Learning Efficient Deep Feature Representations via Transgenerational Genetic Transmission of Environmental Information during Evolutionary Synthesis of Deep Neural Networks

M. J. Shafiee†, E. Barshan†, F. Li†, B. Chwyl†, M. Karg‡, C. Scharfenberger‡, A. Wong†
†Vision & Image Processing Research Group, University of Waterloo, Waterloo, ON, Canada
‡ ADC Automotive Distance Control Systems GmbH, Continental, Germany
†{mjshafiee, ebarshan, f27li, bchwyl, a28wong}@uwaterloo.ca
‡{Michelle.Karg, Christian.Scharfenberger}@continental-corporation.com

Abstract

The computational complexity of deep neural networks for extracting deep features is a significant barrier to widespread adoption, particularly for use in embedded devices. One strategy to addressing the complexity issue is the evolutionary deep intelligence framework, which has been demonstrated to enable the synthesis of highly efficient deep neural networks that retain modeling performance. Here, we introduce the notion of trans-generational genetic transmission into the evolutionary deep intelligence framework, where the intra-generational environmental traumatic stresses are imposed to synapses during training to favor the synthesis of more efficient deep neural networks over successive generations. Results demonstrate the efficacy of the proposed framework for synthesizing networks with significant decreases in synapses (e.g., for SVHN dataset, a 230-fold increase in architectural efficiency) while maintaining modeling accuracy and a significantly more efficient feature representation.

1. Introduction

Deep neural networks [7, 14, 15] have demonstrated tremendous success for learning powerful feature representations from data, leading to state-of-the-art performance in a variety of different applications over the past decade such as object detection [11, 19], semantic image segmentation [2, 17], image classification [14, 23], speech recognition [7, 8], and gene sequencing [1].

There are two contributing factors to the immense success of deep neural networks experienced in recent years for learning feature representations. First, the learning process of deep neural networks can be considered as an end-to-end approach where feature extraction and inference are trained simultaneously based on training data. As such, the different layers of a deep neural network learn the best possible feature representation jointly, for not only fitting the training data but also for optimizing inference performance. In convolutional neural networks [14], the convolution layers and the fully connected layers are designed for feature extraction and inference, respectively, and can compensate for the modeling deficiencies of each other when trained together, thus leading to feature representations that are optimized for inference performance. Given the rise in big data, deep neural networks can learn highly powerful feature representations around such data for very high inference performance.

The second contributing factor is the significant growth of computational power in recent years. The rise in parallel computing devices such as graphics processing units (GPUs) and distributed computing systems has greatly accelerated the training and inference of deep neural networks [14]. The achievements in high-performance computing devices enable researchers to design and build larger and deeper neural networks [23] that can produce more and more powerful feature representations.

Many applications require efficient algorithms that are suitable for embedded systems and consider runtime and memory limits. High-performance deep neural networks are both computationally expensive and require large memories to store an enormous number of network parameters. Fast data transmission is required additionally to support the expensive computation and to load the large network parametrization. These issues associated with computational complexity, memory complexity and bandwith can be considered as the main barriers to widespread adoption of deep neural networks for feature extraction in a variety of operational scenarios and applications. This includes self-driving cars, smart-phone applications, and surveillance cameras where the computational resources are limited due to embedded GPUs or even CPUs, and where very limited memory is available. Due to this fact, there has been a very strong recent interest towards obtaining efficient deep neural networks capable of producing efficient deep features. The majority of methods in previous literature on achieving efficient deep neural networks can be grouped into two main categories: I) methods addressing memory complexity associated with deep neural networks, and II) methods focusing on computational and memory complexity issues together.

In the area of methods tackling memory complexity, Lecun *et al.* [16] addressed this issue in their seminal paper by proposing the optimal brain damage method where synapses were pruned based on their strengths. They utilized the second-derivative information to specify the neuron to be pruned and made a trade-off between the number of parameters and training error. Gong *et al.* [5] took advantage of information-theoretical vector-quantization methods to compress the parameters of the network. They used k-means clustering on the weights to quantize the parameters of the dense connected layers. To further reduce the network structure, Han *et al.* [6] proposed the combination of pruning, quantization and Huffman coding.

