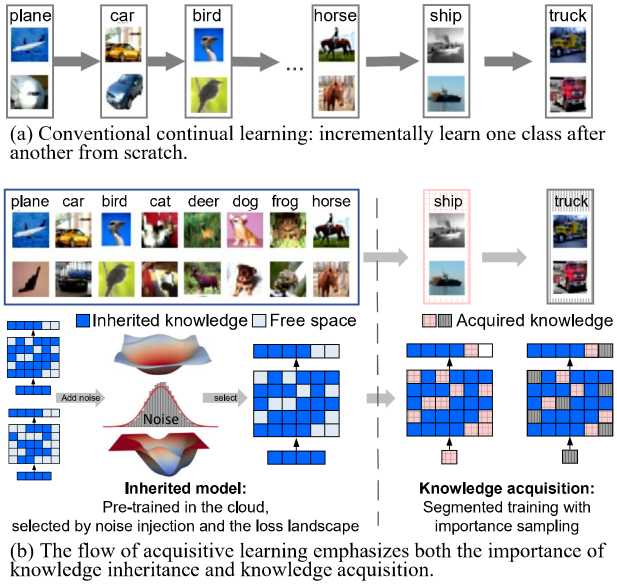
Noise-based Selection of Robust Inherited Model for Accurate Continual Learning

Xiaocong Du[1], Zheng Li[1], Jae-sun Seo[1], Frank Liu[2], Yu Cao[1]
[1]Arizona State University [2]Oak Ridge National Lab
{xiaocong, zhengl11, jseo28, ycao}@asu.edu, liufy@ornl.gov

Abstract

There is a growing demand for an intelligent system to continually learn knowledge from a data stream. Continual learning requires both the preservation of previous knowledge (i.e., avoiding catastrophic forgetting) and the acquisition of new knowledge. Different from previous works that focus only on model adaptation (e.g., regularization, network expansion, memory rehearsal, etc.), we propose a novel training scheme named acquisitive learning (AL), which emphasizes both the knowledge inheritance and knowledge acquisition. AL starts from an elaborately selected model with pre-trained knowledge (the inherited model) and then adapts it to new data using segmented training. The selection is achieved by injecting random noise to various inherited models for better model robustness, which promises higher accuracy in further knowledge acquisition. The approach is validated by the visualization of the loss landscape and quantitative roughness measurement. The combination of the selective inherited model and knowledge acquisition reduces catastrophic forgetting by 10X on the CIFAR-100 dataset.



(a) Conventional continual learning: incrementally learn one class after another from scratch.

(b) The flow of acquisitive learning emphasizes both the importance of knowledge inheritance and knowledge acquisition.

Figure 1. The training flow of (a) conventional continual learning, which starts learning each task from scratch; (b) the proposed acquisitive learning, which acquires new knowledge based on an inherited model.

1. Introduction

Deep neural networks have been widely used in numerous applications such as image classification [10, 14], object detection [4, 12], nature language processing [1, 27], etc. Among them, emerging applications such as self-driving cars, drones, and robots are required to deal with much more dynamic and complicated tasks in real-time. One necessary attribute for such emerging applications is known as *continual learning* [9, 13, 21], which requires the system to continually acquire new knowledge from a data stream and to preserve previously acquired knowledge.

Previous works have proposed different techniques to satisfy the aforementioned needs, aiming to mitigate *catastrophic forgetting*, *i.e*. the learning of new tasks causes overwriting or interfering in model weights. These approaches start learning sequential tasks from a fresh, randomly initialized model, as shown in Figure 1a. However, various extents of catastrophic forgetting still exist [26] and thus, severe accuracy drop on previous tasks is often observed.

On the contrary to artificial intelligent systems, biological intelligent systems adapt to new knowledge based on an inherited model in specific neurophysiological structures, which is selected through a long and careful evolution process [25, 8, 19]. This observation is presented by the Moravec's paradox [19] (tasks humans find complex are easy to teach AI, while simple and sensorimotor skills come instinctively to humans). To some extent, the intelligence in nature may be determined more by the long-term genetics and inheritance rather than the short-term adaptation [25].

