Papers Intensively Reading Selection

Genetic CNN

Lingxi Xie, Alan Yuille
Department of Computer Science, The Johns Hopkins University, Baltimore, MD, USA
198808xc@gmail.com alan.l.yuille@gmail.com

摘要

Authors

本文的第一作者是谢凌曦 (Lingxi Xie)，目前在华为担任高级研究员。他的主要研究方向包括计算机视觉, 图像识别, 多媒体信息检索和机器学习。目前, 主攻 CNN 的自动训练问题 (Automatically machine learning and its application)。他分别在 2010 年和 2015 年在清华大学工学院获得 B.E 和 Phd degree, 之后又前往 UCLA 的CCVL(Compositional Cognition, Vision, and Learning) 实验室从事计算机视觉的研究。该实验室在 2017 年搬去了 Johns Hopkins University(JHU)。本篇论文就是他在 JHU 担任博士后时的研究成果。

以下是从他个人主页截取的片段。

(I am currently a senior researcher at Huawei Inc. My research interests include Computer Vision, Multimedia Information Retrieval and Machine Learning. I have a lot of research work on generic image classification, image retrieval and other vision problems. Recently, I have been working on deep learning, especially the models based on the Convolutional Neural Network (CNN). Recently, I am particularly interested in automated machine learning and its applications. Before I joined Huawei, I obtained my B.E. and Ph.D. degrees from Tsinghua University, under the supervision of Prof. Bo Zhang and Prof. Qi Tian from University of Texas at San Antonio (UTSA). I also served as a research intern at Microsoft Research Asia (MSRA), under the supervision of Dr. Jingdong Wang. After gaining my Ph.D. degree, I spent three years as a post-doctoral researcher at the Computational Cognition, Vision, and Learning (CCVL) under the supervision of Prof. Alan L. Yuille. Our lab was located at the University of California, Los Angeles (UCLA), and then moved to the Johns Hopkins University (JHU).)

Purpose and Methods

本篇文章主要讨论一种自动学习深的神经网络结构的一种可能性。这个 idea 的主要来源于随着网络 layers 的变多，可能的网络结构数量会呈指数增长，在如此丰富的网络结构中，是否有一个结构能够优化训练过程呢。这篇论文就利用**遗传算法**的思想来探索深的神经网络的优化网络结构。

本文的核心思想是提出了一种对神经网络结构的编码方案，这种编码方案可以用一个**定长**的二进制数组来代表。遗传算法初始化为随机的一组个体。在每次进化中，发生选择、突变和交叉，从而产生个体的多样性参与竞争, 淘汰掉那些性能较差的个体。这里的可适应性定义为个体的识别准确度 (recognition accuracy)，这可以通过在参考数据集上的标准训练过程得到。

Conclusion

该论文将设计的遗传算法用在 CIFAR10 上评估, CIFAR10 是一个比较小的数据集。结果表明, 遗传算法可以在正式的大规模训练前找到一个较优的网络结构。得到的优化网络也被引入到了 ILSVRC2012 数据集中，用来训练大尺寸的数据集，得到的结果比一些手工设计的 CNN SOTA 要好。

1 导言

CNN 是当今常用的深度学习模型。在进行一些需要挖掘抽象语义的应用时，往往需要更深的网络 (Deep CNN)，但是，深网络往往会带来梯度消失，过拟合或欠拟合的问题，从而导致识别精度和速度的下降。研究者们提出了许多创新性的网络结构。比如 2012 年，著名的 AlexNei 被提出，极大地优化了图像识别领域的训练性能。2016 年，时任 MSRA 高级研究员的何凯明提出了著名的残差卷积神经网络 (ResNet), 该论文依然是目前 CV 领域引用量最高的论文。作者认为，尽管这些网络非常优美和高效，但是他们都是手工做出来的，缺乏一定的灵活性，在一些特定问题求解时不一定高效。于是提出了，能否能利用神经网络“自己学习”从而找到一个较优的网络结构，提高灵活度。