In the area of methods addressing computational and memory complexity issues simultaneously, low-rank matrix factorization [9, 10] was proposed to approximate the filter structures and convolutional kernels in convolutional layers. For example, Ioannou *et al.* [9] proposed a new training approach such that the network learns a set of small basis filters from scratch via low-rank matrix factorization and by using smaller kernel size addresses the running-time issue. Although Jaderberg *et al.* [10] took advantage of low-rank matrix factorization to learn separable smaller kernels like [9], the separable kernels are optimized after training the network. Wen *et al.* [25] suggested applying regularization techniques to learn the kernel structures and account for structured sparsity, and introduced a new regularization approach to learning the filter shapes and layer depth during training.

Another promising approach to tackling both the computational and memory complexity issues simultaneously is the evolutionary deep intelligence (EvoNet) framework proposed by Shafiee *et al.* [21], where inspirations from evolutionary biology such as random mutation, natural selection, and heredity were leveraged within a probabilistic framework to synthesize increasingly efficient deep neural networks over successive generations, resulting in the learning of highly efficient yet powerful feature representations. While previous works have explored the use of evolutionary computing methods for training and generating deep neural networks [24, 26], they have not only largely focused on accuracy and not on progressively more efficient deep neural networks, but also have leveraged classical methods such as genetic algorithms and evolutionary programming which differs greatly from the probabilistic generative framework proposed in [21].

One of the key aspects of the evolutionary deep intelligence framework greatly influencing the efficiency and quality of the synthesized offspring deep neural networks is the genetic encoding scheme, which acts as a probabilistic 'DNA' to mimic the heredity aspect of biological evolution. For instance, Shafiee & Wong [22] extended the genetic encoding scheme to synthesize deep neural networks with architectures that enable more efficient inference on parallel computing devices such as GPUs. More specifically, they proposed a new genetic encoding scheme to promote the formation of highly sparse sets of synaptic clusters, thus tailoring them to the hardware architecture of GPUs that can execute a set of kernel computing instructions in a highly parallel manner.

To explore the optimization of the genetic encoding scheme even further for learning more efficient feature representations, we take inspiration from a very interesting observation in biological evolution: transgenerational genetic transmission of environmental information. Dias & Ressler [4] studied the inheritance of parental traumatic exposure to their offsprings and found that environmental stimuli imposed on the exposed parents – here, an olfactory traumatic exposure on mice– had a strong genetic influence on their offsprings that were not conceived at the time. Similarly, Klosin *et al.* [12] showed that environmental information, induced by environmental stresses experienced during the lifetime of C. elegans, was transmitted genetically to subsequent generations.

This biological observation brings up an interesting idea: Evolutionary deep intelligence generates several generations of offspring networks, and raises the question: *Would it be possible to mimic the imposition of environmental stresses on an ancestor network during training, so that it results in a genetic encoding favoring the synthesis of more efficient offspring networks?*

To explore this idea as shown in Figure 1, we propose the notion of learning efficient deep feature representations via transgenerational genetic transmission of environmental information using genetic encoding during the evolutionary synthesis of deep neural networks, which we will refer to here as tEvoNet. More specifically, the training of a deep neural network is formulated as a maximum a posteriori (MAP) framework, where intra-generational environmental traumatic stresses to synapses are encoded within the prior model such that the distribution of synaptic strength in an exposed parent deep neural network is tailored to exhibit inherent genetic encodings to favor offspring neural networks with greater efficiency during the synthesis process, thus transmitting the environmental information experienced by a deep neural network from generation to generation.