Inspired by the Moravec's paradox, we propose a novel training scheme named *acquisitive learning* (AL), as shown in Figure 1b. AL emphasizes the importance of both knowledge inheritance and acquisition: the majority of knowledge is first pre-trained and preserved in the inherited model, and then the model is adapted to the new incoming tasks (the acquisition). Through experiments, we further confirm the vital correlation between *the robustness of the inherited model* and *its acquisition capacity on new knowledge*. Accordingly, we propose a noise-based approach to evaluate and select the inherited model with better robustness. Such an approach is validated by visualizing the loss landscape [11] and measuring the roughness of the landscape with quadratic linear regression. For knowledge acquisition, we use segmented training technique proposed by [2], which freezes important parameters for the previously learned tasks, and only trains the secondary parameters to acquire new knowledge. During this learning process, a tiny amount of fixed-size memory data is used to help preserve previous knowledge.

2. Related work

The conventional approach of continual learning starts from a set of randomly initialized network parameters Θ, and each incoming new task updates entire Θ or partial Θ. Regularization approaches [9, 13] constrain weight update by adding a regularization term in the loss function. Parameter isolation approaches [16, 17] allocate a subset of weights for previous tasks and prune the rest to learn new data. Segmented training approach [2] freezes important weights to preserve learned knowledge and keeps the secondary weights to learn new tasks. Network expansion approaches [24, 22] grow new branches or parameters to include new knowledge. Memory replay approaches [21, 15] train the model with a small subset of previously seen data. However, as the network is not inheriting any prior knowledge, each new task easily shifts the weight distribution, causing catastrophic forgetting.

3. Methodology

In Section 3.1 to 3.2, we describe how we prepare candidate models by pre-training and freezing important weights, then use the noise-based approach to evaluate model robustness. Section 3.3 describes evaluation of this approach by landscape visualization and a proposed roughness measurement. After the inherited model is prepared and carefully selected, we leverage the techniques in progressive segmented training (PST) [2] to infuse new knowledge into the inherited model, which is described in Section 3.4.

3.1. Preparation of the inherited models

Acquisitive learning first trains the network with a few classes of data (these data will not appear as new tasks in future online learning), and then sorts filters (in convolutional layers) and neurons (in fully-connected layers) based on a score that has been proven in [18, 2]:

The score is used to measure how important a unit is to the loss function. For a filter $\Theta_l^o \in \mathbb{R}^{I_l\times K\times K}$ in layer l, the score is mathematically described as:

$$|\Delta\mathcal{L}(\Theta_l^o)| = \sum_{i=0}^{I_l}\sum_{m=0}^{K}\sum_{n=0}^{K}|\frac{\partial\mathcal{L}(\mathcal{Y};\mathcal{X};\Theta)}{\partial\Theta_l^{o,i,m,n}}\Theta_l^{o,i,m,n}|, \quad (1)$$

where $\frac{\partial\mathcal{L}(\mathcal{Y};\mathcal{X};\Theta)}{\partial\Theta_l^{o,i,m,n}}$ is the gradient of the loss function with respect to the parameter $\Theta_l^{o,i,m,n}$.

For a neuron $\Theta_l^t \in \mathbb{R}^{1\times I_l}$ in layer l, the score is mathematically described as:

$$|\Delta\mathcal{L}(\Theta_l^t)| \simeq |\frac{\partial\mathcal{L}(\mathcal{Y};\mathcal{X};\Theta)}{\partial\Theta_l^t}\Theta_l^t| = \sum_{i=0}^{I_l}|\frac{\partial\mathcal{L}(\mathcal{Y};\mathcal{X};\Theta)}{\partial\Theta_l^{t,i}}\Theta_l^{t,i}|, \quad (2)$$

where $\frac{\partial\mathcal{L}(\mathcal{Y};\mathcal{X};\Theta)}{\partial\Theta_l^{t,i}}$ is the gradient of the loss with respect to the parameter $\Theta_l^{t,i}$.