本文考虑了一种 stages 数受限制的情况, 文章将这个问题设置为一个在大搜索空间的优化问题, 利用遗传算法在问题的可能解空间进行高效搜索。

本文提出了一种编码的方案，这种编码方案可以用一个**定长**的二进制数组来代表每个网络结构：遗传算法初始化为随机的一组个体。在每次进化中，发生选择、突变和交叉，从而产生个体的多样性参与竞争，淘汰掉那些性能较弱的个体。这里的适应性函数 (fitness function) 为个体的识别准确度 (recognition accuracy)，这可以通过在参考数据集上的标准训练过程得到。在最后，对遗传算法找出的网络结构在大数据集上完整地进行训练过程 (包括 fwd 和 BP 算法)，这一步是独立于遗传算法的。遗传算法在定长二进制字符串确定后就已经结束了。

文章的整体思路其实很明晰:

1. 用遗传算法得到 top-ranked 的 NN structure。因为遗传算法中计算 fitness function 是一件非常耗时的事，因此每次遗传进化的过程中，都用 small dataset (CIFAR10) 进行测试。通过在 CIFAR10 上的测试，确定高效的网络结构
2. 用上面找出的高效网络结构尽心较大数据规模的数据集测试

经过学习筛选后的网络结构，相对于学习前的手工设计网络，大部分都具有更好的性能，无论是小数据集还是大数据集上。

2 相关工作

文章的 related work 介绍了 CNN 和 genetic algorithm。

2.1 CNN

文章引用了一些经典的 CV 领域的论文，包括 ImageNet，AlexNet 等。文章指出，CNN 作为一种针对大范围数据集的启发式模型，其性能主要是基于神经元的数目和网络的深度的。近年来，大数据训练的可行性和以 NVIDIA GPU 为首的深度学习硬件资源的出现，让训练深的神经网络成为可能 (ImageNet)。通常来说，更深的神经网络可以得到更好的识别或分类结果，但同时，加入总线信息也被证明是有用的 (adding highway information has been verified to be useful)。文章还表示，一些使用随机的或是 dense 的网络结构也可以得到好的结果，但是他们也是人为确定好的，限制了作为网络结构设计的灵活性。

2.2 Genetic Algorithm

GA 常用于找到一些优化问题、搜索问题的较优解。

一个标准的 GA 需要两个前提条件：一是对于问题解空间的基因表达，换句话来说就是需要定义个体基因的编码方案，这种编码方案就是待优化问题的解空间。二是需要一个 fitness function(适应度函数)。

重新标准化 GA 的流程：

1. 定义个体的基因编码方案
2. 初始化种群
3. 用 fitness function 衡量个体生存竞争能力的适应度
4. 淘汰适应度低的个体，选择适应度高的个体构成种群下一代的成员（选择）
5. 按一定概率对下一代成员进行基因的交叉与变异（交叉与变异），产生新个体的基因编码方案
6. 评估新种群的适应度

研究者们提出了许多优化的 GA，包括 performing local search, generating random keys。

作者在这里表明了本文和前人工作的区别：We also note that some previous work applied the genetic algorithm to learning the structure or weights of artificial neural networks, but our work aims at learning the architecture of modern CNNs, which is not studied in prior researches.

3 方法

3.1 Binary Network Representation

考虑这些网络结构：可以分为几个 stages，在每个 stage 中，所有 data cube 的几何维度 (width, height, depth) 都不会发生变化。相邻的 stage 通过 pooling(池化层) 连接。同一个 stage 的所有卷积操作都具有相同数量的 filter。