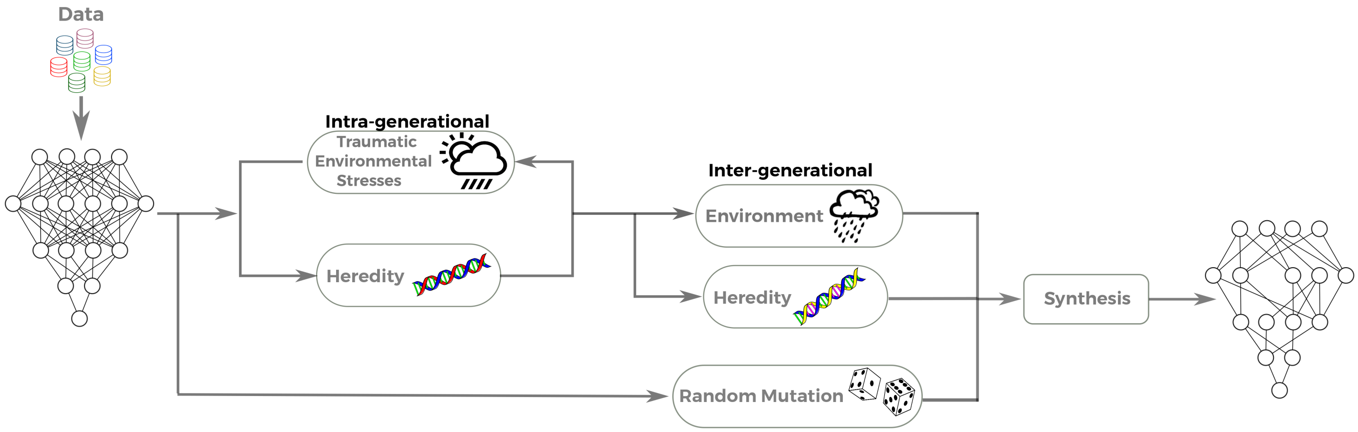

Figure 1. Overview of the proposed tEvoNet evolutionary deep intelligence framework with (i.e., for one generation). The architectural traits of ancestor networks are encoded via probabilistic 'DNA' sequences. The environmental factors are applied at intra-generational and during each epoch of training which enables synapses to be prepared for a traumatic exposure between generation. A new offspring network is synthesized in each generation based on the probabilistic 'DNA' sequences (heredity), inter-generation environmental factors (natural selection), and random mutation.

2. Methodology

We propose tEvoNet, an extended evolutionary deep intelligence framework for learning efficient deep feature representations by using transgenerational genetic transmission of environmental information. In this section, we first explain the concept of evolutionary deep intelligence and the notion of genetic encoding within that framework. We then describe in detail the proposed transgenerational genetic transmission scheme.

2.1. Evolutionary Deep Intelligence

The approach to evolutionary deep intelligence was proposed by Shafiee *et al.* [21], where progressively more efficient deep neural networks are synthesized within a probabilistic framework over multiple generations by leveraging processes that mimic heredity, natural selection and random mutation. More specifically, the architectural traits of a deep neural network are modeled by synaptic probability models that can be considered as the probabilistic 'DNA', and that are used to mimic heredity to pass genetic information to subsequent generations. Offspring deep neural networks with diverse network architectures are synthesized stochastically based on this probabilistic 'DNA' together with probabilistic computational environmental factor models for encouraging progressively increasing network architecture efficiency over generations.

The network architecture of a deep neural network at generation g can be characterized by $\mathcal{S}_g$, where $\mathcal{S}_g = \left\{ s_g^{l,i} \right\}_{l=1:L}^{i=1:I_l}$ is the set of binary states defining the existence of all possible synapses in the network having L possible layers, and I_l possible synapses at each layer l. An offspring deep neural network is synthesized stochastically by a synthesis probability $P(S_g)$, which is expressed by

$$P(S_g) = P(\mathcal{S}_g|\mathcal{W}_{g-1}) \cdot F, \tag{1}$$

with $P(\mathcal{S}_g|\mathcal{W}_{g-1})$ the synaptic probability model, and F the imposed environmental factors. The offspring networks are then trained at each generation to achieve modeling accuracy while preserving the efficiency and architectural diversity.

2.2. Genetic Encoding

We wish to focus on modeling the architectural traits by an effective genetic encoding scheme in this work. Following [21], the genetic encoding of offspring networks is modeled by $P(\mathcal{S}_g|\mathcal{W}_{g-1})$ where $\mathcal{W}_{g-1}$ represents the set of trained synaptic strengths of the network at generation $g-1$ based on the notion, that the desired traits to be inherited by the offspring networks are related to strong synaptic strengths in the ancestor networks. The synaptic strength of $s_{g-1}^{l,i}$ is represented by $w_{g-1}^{l,i} \in \mathcal{W}_{g-1}$, and the non-existence of a synapse is encoded as $w_{g-1}^{l,i} = 0$ and equals $s_{g-1}^{l,i} = 0$.

