forgetting in a slightly practical situation, that is applying it to classification of small sized real images and applying it to the increment of goals. We find that varying goals can improve catastrophic forgetting in a CIFAR-10 based classification problem. We find that when learning a large set of goals, a relatively small switching interval is required to have the advantage of mitigating catastrophic forgetting. On the other hand, when learning a small set of goals, an appropriate large switching interval is preferred since this less worsens the advantage and also can improve accuracy.

1 INTRODUCTION

Learning a variety of different skills for different problems is a long-standing goal in artificial intelligence (Ellefsen et al., 2015). However, in neural networks, when it learns a new skill, it typically losing previously acquired skills (Ellefsen et al., 2015). This problem called catastrophic forgetting, and it occurs because learning algorithms change connections to learn a new skill which encode previously acquired skills (Ellefsen et al., 2015).

Catastrophic forgetting has been studied for a few decades (French, 1999). Recently, in computational biology field, a modular approach for neural networks is considered to be needed as learning problems grow in scale and complexity (Amer and Maul, 2019). (Ellefsen et al., 2015) studied whether catastrophic forgetting can be reduced by evolving topological modular neural networks. They said that modularity intuitively should reduce learning interference by separating functionality into physically distinct modules. Their results suggest that encouraging modularity in neural networks may overcome catastrophic forgetting. However, in their approach, the input data is needed to be partitioned in advance so that different modules can be assigned. Since manual data modularization is usually based on some heuristic, expert knowledge or analytical solution, a good partitioning requires a good prior understanding of the problem and its constraints, which is rarely the case for neural network learning tasks (Amer and Maul, 2019).

Although there are several manual techniques for constructing modular topology, manual formation is hard in practice since manual techniques involve expert design and trial and error(Amer and Maul, 2019). Therefore, an automatic approach is needed. In the field of computational biology, (Kashtan and Alon, 2005) find that evolution under an environment that changes in a modular fashion leads to the spontaneous evolution of modular network structure. That is, they repeatedly switch between several goals, each made of a different combination of subgoals, which they call MVG (modularly varying goals). Although modular structures are usually less optimal than non-modular ones (Kashtan and Alon, 2005; Alon, 2003), they find that modular networks that evolve under such varying goals can remember their history (Parter et al., 2008).

Chen, L. and Masayuki, M.
Mitigate Catastrophic Forgetting by Varying Goals.
DOI: 10.5220/0008950005300537
In *Proceedings of the 12th International Conference on Agents and Artificial Intelligence (ICAART 2020) - Volume 2*, pages 530-537
ISBN: 978-989-758-395-7; ISSN: 2184-433X

Moreover, (Kashtan and Alon, 2005) showed that MVG also leads to spontaneous evolution of network motifs, bifan and diamond motif, which are kinds of four-node subgraphs which occur significantly often than that in random networks. Bifan and diamond motifs are relatively highly connected among the four-node motifs existing in a feedforward neural network. These motifs are interesting because they could be considered that contribute to the modularity since a network with high modularity is considered that networks having densely connected groups of vertices with only sparse connections between them (Newman, 2006). Also, bifan and diamond motifs are interesting because they could be thought as structural motifs able to provide large and diverse functional interactions. (Sporns and Kötter, 2004) suggest that biological neuronal networks have evolved such that their repertoire of potential functional interactions is both large and highly diverse, while their physical architecture is constructed from structural motifs that are less numerous and less diverse. A large functional repertoire facilitates flexible and dynamic processing, while a small structural repertoire promotes efficient encoding and assembly. Since bifan and diamond motifs have symmetric structures, those could be thought as structural motifs able to provide large and diverse functional interactions.

Our study explores the application of MVG to mitigate catastrophic forgetting in a slightly practical situation. In (Kashtan and Alon, 2005), they not only showed switching goals made of a different combination of subgoals can lead to spontaneous evolution of modular network structure, but also pointed out that randomly changing environments do not seem to be sufficient to produce modularity. However, real-world data is neither perfectly separable nor random. For example, it is popular that images have similar features, e.g. edge, intersecting lines, curves (Li et al., 2015). This is not the situation considered in (Kashtan and Alon, 2005). Although MVG uncovers the fundamental mechanism of the spontaneous evolution of modular network structure, there is still a long way to practical use. Therefore, to show the availability of MVG in a practical situation, in this paper we explore the application of MVG in a slightly practical situation, in detail, applying it to classification of small sized real images, such as CIFAR-10. Also, although 2 goals are evaluated in (Kashtan and Alon, 2005), we explore the increment of goals since a large amount of data would be expected to learn in practical use. To distinguish from MVG which is targeted to goals made of a different combination of subgoals, the approach in this paper is renamed as CFVG (mitigate Catastrophic Forgetting by Varying Goals).