文章得出了如下的神经网络架构的基因编码方案：

1. 假设每个网络结构由 S 个 stage 组成
2. 假设第 s 个 stage 包含 K_s 个节点，表示为 $V_{s,k_s}, k_s = 1, 2, ..., K_s$，每一个 stage 的 nodes 都是排好序的，这 K_s 个节点，都包括了卷积 +batch normalization(正则化)+ReLU 的操作，这样的操作逻辑被证实在训练非常深的 CNN 时非常有效。(注：不对全连接网络进行编码)
3. 在每个 stage，用 $1 + 2 + ... + (K_s - 1) = \frac{1}{2}K_s(K_s - 1)$ 个比特来编码表示节点之间的连接关系，第一个比特用来表示 $(v_{s,1}, v_{s,2})$ 是否相连，第 2 和第 3 个比特用来表示 $(v_{s,1}, v_{s,3})$ 和 $(v_{s,2}, v_{s,3})$ 是否相连, 以此类推. 编码结束于最后 $K_s - 1$ 个比特用来表示 $v_{s,1}, v_{s,2}, ..., v_{s,K_s-1}$ 与 v_{s,K_s} 的连接关系。如果比特的值为 1，表示对应的两个节点相连。有边连接表示前一节点输出的特征图会作为后一连接节点的输入，可能会有多个边指向同一个节点，此时 element-wise 相加，此时是一个残差结构（由于位于一个阶段内部，因此特征图的 channel 数一致，只需要 padding 即可相加）

一个具体的例子如下:

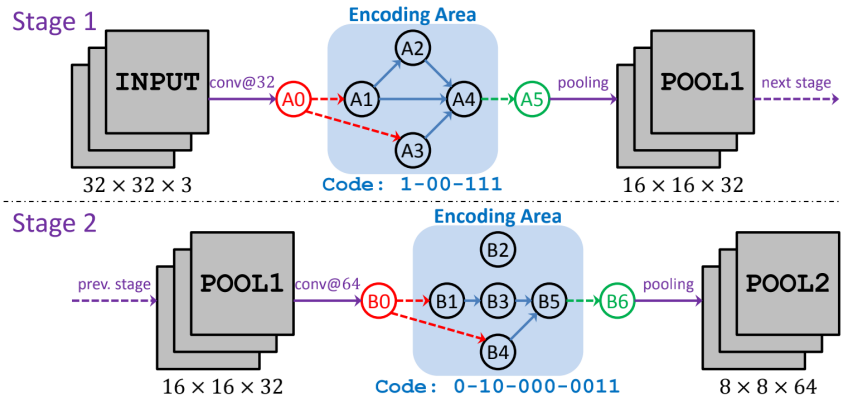

Figure 1. A two-stage network ($S = 2$, $(K_1, K_2) = (4, 5)$) and the encoded binary string (best viewed in color). The default input and output nodes (see Section 3.1.1) and the connections related to these nodes are marked in red and green, respectively. We only encode the connections between the ordinary codes (regions with light blue background). Within each stage, the number of convolutional filters is a constant (32 in Stage 1, 64 in Stage 2), and the spatial resolution remains unchanged (32×32 in Stage 1, 16×16 in Stage 2). Each pooling layer down-samples the data by a factor of 2. ReLU and batch normalization are added after each convolution.

因此，对于一个有 S 个 stages 的 CNN 来说，总共的 encoder 长度 $L=\frac{1}{2}\sum_s K_s(K_s-1)$，因此解空间的大小为 2^L！这是一个指数复杂度的问题，如果想遍历所有的解找到最优解显然不可能的。原文举出了一个例子。因此作者采用了 GA 来进行优化。

3.2 Technical Details

为了让每一个编码均有效，论文在每个阶段中额外定义了两个默认节点 (default nodes)，对于第 s 个阶段来说，这两个节点分别为 $V_{s,0}, V_{s,K_s+1}$，分别为上图中的红色和绿色节点。红色节点接收前一节点的输出，对其进行卷积，并将输出的特征图送往所有没有前驱的节点，而绿色节点接受所有有前驱节点输出的特征图，对其进行 element-wise 相加后，输入到池化层。注意：默认节点不尽兴前面的编码操作。