According to this importance score, filters or neurons are sorted and the top β ones in each layer are frozen (*i.e.*, do not update in future training iterations) in the inherited model, while the rest are kept to acquire new knowledge later. We follow the same setting in [2] for β: it should be roughly proportional to the amount of the inherited knowledge.

3.2. Noise injection

After several candidate models are prepared, we use noise injection to evaluate the model robustness. For each layer l in a neural network, we apply noise as below:

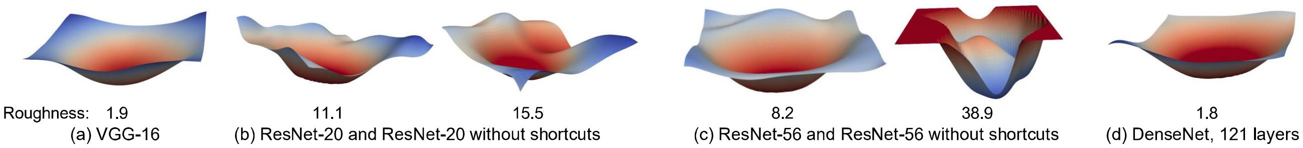
$$\tilde{\Theta_l} = \Theta_l + \alpha\cdot n_l, \quad (3)$$

where Θ_l is the noise-free weight tensor in the l-th layer, α is a constant scaling coefficient, and n_l is the noise tensor of the l-th layer that follows normal distribution $n_l \sim \mathcal{N}(0,\sigma_l^2)$ (σ_l is the standard deviation of Θ_l).

Noise injection methods have been used in other applications such as adversarial attack [6], where noise is treated as a trainable parameter during training. In our work, we perform a one-shot injection of noise to the candidate models. A drop in testing accuracy is observed for the model with noisy tensor $\tilde{\Theta}$ as compared to the model with Θ. Based on this accuracy drop, we are able to monitor the robustness of the inherited model: with noise of the same σ injected, model that has a larger accuracy drop is considered less robust and vice versa. The intuition behind this claim is that a more robust model has a higher tolerance to disturbance.

3.3. Evaluation of the landscape roughness

We further use a visualization tool [11] to visualize the landscape of the loss function and validate the aforementioned noise-based selection approach. A model with a



(a) VGG-16 (b) ResNet-20 and ResNet-20 without shortcuts (c) ResNet-56 and ResNet-56 without shortcuts (d) DenseNet, 121 layers

Figure 2. Landscape visualization of the loss function and the roughness measurement for 6 models. A model (*e.g.* DenseNet-121) with a flatter landscape and a lower roughness measurement is considered to be more robust, and vice versa.

more rough landscape has worse robustness than the one with a flatter landscape [11]. Meanwhile, we propose a quantitative metric to calculate the roughness of the landscape. We fit the three-dimension matrix extracted from visualization tools with quadratic linear regression:

$$\hat{z}_j = w_{j4}x_j^2 + w_{j3}y_j^2 + w_{j2}x_j + w_{j1}y_j + w_{j0}, \quad (4)$$

$$\hat{w} = \underset{w_j}{\text{argmin}} \frac{1}{n} \sum_{j=0}^{n} (z_j - \hat{z}_j)^2, \quad (5)$$

where x, y are the coordinates of the three-dimension matrix, and z is the loss function. Then we obtain the mean square error (MSE), the fitting error of the above regression, to represent the roughness. A landscape with a smaller MSE (*i.e.* the roughness measurement) is considered to be flatter.

3.4. Knowledge acquisition

Following the above approaches, we prepare and select a robust model as the inherited model. Then we use the *importance sampling* and *memory balancing* techniques proposed in PST [2] to infuse new knowledge from a data stream into the inherited model. PST identifies and freezes important filters/neurons for a learned task, and leaves the secondary filters/neurons to learn future new tasks. The memory set is a tiny set (<1% as compared to a full dataset) of uniformly and randomly sampled data from all the previously learned classes. With PST techniques, the acquisitive learning scheme is able to continually acquire new knowledge based on an inherited model. It is worth mentioning that the techniques to consolidate inherited knowledge and to acquire new knowledge are flexible. In this work, we have focused further on the AL scheme.

