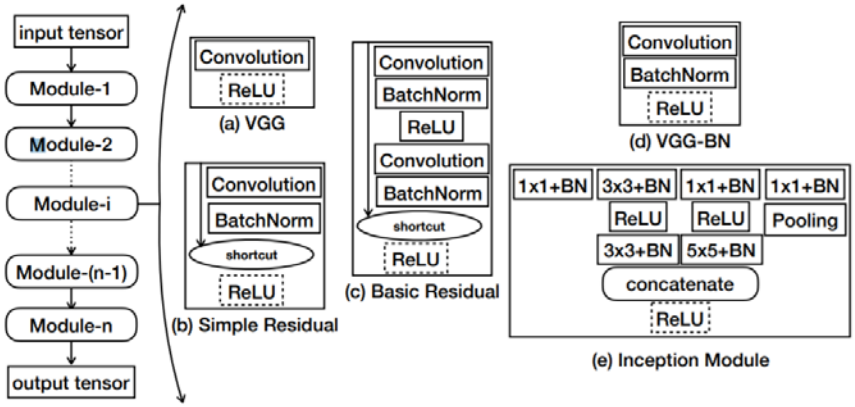
1. ERASERELU: A SIMPLE WAY TO EASE THE TRAINING OF DEEP CONVOLUTION NEURAL NETWORKS

文章的依据：

- 通过堆叠多个非线性模块比较难拟合一个线性映射关系；
- 模块越多，越容易过拟合、越难优化。



这篇文章将每个经典模块的最后一个 ReLU 去除，在 CIFAR10、CIFAR100、ImageNet 上均能得到优于原来网络的结果。

虽然有点效果，但是感觉只是通过减少网络容量，从而降低过拟合风险以及优化难度而已。

2. Can we beat the state of the art from 2013 with only 0.046% of training examples?

2013 年，Kaggle 举办过一个很受欢迎的猫狗识别竞赛（Dogs vs. Cats），比赛内容是识别图像中的是猫还是狗。当时获胜的准确率是 82.7%，使用 13000 张图像进行训练，使用 25000 张图像训练取得 98.914%的准确率。本文作者仅使用 6 张图像作为训练样本，取得 89.97%的准确率。

作者用了迁移学习的方法，在 VGG19 上 finetune 随机挑选出来的 6 张图片，如上图。

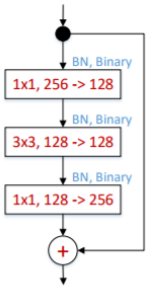
经过 41 epochs 的训练后，该模型达到了 89.97%的准确率，验证集大小是 24994。作者没有用数据增强，也没有调整学习率和正则化就可以达到这样的效果，这意味着要将迁移学习使得深度学习在医学等其他领域的应用也会更加便

捷。

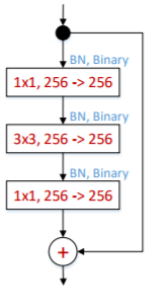
3. Binarized Convolutional Landmark Localizers for Human Pose Estimation and Face Alignment with Limited Resources

这篇文章讲述了如何为定位网络设计一个二值网络：

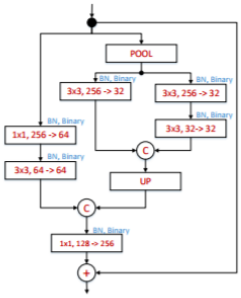
- 第一层和最后一层用真值，其余的层用二值表示；
- 加宽网络宽度可以提升网络性能，如下图 a)到 b)；
- 多尺度卷积核更加适合二值网络，如下图 a)到 c)；
- 1*1 的卷积核不适合二值网络，应该去掉，如下图 a)到 d)；
- 用下图 e)这种多层次多尺度的模块相对于图 a)可以在不增加额外的参数的情况下提升网络性能。



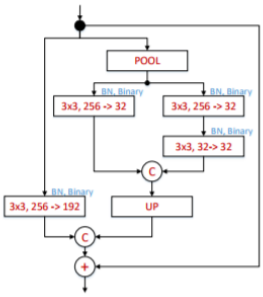
(a) The **Original Bottleneck** block with pre-activation, as defined in [11]. Its binarized version is described in Section 4.1.



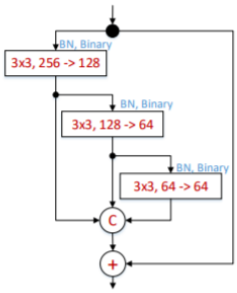
(b) The **Wider** version of (a) produced by increasing the number of filters in the second layer. See Subsection 4.2.