文章指出了两种特殊情况:

1. 如果一个节点不与任何节点连接，则 simply ignore it。这是有道理，一个包含更多节点的网络的所有结构肯定包含他的一部分网络的结构。
2. 如果在一个 stage 没有任何连接 (encode 为 0)，卷积操作只会进行 1 次

一些著名的网络结构的编码例子如下:

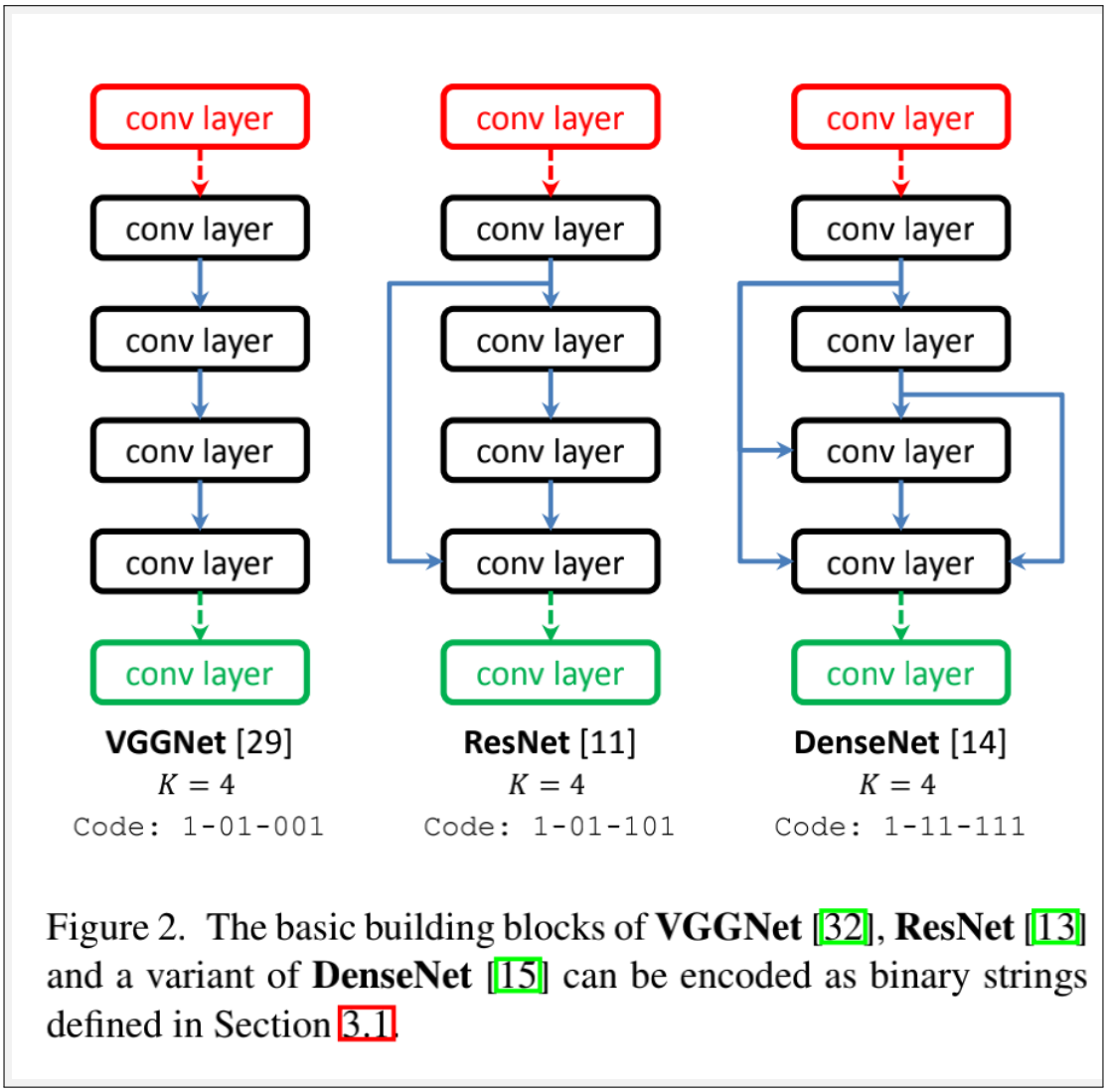

Figure 2. The basic building blocks of **VGGNet** [32], **ResNet** [13] and a variant of **DenseNet** [15] can be encoded as binary strings defined in Section 3.1.