4. Experimental results

The experiments are performed with PyTorch [20] on one NVIDIA GeForce RTX 2080 platform. We use stochastic gradient descent with a momentum of 0.9 and a weight decay of 0.0005. For each experiment, we run multiple times and report the average accuracy.

Datasets and network: we first train a subset of classes to produce the inherited model, and then we treat the *unseen* classes as new tasks. The balanced memory set contains 200 and 20 images for each class for CIFAR-10 and CIFAR-100, respectively, so that the total memory size is bounded within 2,000 images for both datasets, aligning with previous works [21, 2]. The network structures of VGG-16 [23], ResNets [5], DenseNet [7] used in the following experiment are standard structures following [11].

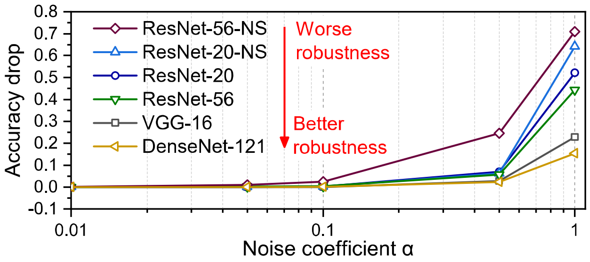


Figure 3. After injecting noise with the same α, more accuracy drop is observed for less robust models, and vice versa. This ranking is consistent with that from the landscape visualization and roughness measurement shown in Figure 2.

Evaluation protocol: we use single-head evaluation, which is considered to be more realistic and challenging as compared to multi-head evaluation [3]. 'Accuracy of the inherited model' refers to the testing accuracy of $(s-1)$-class classifier if the inherited knowledge is $\{X^1, \ldots, X^{s-1}\}$. 'Accuracy on the new task' refers to the testing accuracy of $(t-s+1)$-class classifier for input data $\{X^s, \ldots, X^t\}$ as new observations. 'Overall accuracy' refers to the testing accuracy of t-class classifier on all the data seen so far. 'Accuracy forgetting' refers to the accuracy drop from the accuracy of the inherited model to overall accuracy.

4.1. Robustness of the inherited model

After the preparation of several candidate models, we inject noise to each model following Equation 3 with α = 0.01, 0.05, 0.1, 0.5 and 1.0, and document the accuracy drop caused by this disturbance in Figure 3. For example, with noise of $\alpha = 1.0$ injected, ResNet-56 without shortcuts (ResNet-56-NS) drops 70.9% in accuracy, while DenseNet-121 drops 15.4% in accuracy. Thus, we consider the former model is worse than the latter one in model robustness. Such a claim is verified by the landscape of the loss function [11] and their corresponding roughness in Figure 2: VGG-16, ResNet-20, ResNet-56, and DenseNet-121 have relatively flat landscapes and lower roughness; ResNet-20 without shortcuts (ResNet-20-NS) and ResNet-56 without

Model	Accuracy of the inherited model (9 classes)	Accuracy on the new task (1 class)	$\Delta Accuracy$
ResNet-56-NS	79.0%	59.7%	19.3%
ResNet-20-NS	90.1%	81.0%	9.1%
ResNet-20	91.5%	85.1%	6.4%
ResNet-56	92.3%	86.0%	6.3%
VGG-16	92.7%	86.5%	6.2%
DenseNet-121	93.5%	88.3%	5.2%

Table 1. Acquisition capacity for different models. '9+1' experiment with CIFAR-10 dataset is presented here. The ranking of acquisition capacity is consistent with the robustness shown in Figure 3.

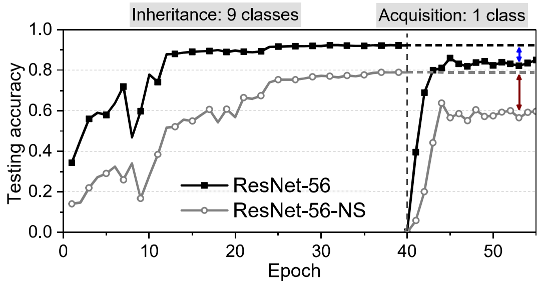


Figure 4. The learning curves for '9+1' experiment on CIFAR-10 dataset. Since ResNet-56 has better robustness than ResNet-56-NS, its accuracy drop after learning a new task is less.

shortcuts (ResNet-56-NS) have relatively sharp landscapes and higher roughness.