In the evaluation, we compare CFVG with neural networks learned a single goal, which is the most common method in practical use. The reason why there is no comparison with existing catastrophic forgetting methods is that there is no CIFAR-10 sized image learn-able method with generating modules, which is considered able to learn problems grow in scale and complexity (Amer and Maul, 2019). In recent research, (Kirkpatrick et al., 2017) has proposed a practical solution to overcome catastrophic forgetting to train a neural network by protecting the weights important for previous goals. However, exact recognition (French, 1999) is required which could be inferred having a limitation in learning goals. In fact, there is a parameter that exists, which sets how important the old goal is compared with the new one. Instead of it, in this paper, the amount of time it required to relearn the original goal is measured, which does not require exact recognition, therefore could be expected to deal with goals grow in scale and complexity. Also, the existing scenario for evaluating CIFAR-10 is not used. This is because it does not capture the property of real world data. For example, in (Kirkpatrick et al., 2017), they generated goals by shuffling the order of pixels. Although this leads to the equal difficulty for each goal, it is easy to infer that the procedure disorder features presented in the original images.

From the result, we find that varying goals can improve catastrophic forgetting compared to neural networks learned a single goal in a CIFAR-10 based classification problem. And, we find that when learning a large set of goals, a relatively small switching interval is required to have the advantage of mitigating catastrophic forgetting. On the other hand, when learning a small set of goals, an appropriate large switching interval is preferred since this less worsens the advantage and also can improve accuracy. Moreover, from exploring the obtained neural network structure, we find that, after pruning some unimportant connections, it shows strong motif of bifan and diamond motifs, which suggest that the obtained neural networks are modular, and this could be the reason of mitigating catastrophic forgetting.

This paper is organized as follows. Section 2 briefly explains goals for evaluation. Section 3 shows the effect of CFVG toward classification based on CIFAR-10. Section 4 shows the effect of CFVG toward increment of goals. Section 5 shows network motifs of neural networks obtained by CFVG.

2 GOALS FOR EVALUATION

MVG (Modularly varying goals) is to repeatedly switch between several goals, each made of a different combination of subgoals (Kashtan and Alon, 2005). Although there is no detail information about how to generate the goals, two examples are given in (Kashtan and Alon, 2005). For electronic combinatorial logic circuits problem, the switching goals are given as G1 and G2:

$$\text{G1} = (\text{X XOR Y}) \text{ AND } (\text{Z XOR W}), \quad (1)$$

$$\text{G2} = (\text{X XOR Y}) \text{ OR } (\text{Z XOR W}). \quad (2)$$

where, X, Y, Z, W are inputs, and G1, G2 are outputs. G1 and G2 have share subproblems (X XOR Y) and (Z XOR W). Circuits implementing each goal with NAND gates are explored by GA. Another example is for 8 bit-sized pattern recognition problem. The goal is to recognize objects in the left and right sides of the retina. The switching goals are given as G3 and G4:

$$\text{G3} = \text{L and R}, \quad (3)$$

$$\text{G4} = \text{L or R}. \quad (4)$$

A left object exists if the four left pixels match one of the patterns of a predefined set L. A right object exists if the four right pixels match one of the patterns of a predefined set R. G3 and G4 have shard subproblems L and R. Neural networks implementing each goal are explored by GA.

In this paper we explore the application of MVG in a slightly practical situation, that is applying it to classification of small sized real images, such as CIFAR-10. Since real-world data is neither perfectly separable nor random, it is not the situation considered in (Kashtan and Alon, 2005). Goals are set to learn whether images belong to a given class. Since CIFAR-10 is used for evaluation, for example, the goals can be expressed as:

$$\text{G5} = \text{Airplane}, \quad (5)$$

$$\text{G6} = \text{Ship}. \quad (6)$$

Different goals have different labels for each input, which represents whether the input image is the target goal or not. The input data is the same for every goal. Therefore, changing goals means to change the label. Although the goals are set in a rough fashion, and it is true that the goals do not share clear subproblems, it is popular that images have similar features, e.g. edge, intersecting lines, curves (Li et al., 2015). Since real-world data is rarely perfectly separable, it is valuable to explore using such goals. To distinguish from MVG which is targeted to goals made of a different combination of subgoals, the approach in this paper is renamed as CFVG (mitigate Catastrophic Forgetting by Varying Goals).