(c) Largely departing from (b), this block consists of **Multi-Scale (MS)** filters for analyzing the input at multiple scales. See Subsection 4.3.



(d) A variant of the MS block introduced in (c) after removing all convolutional layers with 1×1 filters (**MS Without 1×1 filters**). See Subsection 4.3.



(e) The proposed **Hierarchical, Parallel & MS** (denoted in the paper as **Ours, final**) block incorporates all ideas from (b), (c) and (d) with an improved gradient flow. See Subsection 4.5

4. All You Need is Beyond a Good Init: Exploring Better Solution for Training Extremely Deep Convolutional Neural Networks with Orthonormality and Modulation

今年 CVPR 的一篇论文，提出了正交的正则化方法，可以在不使用 residual 模块和 inception 模块的情况训练很深的网络。

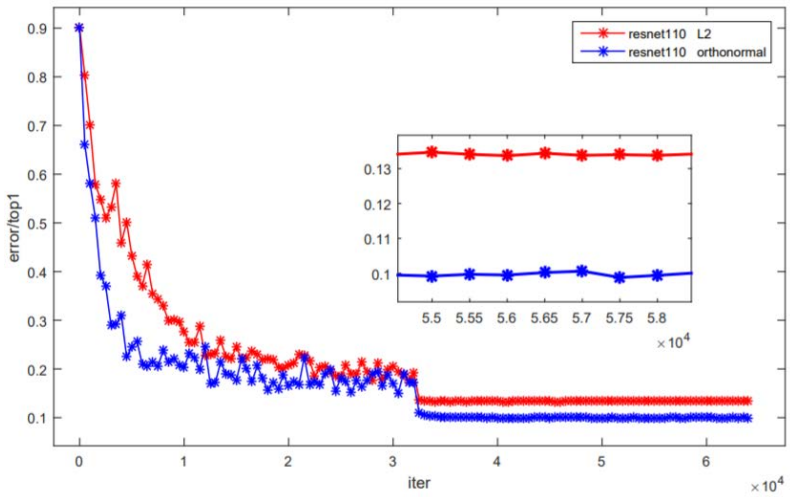
$$\|\mathbf{y}\| = \sqrt{\mathbf{y}^T\mathbf{y}} = \sqrt{\mathbf{x}^T\mathbf{W}\mathbf{W}^T\mathbf{x}} = \sqrt{\mathbf{x}^T\mathbf{x}} = \|\mathbf{x}\| \; if f. \; \mathbf{W}\mathbf{W}^T = \mathbf{I}$$

在正交的约束下，可以使得输入与输出的 norm 大小得到保持，在一定程度上能减轻梯度消失或者梯度爆炸的现象。只需在 loss 函数后面加上下列正则项

即可。

$$\frac{\lambda}{2}\sum_{i=1}^{D}\|\mathbf{W}_l^T\mathbf{W}_l-\mathbf{I}\|_F^2$$

加了正交正则项可以提升网络性能，以下图为例，可以提升 resnet110 的性能。



5. UNDERSTANDING DEEP LEARNING REQUIRES RETHINKING GENERALIZATION

这是 ICLR2017 比较有争议的一篇 best paper。作者从泛化性的角度来对深度学习展开剖析。

1）作者将 ground-true label 全部换成 random label，网络可以收敛到 0 的训练误差，即使把输入图片的像素也噪声化，网络依然可以很好得拟合噪声图片与随机 label，并达到 0 的训练误差。当然，测试误差当然不会比随机猜的要好。这说明了，深度神经网络可以暴力地记住所有的训练数据，甚至是噪声。

2）显式正则在深度学习中的角色（Explicit regularization may improve generalization performance, but is neither necessary nor by itself sufficient for controlling generalization error）：显式正则例如 L2 正则、dropout、数据增强等或许可以降低测试误差，但并不意味着它是控制泛化误差的保证。这里我更加同意知乎上一个人的说法，显示正则的目的并不是提升泛化性，而更加像是作为一种数据的先验分布作用在模型上（从贝叶斯的角度上来看），如果这种先验分布是错误，显然并不会提升网络性能。

3）有限的样本表达：足够大的神经网络可以表达任意标签的训练数据。文章作者以一个包含 2n+d 个参数的两层 ReLu 网络来拟合 n 个 d 维样本带任何标签。个人觉得这只是一个理论上界的值，作者并没有讨论要达到这个上界所要的条