Shafiee *et al.* [20, 22] further decomposed $P(\mathcal{S}_g|\mathcal{W}_{g-1})$ into a multi-factor probability distribution to promote the formation of synaptic clusters, resulting in the synthesis of offspring deep neural networks that are tailored to be more efficient for computation on parallel computing systems:

$$P(\mathcal{S}_g|\mathcal{W}_{g-1}) = \prod_{c\in C} \left[P(S_g^c|\mathcal{W}_{g-1}) \cdot \prod_{(i,l)\in c} P(s_g^{l,i}|w_{g-1}^{l,i}) \right] \tag{2}$$

where $S_g^c \subset \mathcal{S}_g$ is a cluster of synapses at generation g. A cluster c can be encoded as a subset of synapses of the network, with a filter or a kernel inside a filter as examples of synaptic clusters in the genetic encoding scheme (2).

The creation of the genetic encoding $P(\mathcal{S}_g|\mathcal{W}_{g-1})$ is very important as it has a significant influence over the network architectures of offspring deep neural networks. As mentioned before, the genetic encoding scheme previously proposed is highly dependent on the synaptic strengths of the ancestor deep neural network, i.e., $\mathcal{W}_{g-1}$; therefore, optimizing the distribution of synaptic strengths in $\mathcal{W}_{g-1}$ in a way that promotes optimal genetic encoding $P(\mathcal{S}_g|\mathcal{W}_{g-1})$ favoring the synthesis of offspring neural networks with greater architectural efficiency is highly desired.

2.3. Transgenerational Genetic Transmission of Environmental Information

Inspired by [4, 12], the incorporation of transgenerational genetic transmission of environmental information within the deep evolutionary intelligence framework for improving the architectural efficiency of offspring deep neural networks can be described as follows. The general idea is the imposition of epoch-level traumatic environmental stresses to weaken the strengths of a subset of synapses. Here, the intra-generational environmental stresses imposed on the exposed parent deep neural network influence the distribution of synaptic strengths in a deep neural network to favor offspring network architectural efficiency. This effect is transmitted genetically to the next generation, i.e., by intergenerational transmission using probabilistic genetic encoding. More specifically, the intra-generational environmental stresses encourage configurations of $\mathcal{W}_{g-1}$ that enable more effective genetic encodings $P(\mathcal{S}_g|\mathcal{W}_{g-1})$ linked to synthesized offspring networks with greater architectural efficiencies.

Let us model a neural network as a probabilistic model [3] $P(y|x;\mathcal{W})$ where $x \in \mathbb{R}^d$ is the d-dimensional input to the network, and the network assigns a probability to each possible output $y \in \mathcal{Y}$ regarding the set of trained synaptic strengths $\mathcal{W}$. The learning process of synaptic strengths $\mathcal{W}$ within a deep neural network can be formulated as a maximum likelihood estimation (MLE) given a set of training data $\mathcal{D} = (x_i, y_i)$:

$$\begin{aligned}\hat{\mathcal{W}} &= \underset{\mathcal{W}}{arg\ max} \quad \log P(\mathcal{D}|\mathcal{W}) \\ &= \underset{\mathcal{W}}{arg\ max} \quad \sum_i \log P(y_i|x_i;\mathcal{W}). \end{aligned} \tag{3}$$

This optimization is usually performed by a gradient descent approach with the assumption that $\log P(\mathcal{D}|\mathcal{W})$ is differentiable in $\mathcal{W}$.

We now wish to impose prior knowledge to the synaptic strengths $\mathcal{W}$, and re-formulate the problem as a maximum a posteriori (MAP) problem:

$$\begin{aligned}\hat{\mathcal{W}} &= \underset{\mathcal{W}}{arg\ max} \quad \log P(\mathcal{W}|D) \\ &= \underset{\mathcal{W}}{arg\ max} \quad \log P(D|\mathcal{W}) + \log P(\mathcal{W}) \end{aligned} \tag{4}$$

where $P(\mathcal{W})$ is the prior model imposed during the training stage. Here, we use the prior model to encode the intra-generational environmental traumatic stresses imposed to synapses during the training at each generation. As shown in Figure 1, we encode the intra-generational environmental traumatic stresses to synapses within the prior model during the training of the deep neural network at each generation.