In more detail, since the aim of this research is to apply MVG to a practical situation, CIFAR-10 CNN introduced in Keras documentation[1] is used. The dataset used for evaluation is 50,000 32x32 color training images, which is originally labeled over 10 categories. The input data is the same for every goal, which is 50,000 32x32 color training images. The output data is relabeled based on the original labels from 10 categories to 2 categories, which represents whether the input image is the target goal or not. The labels are different for the same input for different goals.

3 EVALUATION FOR CATASTROPHIC FORGETTING

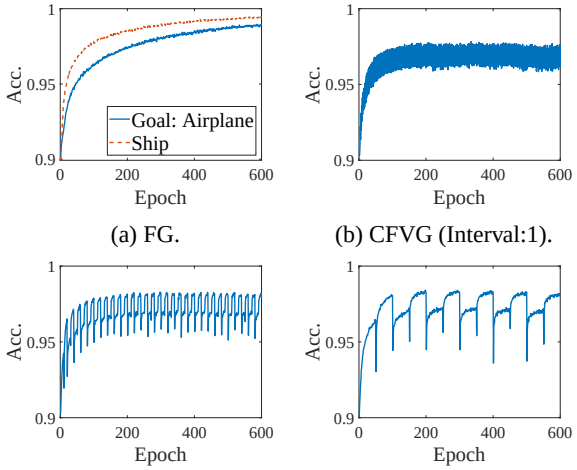
In this section, we show that CFVG can mitigate catastrophic forgetting. Forgetting is discussed by a traditional measurement that measures the amount of time it required to relearn the original goal (French, 1999).

Since the aim of this research is to apply MVG to a practical situation, CIFAR-10 CNN introduced in Keras documentation is used. The dataset used for evaluation is 50,000 32x32 color training images, which is originally labeled over 10 categories(airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck), and relabeled to 2 categories, which represents whether the input image is the target goal or not. The input data is the same for every goal, which is 50,000 32x32 color training images, and the labels are different for each goal. To balance the retagged training data, class weight is set. The layer structure of the neural network is unchanged from CIFAR-10 CNN except for the output layer since the number of categories changed from 10 to 2, which is afterword:

$$input - 32C3 - 32C3 - MP2 - 64C3 - 64C3 - MP2 - 512FC - 2softmax$$

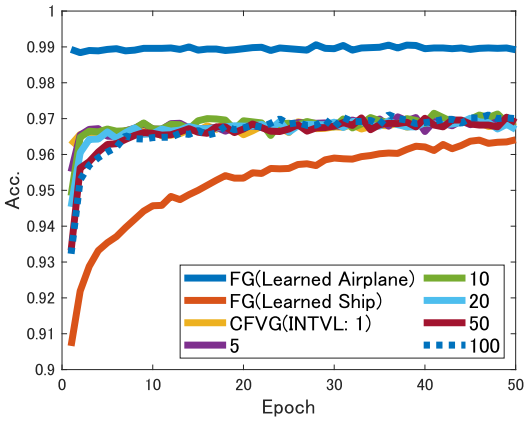
Again, since the aim of this research is to apply MVG to a practical situation, those parameters that have been shown to be useful for practical use is remain unchanged from Keras documentation. The optimization algorithm is changed from RMSprop to SGD without momentum and decay. Since RMSprop decreases the learning rate, it is not compatible with CFVG. The learning rate of SGD is set to 0.1 after several times tuning. Note that, preprocessing and

[1] https://github.com/fchollet/keras



(a) FG. (b) CFVG (Interval:1).

(c) CFVG (Interval:10). (d) CFVG (Interval:50).



(e) Evaluation (Goal: Airplane).

Figure 1: Evaluation of learning 2 goals. (a) Accuracy of neural networks learning airplane and ship respectively(FG). (b, c, d) Accuracy of neural networks learning airplane and ship by CFVG in switching interval 1, 10, 50 respectively. (e) Evaluation against neural network obtained in (a, b, c, d). Accuracy of learning airplane against those neural networks are shown.