不难验证，这是对的。

3.3 Genetic Operations

3.3.1 Initialization

初始化一个随机模型集合 $\{\mathbb{M}_{0,n}\}_{n=1}^{N}$，集合里每个模型都是一个长度为 L 的二进制串，每个二进制串都是独立地服从伯努利分布: $b_{0,n}^{l}\ B(0.5), l = 1, 2, ..., L$。

作者发现初始化的方式似乎不太影响 GA 的性能，就算用一个 naive approach，ie 全为 0，GA 也能找出一个不错的优化解。

3.3.2 Selection

选择在每一次进化开始时进行，更准确的表述是:

第 t 次进化前，第 n 个随机模型 $\mathbb{M}_{t-1,n}$ 会通过 fitness function 进行评估得到适应度 $r_{t-1,n}$，这一定程度上代表了该模型在这次进化中存活的概率。

文章采用 Russian roulette(轮盘赌) 决定模型的存活。根据文章的思想，fitness function 设置为该模型在 CIFAR 上的 recognition accuracy。

3.3.3 Mutation and Crossover

变异操作的单位是模型编码的每一位，即比特翻转 $\mathbb{M}_{t,n}$ 发生变异的概率为 p_M，其中每一位比特发生翻转的概率为 q_M，作者表示，q_M 一半很小。

交叉操作的单位是每个 stage，这是因为保留网络的结构 layer 不变，只改变 order，有时候也能得到好的结果。发生交叉的概率也很小，为 q_C

3.3.4 Evaluation

对进化后的模型集合中每个模型重新训练，对于旧模型，采用其历史准确率的平均值作为适应度；对于新模型，采用其准确率作为适应度。这么做可以一定程度上对抗训练噪声，即有些模型之所以有较高的准确率，是因为“幸运”，而不是这类模型的架构优秀

算法的 pseudo code 如下：

Algorithm 1 The Genetic Process for Network Design

Input: the reference dataset $\mathcal{D}$, the number of generations T, the number of individuals in each generation N, the mutation and crossover probabilities p_{M} and p_{C}, the mutation parameter q_{M}, and the crossover parameter q_{C}.
Initialization: generating a set of randomized individuals $\{\mathbb{M}_{0,n}\}_{n=1}^{N}$, and computing their recognition accuracies;
for $t = 1, 2, \ldots, T$ **do**
  Selection: producing a new generation $\{\mathbb{M}'_{t,n}\}_{n=1}^{N}$ with a Russian roulette process on $\{\mathbb{M}_{t-1,n}\}_{n=1}^{N}$;
  Crossover: for each pair $\{(\mathbb{M}_{t,2n-1}, \mathbb{M}_{t,2n})\}_{n=1}^{\lfloor N/2 \rfloor}$, performing crossover with probability p_{C} and parameter q_{C};
  Mutation: for each non-crossover individual $\{\mathbb{M}_{t,n}\}_{n=1}^{N}$, doing mutation with probability p_{M} and parameter q_{M};
  Evaluation: computing the recognition accuracy for each new individual $\{\mathbb{M}_{t,n}\}_{n=1}^{N}$;
end for
Output: a set of individuals in the final generation $\{\mathbb{M}_{T,n}\}_{n=1}^{N}$ with their recognition accuracies.