Given the goal that $P(\mathcal{S}_g|\mathcal{W}_{g-1})$ better promotes the synthesis of offspring networks with more effective and efficient network architectures:

$$P(\mathcal{S}_g|\mathcal{W}_{g-1}) \approx P\Big(\mathcal{S}_g|\hat{\mathcal{W}}_{g-1}\Big), \tag{5}$$

we take advantage of (4) and benefit from $P(\mathcal{W}) := P(\hat{\mathcal{W}}_{g-1})$ to provide a more effective genetic encoding scheme.

Here, the prior model, $P(\mathcal{W}_{g-1})$, is realized as a Binomial probability distribution such that the strengths of a subset of synapses are weakened at each epoch level during the training, and is formulated as follows:

$$\begin{aligned}\hat{\mathcal{W}}_{t+1,g-1} &= \Big[Q_{t,g-1} \geq \bar{U}\Big] \cdot \hat{\mathcal{W}}_{t,g-1} \\ &\quad + \beta \cdot \Big[Q_{t,g-1} < \bar{U}\Big] \cdot \hat{\mathcal{W}}_{t,g-1} \end{aligned} \tag{6}$$

where $Q_{t,g-1}$ is the Binomial distribution formulating such that $P(\hat{\mathcal{W}}_{t,g-1})$, $\bar{U}$ is a set of uniformly distributed random numbers based on uniform distribution $U(0,1)$, $\Big[\cdot\Big]$ is the Iverson bracket determining whether a synapse is selected in (6) at epoch t for generation $g-1$, and $\hat{\mathcal{W}}_{t,g-1}$ encodes the set of trained synaptic strengths of epoch t at generation g. The Binomial distribution $Q_{t,g-1}$ is formulated based on the trained synaptic strengths $\hat{\mathcal{W}}_{t,g-1}$ at epoch t:

$$Q_{t,g-1} = q^1_{t,g-1} \cdot q^2_{t,g-1} \cdot \dots \cdot q^n_{t,g-1} \tag{7}$$

$$q^i_{t,g-1} = \exp\Big(\frac{\hat{w}^i_{t,g-1}}{z^i} - 1\Big) \quad 1 \leq i \leq n \tag{8}$$

where $q^i_{t,g-1}$ is a Bernoulli distribution for the ith synapse in a network containing n synapses computed based on $\hat{w}^i_{t,g-1} \in \hat{\mathcal{W}}_{t,g-1}$, and z^i a normalization factor.

The factor $0 < \beta \leq 1$ is the intra-generational environmental factor applied at each epoch of training to weaken the strength of stochastically selected synapses. The factor β imposes minor environmental traumatic stress to the deep neural network at the epoch level. These stochastically selected synapses at each epoch are meant to be less important to the modeling power of the deep neural network than other synapses, and weakening them has a minimal effect on the modeling accuracy. However, the cumulation of tiny changes shapes the distribution of synaptic strengths to promote the formation of a synaptic probability model $P\left(\mathcal{S}_g|\hat{\mathcal{W}}_{g-1}\right)$ favoring the synthesis of offspring deep neural networks with more efficient yet effective network architectures.

3. Experimental Results

To evaluate the efficacy of the proposed tEvoNet framework for learning efficient yet powerful feature representations, we investigated and tested the framework on the CIFAR10 [13] and SVHN [18] benchmark datasets, built for the purpose of image classification.

3.1. Experimental Setup

The CIFAR10 image dataset [13] consists of $50,000$ training, and $10,000$ test images with a size of 32×32 pixels, and 10 different object classes. The SVHN image dataset [18] consists of $604,388$ training and $26,032$ test images captured of of digits in natural scenes. Each image is an RGB image with a size of 32×32 pixels. Examples from the CIFAR10 and SVHN datasets are shown in Figure 2.

For the experiments performed on these two datasets, an AlexNet architecture [14] is selected as the network architecture of the original, first-generation ancestor network, with the first layer modified to utilize $5 \times 5 \times 3$ kernels instead of $11 \times 11 \times 3$ kernels given the smaller images in the two datasets.