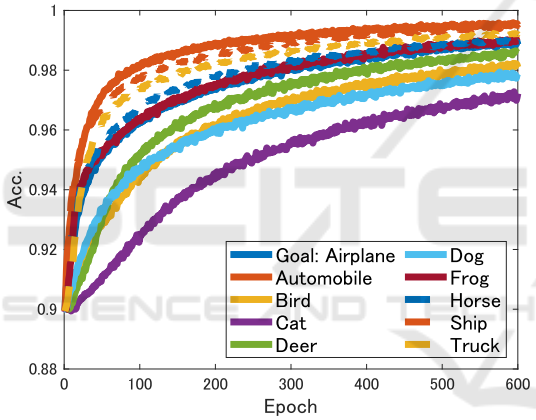


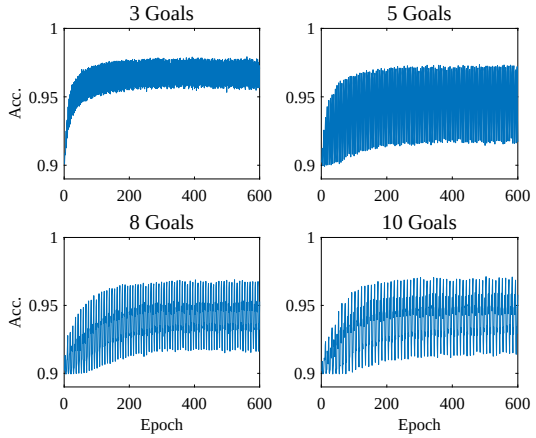
Figure 2: Accuracy of neural networks learning each goal respectively(FG).

data augmentation originated in Keras documentation is left unchanged since those are done for practical use. The accuracy showed in the evaluation below is training accuracy, and it is categorical accuracy. Since this research focusing on catastrophic forgetting, evaluating training accuracy is more clear than showing test accuracy which could be affected by other reasons.

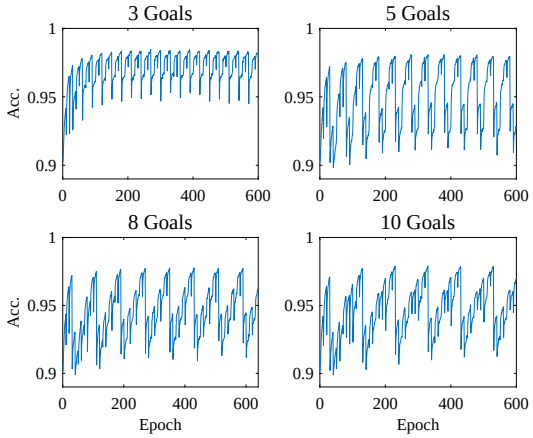
For comparison, we do not compare using existing scenario, since it does not capture the property of real-world data that they are intermediate modular, which is neither perfectly separable nor random, for example, images have similar features, e.g. edge, intersecting lines, curves (Li et al., 2015). In recent research for overcome catastrophic forgetting (Kirkpatrick et al., 2017), for evaluation goals, they generated goals by shuffling the order of pixels. Although this leads to the equal difficulty to each goal, it is easy to infer that the procedure disorder the features presented in the original images. Therefore, instead of comparing with it, we compare with neural networks learned a single goal, which is the most common method in practical use. Note that, to avoid exploding gradients, the learning rate is set to 0.01 for FG, which is less aggressive than that used in CFVG. However, the results below show significant difference that could not be improved only by setting the learning rate.

Figure 1a shows the accuracy of neural networks learning airplane and ship respectively. This is called FG(Fixed Goal). We regard FG as a neural network catastrophically forgotten previous goals. From the result, we can see that both achieved more than 0.98. Figure 1b to Fig. 1d shows the accuracy of neural networks learning airplane and ship by CFVG in switching interval 1, 10, 50 respectively. We can see, although the accuracy trained using CFVG is lower than that of FG, the trained neural network can learn each task faster as the epoch increase.

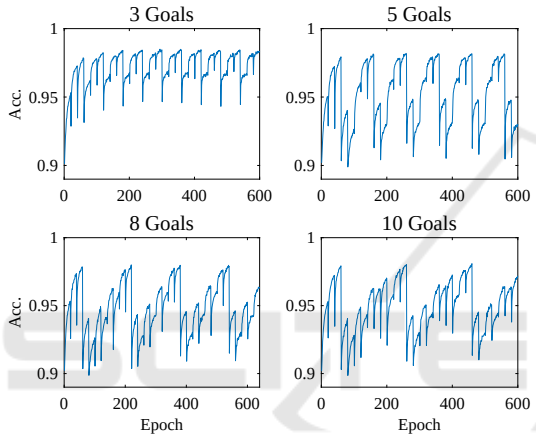
To show the amount of time it required to relearn the original goal, the accuracy of learning airplane is shown for some trained FG and CFVG neural networks. The neural networks used for evaluation are the neural networks trained by airplane and ship using CFVG in switching interval 1, 5, 10, 20, 50, 100 for 600 epoch. From Fig. 1e, we can see that CFVG can reach 0.965 in a few epochs, which is much smaller compare to that with Fig. 1a. Moreover, we can see from Fig. 1 that CFVG neural networks learn the original goal faster than FG learned ship. Therefore, the results suggest that catastrophic forgetting can be mitigated by CFVG.