4 实验

作者表示，和其他训练神经网络以至于得到一个较好的结构的方法一样，本文的 GA 也非常消耗计算资源，因此只能在一些小数据集上进行训练和探索用于评估架构性能，再将探索到的网络结构应用在大数据集上，比较和一些 manual design CNN 的性能差距。

4.1 CIFAR10

作者先用一个简单的 CNN：LeNet 进行实验，LeNet 是一个包含 3 层卷积层，3 层最大池化层和 1 层全连接层的神经网络，dropout ratio 为 0.5(用来避免 over-fitting)，120 轮下学习率为 0.01，60 轮 $\alpha = 0.001$，40 epochs $\alpha = 0.0001$, 20 epochs $\alpha = 0.00001$。

GA 时，设置 S=3，每一层的节点数为 (3,4,5)，因此 L=3 + 6 + 10 = 19，解空间大小为 524288，初始化 20 个模型，进化次数为 50 轮，$p_M = 0.8, q_M = 0.05, p_C = 0.2, q_C = 0.2$，GA 算法在 GPU 上运行了 2 天，作为对比，完全遍历解空间找最优解的工作量至少是 100 倍。

实验并行进行了两次 GA 算法，得到的实验数据如下：

Gen	Max %	Min %	Avg %	Med %	Std-D	Best Network Structure
00	75.96	71.81	74.39	74.53	0.91	0-01|0-01-111|0-11-010-0111
01	75.96	73.93	75.01	75.17	0.57	0-01|0-01-111|0-11-010-0111
02	75.96	73.95	75.32	75.48	0.57	0-01|0-01-111|0-11-010-0111
03	76.06	73.47	75.37	75.62	0.70	1-01|0-01-111|0-11-010-0111
05	76.24	72.60	75.32	75.65	0.89	1-01|0-01-111|0-11-010-0011
08	76.59	74.75	75.77	75.86	0.53	1-01|0-01-111|0-11-010-1011
10	76.72	73.92	75.68	75.80	0.88	1-01|0-01-110|0-11-111-0001
20	76.83	74.91	76.45	76.79	0.61	1-01|1-01-110|0-11-111-0001
30	76.95	74.38	76.42	76.53	0.46	1-01|0-01-100|0-11-111-0001
50	**77.19**	**75.34**	**76.58**	**76.81**	0.55	1-01|0-01-100|0-11-101-0001

Table 1. Recognition accuracy (%) on the **CIFAR10** testing set. The zeroth generation is the initial population. We set $S = 3$ and $(K_1, K_2, K_3) = (3, 4, 5)$. The best individual in each generation is also shown in binary codes.

实验结果表明，随着 GA 的轮数增加，error rate 是在减小的，最后可以从 24.04% 降到 22.81%

作者还讨论了不同的初始化对于 GA 的影响。分为 random initialization 和 All-zero naive initialization，得到的结果图如下：