For the environmental factor model being imposed at different generations, F is formulated here such that an offspring deep neural network should not have more than 80% of the total number of synapses in its direct ancestor network. The intra-generational factor β applied at each epoch of training to weaken the stochastically selected synapses is set to $\beta = 0.7$ in this study, and the environmental stress was imposed for 30 epochs at each generation. Finally, for this study, successive offspring deep neural networks were synthesized one generation after another generation until the modeling accuracy of the offspring deep neural network at the last generation exceeds a performance drop of 3% when compared to the first-generation ancestor network.

3.2. Discussion

Our approach tEvoNet is compared against the cluster-driven evolutionary deep intelligence framework EvoNet [22] to demonstrate the efficacy of the transgenerational genetic transmission of environmental information during the evolutionary synthesis of deep neural networks.

Figure 2. Example images from the CIFAR10 and SVHN datasets.

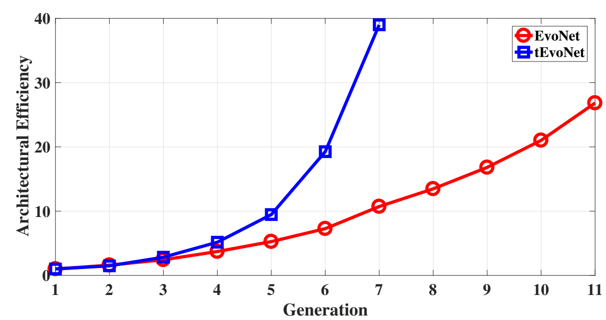

Figure 3. Architectural efficiency vs. generation for CIFAR10, with a stopping criteria of a performance drop of 3% in modeling accuracy. The proposed tEvoNet framework is compared against EvoNet [22]. tEvoNet can synthesize offspring deep neural networks with a 40-fold increase of architectural efficiency with a performance drop of $\sim$3% in modeling accuracy after 7 generations, while EvoNet can synthesize a network with 27-fold architectural efficiency at similar accuracy after 11 generations only.

3.3. Experiment 1: CIFAR10

Figure 3 shows a comparison of the architectural efficiency over generations on the CIFAR10 dataset for the tEvoNet framework and the EvoNet framework [22]. We wish to define the architectural efficiency as the total number of synapses of the first-generation ancestor network divided by the total number of synapses of the current generation. Figure 3 shows that the proposed tEvoNet framework can synthesize an offspring deep neural network with a 40-fold increase of architectural efficiency after 7 generations for the CIFAR10 dataset, while the EvoNet framework

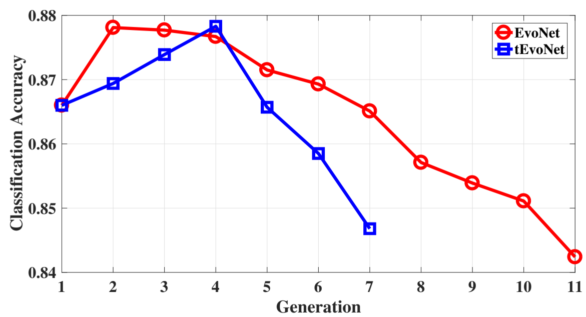

Figure 4. The modeling accuracy over the network generation using the CIFAR10 dataset, with a stopping criteria of a performance drop of 3% in modeling accuracy. It can be observed that offspring deep neural networks synthesized by tEvoNet and EvoNet can largely preserve modeling accuracy over multiple generations, although tEvoNet synthesizes offspring deep neural networks with significantly greater architectural efficiency and achieves higher architectural efficiency at earlier generations.

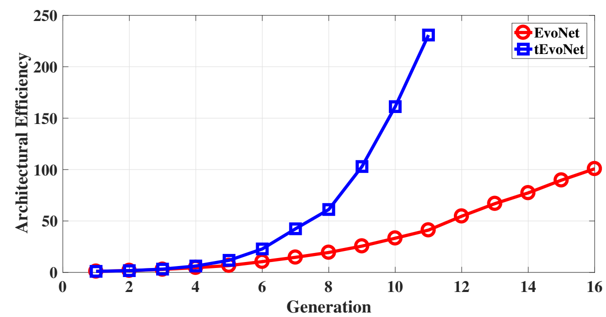

Figure 5. Architectural efficiency over the generation for the SVHN dataset, with a stopping criteria of a performance drop of 3% in modeling accuracy. tEvoNet is compared against EvoNet [22], and the comparison shows that tEvoNet synthesizes offspring deep neural networks with a 230-fold increase in architectural efficiency with a $\sim$3% drop in accuracy after 11 generations, while EvoNet synthesizes a network with 100-fold architectural efficiency at similar accuracy after 16 generations.

can synthesize an offspring deep neural network with a 27-fold increase of architectural efficiency after 11 generations only.