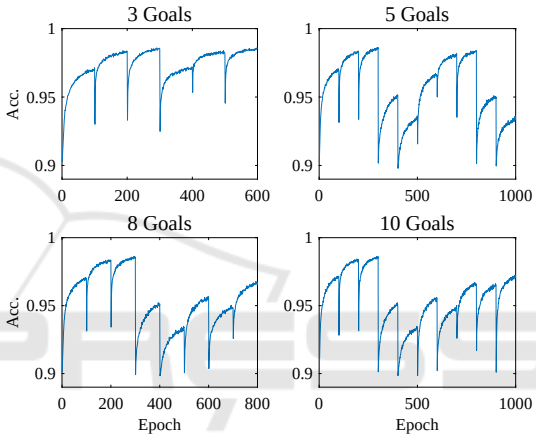
(a) CFVG (Interval: 1).



(b) CFVG (Interval: 10).



(c) CFVG (Interval: 20).



(d) CFVG (Interval: 100).

Figure 3: Accuracy of neural networks learning 3, 5, 8, 10 goals in different switching intervals. (a) Learning with switching interval 1. (b) Learning with switching interval 10. (c) Learning with switching interval 20. (d) Learning with switching interval 100.

4 EVALUATION FOR INCREMENT OF GOALS

In this section, we evaluate CFVG against increment of goals.

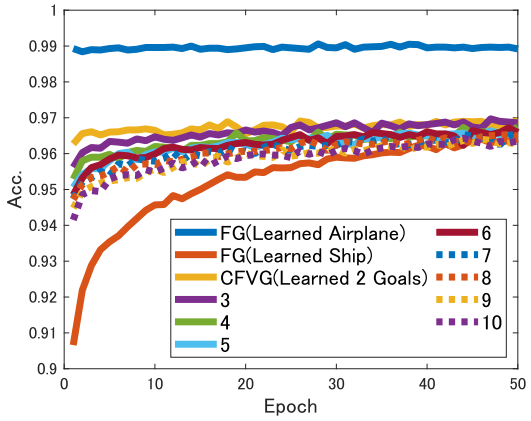
Before showing the results of CFVG, the accuracy of training different classes using FG is shown in Fig. 2. We can see that after 600 epochs the accuracy are all above 0.96. Also, we can see that the accuracy of learning automobile, ship, and truck are higher, and that of bird, dog, and cat is lower. This could be because non-rigid objects like animals are difficult to classify since their intra-class pose and appearance variations are expected to be very high(Ramesh et al., 2019).

Figure 3 shows the accuracy of training by CFVG for a different number of goals with different switching intervals. Goals are given by a certain order as

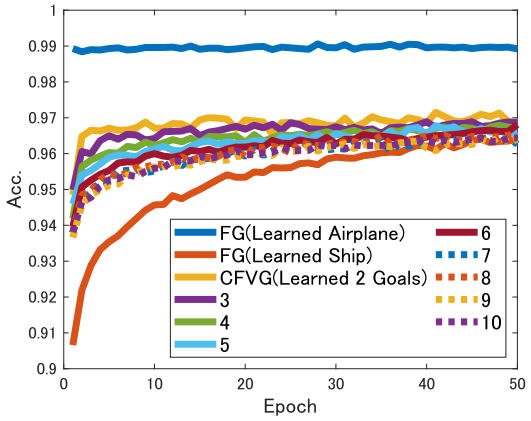
Table 1: Goals For Evaluation.