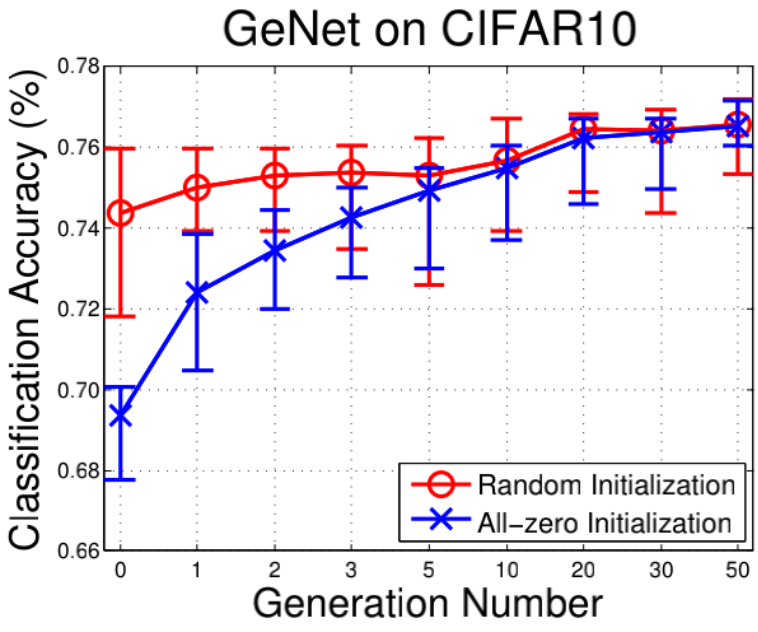

Figure 3. The average recognition accuracy over all individuals with respect to the generation number. The bars indicate the highest and lowest accuracies in the corresponding generation.