Figure 4 shows the modeling accuracy of offspring deep neural network over generations on the CIFAR10 dataset for the tEvoNet and EvoNet frameworks. It can be observed that the offspring deep neural networks synthesized by both the proposed tEvoNet framework and EvoNet framework can largely preserve modeling accuracy over multiple generations. These results demonstrate that the proposed tEvoNet framework synthesizes offspring deep neural networks with significantly greater architectural efficiency than the EvoNet framework and achieves a higher architectural efficiency at earlier generations.

3.4. Experiment 2: SVHN

For the second experiment, the proposed tEvoNet framework is evaluated against the EvoNet framework on the SVHN dataset. Figure 3 shows a comparison of the architectural efficiency over generations on the SVHN dataset for both the proposed tEvoNet framework and the EvoNet framework. It can be observed that tEvoNet synthesizes an offspring deep neural network with a 230-fold increase of architectural efficiency after 11 generations, while the EvoNet synthesizes an offspring deep neural network with a 100-fold increase in architectural efficiency after 16 generations only. In addition, Figure 6 illustrates the modeling accuracy of offspring deep neural network over generations on the SVHN dataset for the tEvoNet and EvoNet frameworks. The offspring deep neural networks synthesized by both the proposed tEvoNet and EvoNet framework can largely preserve modeling accuracy over multiple generations. These results further reinforce the fact that tEvoNet synthesizes offspring deep neural networks with significantly greater architectural efficiency than the EvoNet and achieves a higher architectural efficiency at an earlier generation.

The experimental results showed the efficacy of the intra-generational evolution in the evolutionary deep intelligence framework which optimizes the genetic encoding scheme to synthesize more efficient network architecture in fewer generations. This approach can have a significant effect to form a new deep neural network architectures which are very efficient yet preserve the modeling accuracy. As mentioned before, the deep neural network architectures can be considered as effective feature extraction framework which are trained end-to-end. By using the proposed tEvoNet framework, it is possible to generate very compact feature extractors that are very efficient yet effective in modeling frameworks.

4. Conclusion

The design of efficient deep neural network architectures for the purpose of learning efficient deep feature representations is a vital factor when dealing with low-cost, low-energy computing devices such as embedded CPUs or even embedded GPUs. In this paper, we extended upon the evolutionary deep intelligence framework by incorporating the notion of learning efficient deep feature representations via the transgenerational genetic transmission of environmental information during the evolutionary synthesis of deep neural networks, which we refer to as tEvoNet. Intra-generational environmental traumatic stresses were imposed to synapses during training in a way that results in more optimized genetical encodings that favor the synthesis of more efficient deep neural networks, thus transmitting the environmental information experienced by a deep

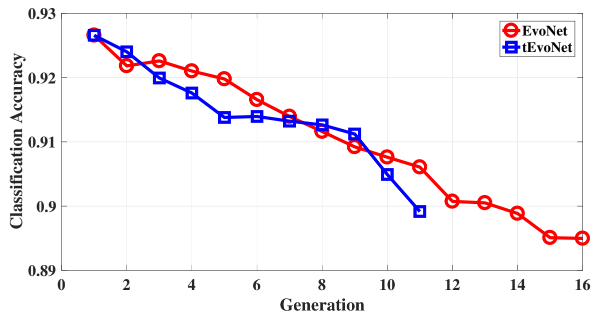

Figure 6. The modeling accuracy over the generation for SVHN, with a stopping criteria of a 3% drop in modeling accuracy. It can be observed that the offspring deep neural networks synthesized by both the proposed tEvoNet and EvoNet can largely preserve modeling accuracy over multiple generations, although tEvoNet synthesizes offspring deep neural networks with significantly greater architectural efficiency than the EvoNet and achieves a larger architectural efficiency at an earlier generation.