3 Goals	Airplane, Ship, Automobile
4 Goals	Airplane, Ship, Automobile, Bird
5 Goals	Airplane, Ship, Automobile, Bird, Cat
6 Goals	Airplane, Ship, Automobile, Bird, Cat, Deer
7 Goals	Airplane, Ship, Automobile, Bird, Cat, Deer, Dog
8 Goals	Airplane, Ship, Automobile, Bird, Cat, Deer, Dog, Frog
9 Goals	Airplane, Ship, Automobile, Bird, Cat, Deer, Dog, Frog, Horse
10 Goals	Airplane, Ship, Automobile, Bird, Cat, Deer, Dog, Frog, Horse, Truck

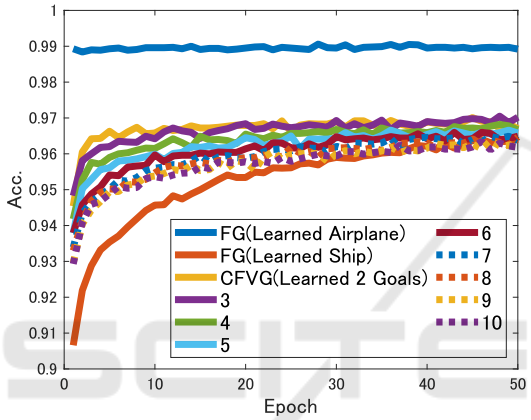
Tab. 1. Figure 3a shows the result of switching interval 1. We can see that, as goals increase, the upper end of accuracy decreases. Figure 3b, Fig. 3c, Fig. 3d shows the results of switching interval 10, 20, 100 re-



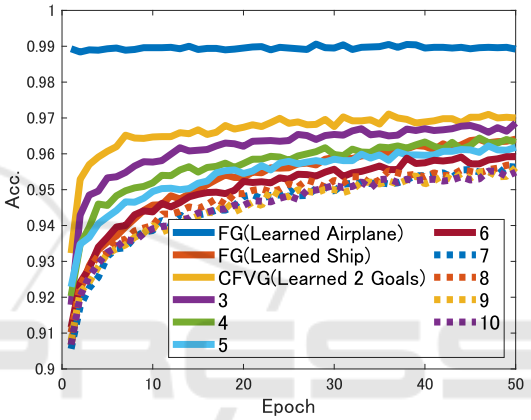
(a) CFVG (Interval: 1).



(b) CFVG (Interval: 10).



(c) CFVG (Interval: 20).



(d) CFVG (Interval: 100).

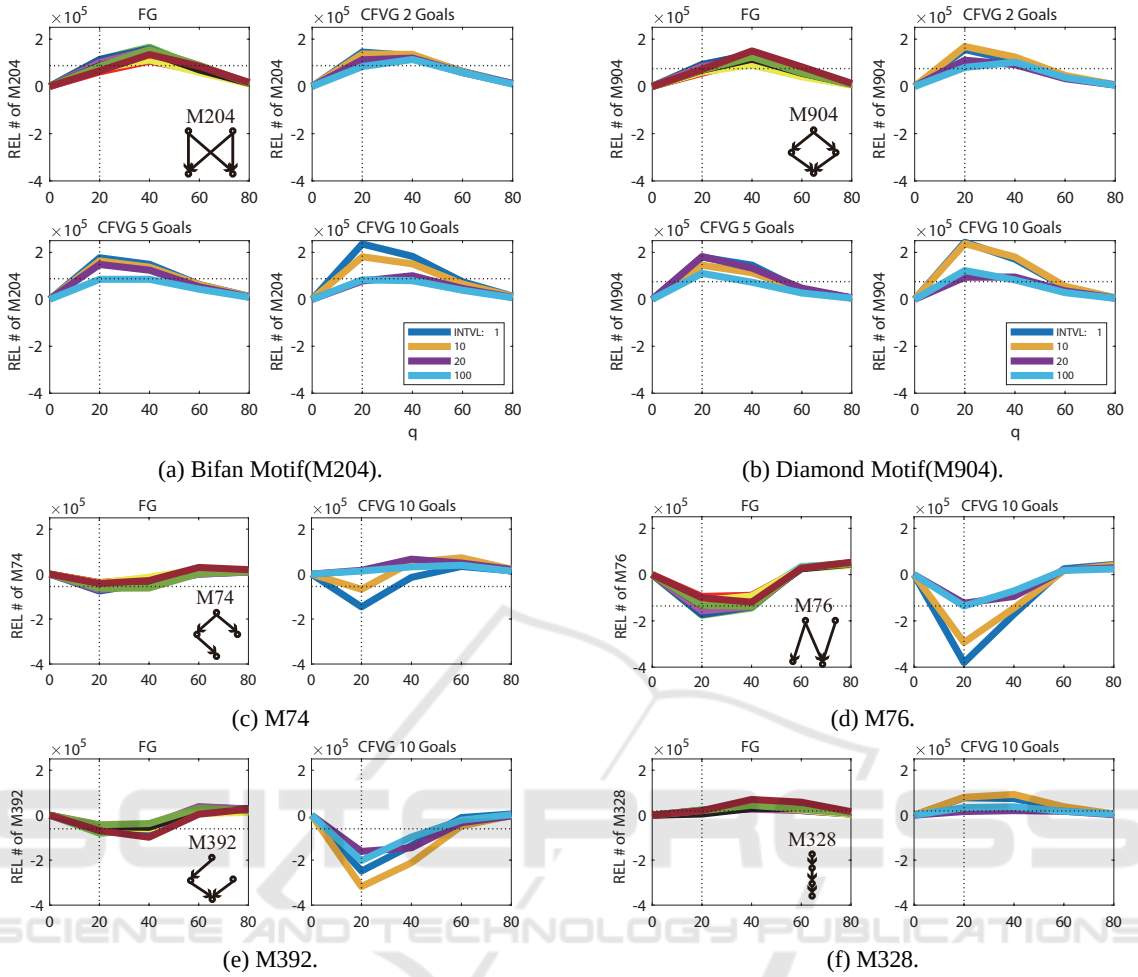
Figure 4: Evaluation against neural network obtained by CFVG(Fig.3). Accuracy of learning airplane against those neural networks are shown. Although Fig.3 only showed several results of learning goals for page limitation, the number of learning goals from 2 to 10 are all evaluated. (a) Learning with switching interval 1. (b) Learning with switching interval 10. (c) Learning with switching interval 20. (d) Learning with switching interval 100.