We observe that the more robust the inherited model is, the better knowledge acquisition is. For the models shown in Figures 2 and 3, we perform '9+1' experiments on CIFAR-10 dataset, where '9+1' means that 9 classes are pre-trained and deployed in the inherited model and 1 class is learned as the new acquisition. Table 1 presents the pre-trained accuracy on 9 classes and the accuracy on the new class, and the corresponding drop in accuracy, *i.e.* $\Delta accuracy$. $\Delta Accuracy$ reveals the generalization ability of the pre-trained model on new observations, *i.e.* the acquisition capacity of the inherited model. The lower $\Delta accuracy$ is, the better acquisition capacity is. We further focus on two models, ResNet-56 and ResNet-56-NS, and plot their learning curve in Figure 4. ResNet-56-NS has worse acquisition capacity on the new task than ResNet-56. These results indicate that the quality of the inherited model, particularly its model robustness, is a vital factor in knowledge acquisition.

4.2. Amount of the inherited knowledge

We mimic different amounts of inherited knowledge using different numbers of pre-trained classes. In Figure 5, 'X+Y' means that X classes are pre-trained in the inherited model and Y classes need to be acquired. There is no overlap between X and Y. For instance, the accuracy forgetting after learning 10 new classes is 40.5% for '10+10' case but only 7.7% for '90+10' case. The more knowledge embedded in the inherited model, the less forgetting in acquisition. Such a trend gradually saturates.

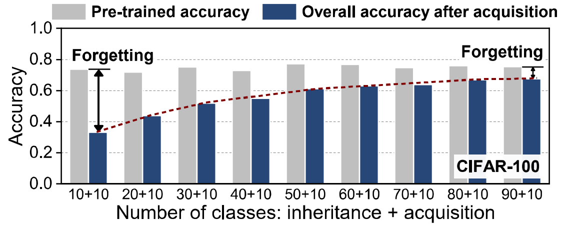


Figure 5. Accuracy drop is minimized with the an increasing amount of knowledge in the inherited model.

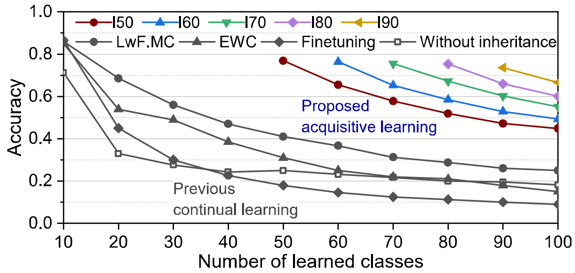


Figure 6. The comparison of overall accuracy between conventional continual learning and the proposed acquisitive learning when incrementally learning 10 classes in a sequence on CIFAR-100. 'I50' means that AL starts training from an inherited model that is pre-trained on 50 classes, and so on.

4.3. Learning from a data stream with AL

We design experiments to verify the efficacy of acquisitive learning when learning in a data stream. To align with previous works, we use ResNet-20 as the learning model here, although it is not the most robust model according to Section 4.1. In Figure 6, we simulate conventional continual learning approaches [9, 13] that starts learning from scratch and learns each task (10 classes from CIFAR-100) in a sequence. The overall single-head accuracy of the conventional approaches are plotted in gray. *Finetuning* denotes the simulation that the network trained on previous tasks is directly fine-tuned by new tasks, without strategies to prevent catastrophic forgetting. *LwF.MC* denotes the method that uses LwF [13] but is evaluated with multi-class single-head classification. *EWC* denotes the approach proposed in [9]. *Without inheritance* means that we start the training from a randomly initialized model and acquire 10 new classes using importance sampling and memory balancing.