neural network from generation to generation. Experimental results performed on the CIFAR10 and SVHN datasets demonstrated the efficacy of the proposed tEvoNet framework, as it was shown that it was capable of synthesizing new network architectures with significant decreases in synapses compared to the original deep neural network architectures (e.g., for SVHN, a ~230-fold increase in synapse efficiency was achieved with tEvoNet, compared to a 100-fold in synapse efficiency using the original EvoNet) while maintaining modeling accuracy, thus resulting in a significantly more efficient deep feature representation. Future work includes investigating alternative intra-generational environmental stresses, as well as dynamic strategies for adapting the degree of intra-generational environmental stress based on the intrinsic characteristics of the deep neural network.

References

[1] B. Alipanahi, A. Delong, M. T. Weirauch, and B. J. Frey. Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. *Nature Biotechnology*, 2015. 1

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *arXiv preprint arXiv:1511.00561*, 2015. 1

[3] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra. Weight uncertainty in neural network. In *International Conference on Machine Learning*. PMLR. 4

[4] B. Dias and K. Ressler. Parental olfactory experience influences behavior and neural structure in subsequent generations. *Nature Neuroscience*, 2014. 2, 4

[5] Y. Gong, L. Liu, M. Yang, and L. Bourdev. Compressing deep convolutional networks using vector quantization. *arXiv preprint arXiv:1412.6115*, 2014. 2

[6] S. Han, H. Mao, and W. J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *CoRR, abs/1510.00149*, 2015. 2

[7] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, et al. Deep speech: Scaling up end-to-end speech recognition. *CoRR, abs/1412.5567*, 2014. 1

[8] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Process. Mag.*, 2012. 1

[9] Y. Ioannou, D. Robertson, J. Shotton, R. Cipolla, and A. Criminisi. Training cnns with low-rank filters for efficient image classification. *arXiv preprint arXiv:1511.06744*, 2015. 2

[10] M. Jaderberg, A. Vedaldi, and A. Zisserman. Speeding up convolutional neural networks with low rank expansions. *arXiv preprint arXiv:1405.3866*, 2014. 2

[11] A. F. Joseph Redmon. Yolo9000: Better, faster, stronger. In *CVPR*, 2017. 1

[12] A. Klosin, E. Casas, C. Hidalgo-Carcedo, T. Vavouri, and B. Lehner. Transgenerational transmission of environmental information in c. elegans. *Nature Neuroscience*, 2017. 2, 4

[13] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images, 2009. 5

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2012. 1, 5

[15] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 2015. 1

[16] Y. LeCun, J. S. Denker, S. A. Solla, R. E. Howard, and L. D. Jackel. Optimal brain damage. In *Advances in Neural Information Processing Systems (NIPS)*, 1989. 2

[17] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 1

[18] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*. Granada, Spain, 2011. 5

[19] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015. 1

[20] M. J. Shafiee, E. Barshan, and A. Wong. Evolution in groups: A deeper look at synaptic cluster driven evolution of deep neural networks. *arXiv preprint arXiv:1704.02081*, 2017. 3

[21] M. J. Shafiee, A. Mishra, and A. Wong. Deep learning with Darwin: Evolutionary synthesis of deep neural networks. *arXiv preprint arXiv:1606.04393*, 2016. 2, 3

[22] M. J. Shafiee and A. Wong. Evolutionary synthesis of deep neural networks via synaptic cluster-driven genetic encoding. In *NIPS Workshop on Efficient Methods for Deep Neural Networks*, 2016. 2, 3, 5, 6

[23] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[24] K. O. Stanley and R. Miikkulainen. Evolving neural networks through augmenting topologies. *Evolutionary computation*, 2002. 2

[25] W. Wen, C. Wu, Y. Wang, Y. Chen, and H. Li. Learning structured sparsity in deep neural networks. *arXiv preprint arXiv:1608.03665*, 2016. 2

[26] D. White and P. Ligomenides. Gannet: A genetic algorithm for optimizing topology and weights in neural network design. In *International Workshop on Artificial Neural Networks*. Springer, 1993. 2