spectively. We can see that the accuracy of 3 goals increases as the switching interval increase. Also, we can see that the decrease in the upper end become less as switching interval increase.

Figure 4a to Fig. 4d shows the amount of time required to relearn the original goal for neural networks trained by CFVG with different number of goals. Accuracy of learning airplane is shown for some trained CFVG neural networks. The neural networks used for evaluation are the neural networks obtained in the last epoch in Fig. 3. Those are trained until the epoch where the goals go around for the first time beyond 600 epoch. For comparison, The accuracy for neural networks trained using FG against airplane and ship are shown. Figure 4a shows the result of switching interval 1. We can see that CFVG learns faster than FG learned ship. In detail, the smaller number of goals it learns, the faster it learns the original goal. Figure 4b, Fig. 4c, Fig. 4d shows the results of switching interval 10, 20, 100 respectively. We can see that CFVG learns less fast as the interval increase, and for switching interval 100, training with more than 6 goals will reduce the speed of relearning. Therefore, when learning a large set of goals, a small switching interval relative to the total learning epoch is required to have the advantage of mitigating catastrophic forgetting. On the other hand, when learning a small set of goals, an appropriate large switching interval is preferred since this less worsens the advantage and also can improve accuracy.

5 EVALUATION FOR NETWORK MOTIFS

In this section, we show whether CFVG leads to module structure. (Kashtan and Alon, 2005) showed MVG lead to spontaneous evolution of modular net-



(a) Bifan Motif(M204).

(b) Diamond Motif(M904).

(c) M74

(d) M76.

(e) M392.

(f) M328.

Figure 5: The relative number of motifs to weight randomized networks. (a) Bifan motif of FG and CFVG obtained neural networks with switching between 2 goals, 5 goals, and 10 goals. For CFVG, results of switching interval with 1, 10, 20, 100 is shown. For FG, results against 10 different goals are shown, and so on. (b) Same as (a) but for diamond motif. (c, d, e) Motifs with one link removed from bifan and diamond motif of FG and CFVG with switching between 10 goals. (f) Same as (c, d, e) but for a motif supposed to have almost the same number with weight randomized networks.

work and network motifs. Unlike MVG, since CFVG is varying goals that are neither perfectly separable nor random, it is interesting to know whether the obtained neural networks also gain such modular structure. To unclear this, we evaluate network motifs against obtained neural network structure.

We evaluate the number of motifs of obtained CNN to weight randomized neural networks, refer to Z-score which is not calculable because of the network size. Since the entire network of CNN is too large to compute, a network is extracted from convolutional layers of the CNN. We assume channels as nodes. A link exists from a node represent a input channel X to a node represent an output channel Y if the kernel used against X for calculating Y has a larger variance than p in its elements. p is calculated for each layer, and it is the qth percentile of the variance of all the kernels in a layer. In other words, unimportant connections are pruned following q (Hou and Kwok, 2018). Note that, no links are pruned when q is 0. Since the two dense layers have a large number of nodes, they are not included in the network for calculating network motif. However, to delete redundant links in the network, the dense layers are once connected, and links that are unable to reach output layers are deleted. To connect the dense layer, we consider neuron in dense layer as a node. For links between a convolutions layer and a dense layer, it exists if the variance of weights of links headed to the same neuron is above the qth percentile of all the variance. For links between two dense layers, it exists if the link weight is above the qth percentile of all the weights. For weight randomized neural network, the number of layers and the number of nodes in each