作者还讨论了 better model 进行 GA 后会不会更有可能得到 better model 的情况，因为与 GA 本身无关，这里不做讨论了。

两个可视化网络结构如下：

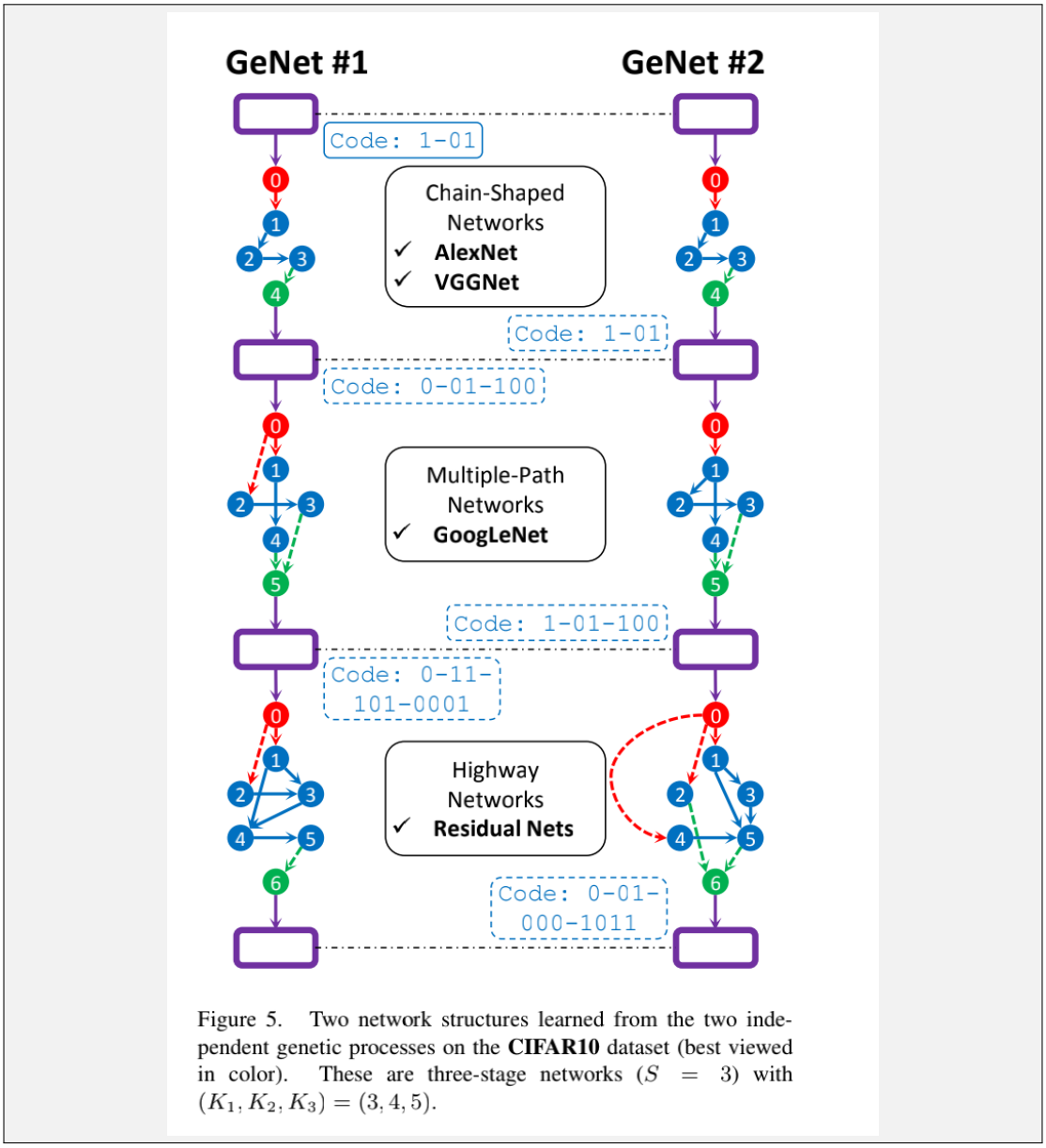

Figure 5. Two network structures learned from the two independent genetic processes on the **CIFAR10** dataset (best viewed in color). These are three-stage networks ($S = 3$) with $(K_1, K_2, K_3) = (3, 4, 5)$.

作者用上面的两个模型，GeNet1 和 GeNet2 与 SOTA 的 manual 结构在一些 small-scale dataset 和 large-scale dataset 上进行了对比，结果如下：

	SVHN	CF10	CF100
Zeiler *et.al* [43]	2.80	15.13	42.51
Goodfellow *et.al* [10]	2.47	9.38	38.57
Lin *et.al* [26]	2.35	8.81	35.68
Lee *et.al* [24]	1.92	7.97	34.57
Liang *et.al* [25]	1.77	7.09	31.75
Lee *et.al* [23]	1.69	6.05	32.37
Zagoruyko *et.al* [42]	1.77	5.54	25.52
Xie *et.al* [39]	1.67	5.31	25.01
Huang *et.al* [16]	1.75	5.25	24.98
Huang *et.al* [15]	**1.59**	**3.74**	**19.25**
GeNet after G-00	2.25	8.18	31.46
GeNet after G-05	2.15	7.67	30.17
GeNet after G-20	2.05	7.36	29.63
GeNet #1 (G-50)	1.99	7.19	**29.03**
GeNet #2 (G-50)	**1.97**	**7.10**	29.05
GeNet from WRN [42]	**1.71**	**5.39**	**25.12**

Table 2. Comparison of the recognition error rate (%) with the state-of-the-arts. We apply data augmentation on all these datasets. **GeNet** #1 and **GeNet** #2 are the structures shown in Figure 5.

	Top-1	Top-5	# Paras
AlexNet [19]	42.6	19.6	62M
GoogLeNet [36]	34.2	12.9	13M
VGGNet-16 [32]	28.5	9.9	138M
VGGNet-19 [32]	28.7	9.9	144M
GeNet #1	28.12	9.95	156M
GeNet #2	**27.87**	**9.74**	156M

Table 3. Top-1 and top-5 recognition error rates (%) on the **ILSVRC2012** dataset. For all competitors, we report the single-model performance without using any complicated data augmentation in *testing*. These numbers are copied from this page: `http://www.vlfeat.org/matconvnet/pretrained/`. **GeNet** #1 and **GeNet** #2 are the structures shown in Figure 5.

总体上来看，经过 GA 的网络结构会有一定的性能提升。

5 个人评价

Xie 的思想其实很朴素，在越来越多奇形怪状的 manual design CNN 的情况下，他表示，可不可以“训练”神经网络的结构，然后得到一个训练出的优化结构。在优化这个问题上选取了遗传算法，并且朴素且巧妙地设计了编码的方式。这篇文章的参考价值还是非常高的。

以下是 Xie 对本文的个人评价：

"Throughout this work, the genetic algorithm is only used to propose new network structures, the parameters and recognition accuracy of each individual are obtained via a standalone training-from-scratch." [1]

参考文献

[1] Lingxi Xie and Alan Yuille. Genetic cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.