On the other hand, by assuming inherited knowledge contains much more classes than new observations, we prepare the inherited models (following Section 3.1) with 50 to 90 pre-trained classes from CIFAR-100 dataset and then incrementally train the model (following Section 3.4) with 10 new classes from the rest of the dataset. β is set as 0.5 for the inherited model in 'I50' experiment, and similarly, 0.9 for the inherited model in 'I90' experiment. The overall single-head accuracy of the AL is plotted in color. For CIFAR-100, the conventional scheme forgets 61.0%-76.0% accuracy after learning 100 classes, while acquisitive learning limits the accuracy drop to 7.1% in the best case, reducing the accuracy forgetting by $\sim 10\times$.

5. Conclusion

In this paper, we propose acquisitive learning that emphasizes the importance of both knowledge inheritance and knowledge acquisition. We validate that the robustness of the inherited model is strongly related to knowledge acquisition and thus, the inherited model should be carefully selected. We further propose to inject noise to select the most robust inherited model and validate it by landscape visualization and roughness measurement. With extensive experiments, we demonstrate that the combination of the above steps reduces accuracy drop by $10\times$ on CIFAR-100 dataset.

Acknowledgment

This work was supported in part by the Semiconductor Research Corporation (SRC) and DARPA. It was also partially supported by National Science Foundation (NSF) under CCF #1715443.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 1

[2] Xiaocong Du, Gouranga Charan, Frank Liu, and Yu Cao. Single-net continual learning with progressive segmented training. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pages 1629–1636, Dec 2019. 2, 3

[3] Sebastian Farquhar and Yarin Gal. Towards robust evaluations of continual learning. *arXiv preprint arXiv:1805.09733*, 2018. 3

[4] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. 1

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[6] Zhezhi He, Adnan Siraj Rakin, and Deliang Fan. Parametric noise injection: Trainable randomness to improve deep neural network robustness against adversarial attack. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 588–597, 2019. 2

[7] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 3

[8] Madhura Ingalhalikar, Alex Smith, Drew Parker, Theodore D Satterthwaite, Mark A Elliott, Kosha Ruparel, Hakon Hakonarson, Raquel E Gur, Ruben C Gur, and Ragini Verma. Sex differences in the structural connectome of the human brain. *Proceedings of the National Academy of Sciences*, 111(2):823–828, 2014. 1

[9] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 1, 2, 4

[10] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012. 1

[11] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In *Advances in Neural Information Processing Systems*, pages 6389–6399, 2018. 2, 3

[12] Zheng Li, Xiaocong Du, and Yu Cao. Gar: Graph assisted reasoning for object detection. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 1295–1304, 2020. 1

[13] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017. 1, 2, 4

[14] Zheng Li, Chenchen Liu, Hai Li, and Yiran Chen. Neuromorphic hardware acceleration enabled by emerging technologies. In *Emerging Technology and Architecture for Big-data Analytics*, pages 217–244. Springer, 2017. 1

[15] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, pages 6467–6476, 2017. 2

[16] Arun Mallya, Dillon Davis, and Svetlana Lazebnik. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 67–82, 2018. 2

[17] Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7765–7773, 2018. 2

[18] Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. Pruning convolutional neural networks for resource efficient inference. *arXiv preprint arXiv:1611.06440*, 2016. 2

[19] Hans Moravec. *Mind children: The future of robot and human intelligence*. Harvard University Press, 1988. 1

[20] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017. 3

[21] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 1, 2, 3

[22] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. 2

[23] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 3

[24] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. *arXiv preprint arXiv:1708.01547*, 2017. 2

[25] Anthony M Zador. A critique of pure learning and what artificial neural networks can learn from animal brains. *Nature communications*, 10(1):1–7, 2019. 1

[26] Junting Zhang, Jie Zhang, Shalini Ghosh, Dawei Li, Serafettin Tasci, Larry Heck, Heming Zhang, and C-C Jay Kuo. Class-incremental learning via deep model consolidation. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 1131–1140, 2020. 1

[27] Peng Zhang, Yash Goyal, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Yin and yang: Balancing and answering binary visual questions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5014–5022, 2016. 1