layer is set to the same value as the real trained neural network. Then, pruning unimportant connections following q. Although a network with 5 layers is calculated, to set the conditions the same as the real trained neural network, nodes in the dense layers are attached for deleting links that are not reachable to the output layer. The number of motifs is calculated against 10 weight randomized neural networks. The average value of the result is used as a reference value. The value shown in Fig. 5 are values that are subtracted by the reference value. Note that, since the structure of neural networks obtained by FG and CFVG is the same, the same weight randomized networks are used. MFINDER1.21 is used for detecting network motifs (Milo et al., 2004).

From the result of Fig. 5, we can see a tendency that neural network obtained with CFVG have a relatively large number of bifan and diamond motif, while they have less number of those motifs with one link removed from bifan and diamond motif especially when q is 20. Moreover, although FG has the same tendency, the tendency is even stronger in CFVG obtained neural networks. This suggest that neural networks obtained by CFVG are more modular than that by FG. And, the reason for CFVG having the advantage in mitigating catastrophic forgetting could be because of the such structure the networks gained.

6 CONCLUSION AND FUTURE WORK

In this paper, we explore the application of MVG to mitigate catastrophic forgetting in a slightly practical situation, that is applying it to classification of small sized real images and applying it to the increment of goals. From the result, we find that varying goals can improve catastrophic forgetting using SGD in a CIFAR-10 based classification problem. We find that, when learning a large set of goals, a relatively small switching interval is required to have the advantage of mitigating catastrophic forgetting. On the other hand, when learning a small set of goals, an appropriate large switching interval is preferred since this less worsens the advantage and also can improve accuracy. Also, from exploring the obtained neural network structure, we find that, after pruning some unimportant connections, it shows strong motif of bifan and diamond motifs, which suggest that the obtained neural networks are modular, and this could be the reason of mitigating catastrophic forgetting.

For future work, the proposed approach should be examined in other tasks and other layer structures, and theoretical analysis is needed.

ACKNOWLEDGMENT

Thanks to Nadav Kashtan for providing his source code. This work was supported by JSPS KAKENHI Grant Number JP19K20415. The computational resource was partially provided by large-scale computer systems at the Cybermedia Center, Osaka University.

REFERENCES

Alon, U. (2003). Biological networks: The tinkerer as an engineer. *Science*, 301(5641):1866–1867.

Amer, M. and Maul, T. (2019). A review of modularization techniques in artificial neural networks. *Artificial Intelligence Review*, 52(1):527–561.

Ellefsen, K. O., Mouret, J.-B., and Clune, J. (2015). Neural modularity helps organisms evolve to learn new skills without forgetting old skills. *PLoS Computational Biology*, 11(4):e1004128.

French, R. M. (1999). Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135.

Hou, L. and Kwok, J. T. (2018). Power law in sparsified deep neural networks. *arXiv e-prints*, page arXiv:1805.01891.

Kashtan, N. and Alon, U. (2005). Spontaneous evolution of modularity and network motifs. *Proceedings of the National Academy of Sciences*, 102(39):13773–13778.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., and Hadsell, R. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526.

Li, Y., Wang, S., Tian, Q., and Ding, X. (2015). A survey of recent advances in visual feature detection. *Neurocomputing*, 149:736–751.

Milo, R., Itzkovitz, S., Kashtan, N., Levitt, R., Shen-Orr, S., Ayzenshtat, I., Sheffer, M., and Alon, U. (2004). Superfamilies of evolved and designed networks. *Science*, 303(5663):1538–1542.

Newman, M. E. (2006). Modularity and community structure in networks. *Proceedings of The National Academy of Sciences*, 103(23):8577–8582.

Parter, M., Kashtan, N., and Alon, U. (2008). Facilitated variation: How evolution learns from past environments to generalize to new environments. *PLoS Computational Biology*, 4(11):e1000206.

Ramesh, B., Yang, H., Orchard, G. M., Le Thi, N. A., Zhang, S., and Xiang, C. (2019). Dart: Distribution aware retinal transform for event-based cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1.

Sporns, O. and Kötter, R. (2004). Motifs in brain networks. *PLoS Biology*, 2(11):e369.