件。。。

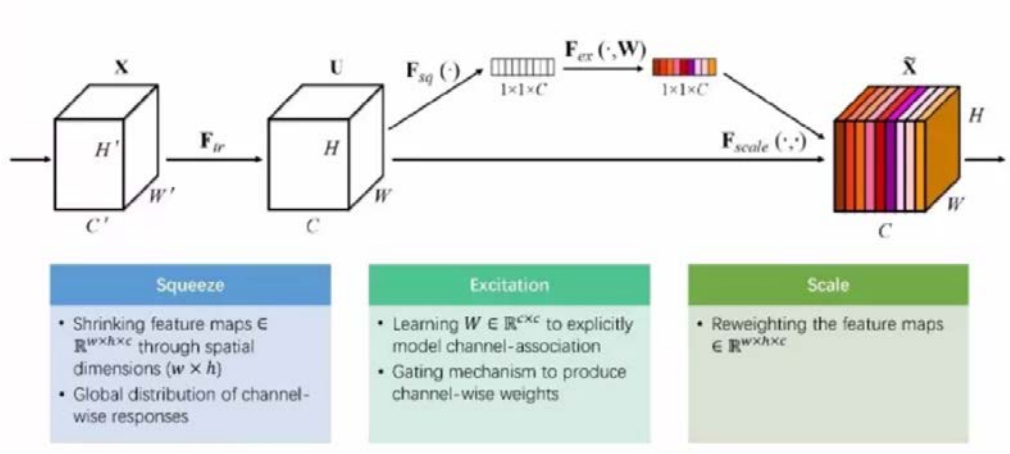
4）隐式正则在深度学习泛化过程中的角色：文章举了个线性模型的 SGD 优化为例，指出 SGD 其实也是一个隐式的正则器，用 SGD 来训练一些小的数据集，即使没有加正则项也可以泛化得很好。

6. Estimated Depth Map Helps Image Classification

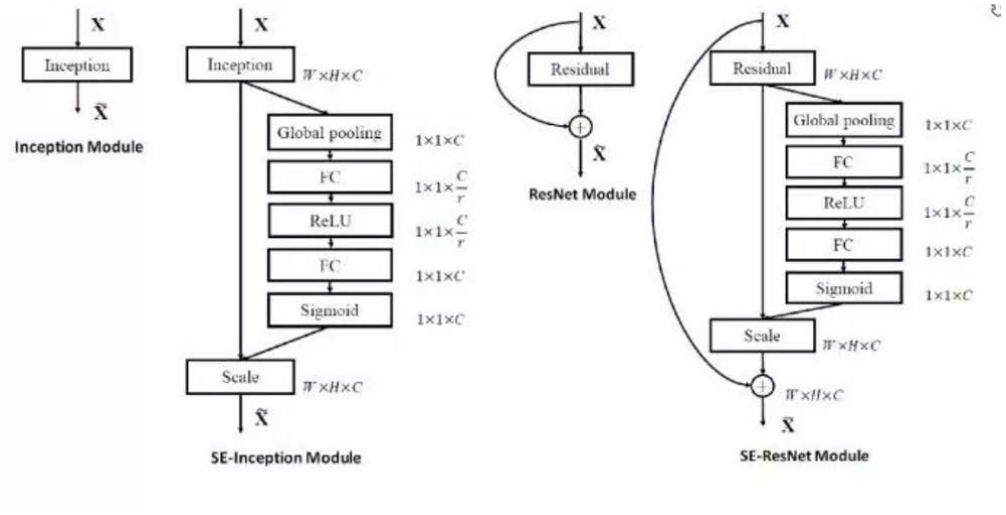
这篇文章涉及到迁移学习领域，讲的是在其他带深度信息标签的数据集训练得到的生成深度图的模型，能为另一个不带深度信息的分类数据集生成深度图，进一步投入训练，能提升该分类模型的精度。文章在 CIFAR10 做了一个实验，在 Resnet-20 上和 Resnet-56 上分别能提升 0.55%和 0.53%的分类精度。

7. Squeeze-and-Excitation Networks

这是 Face++在今年 ImageNet Classification 上获得第一的方法。



整套方法也是比较简单理解的，分别用到了两个操作，Squeeze 和 Excitation 两个操作。其中 Squeeze 将 C*H*W 的 feature map 压缩为 C*1*1，提取每个 channel 的 feature map 的 global 信息，而 Excitation 则用一个 C*C 的矩阵来提取每个 channel 之间的联系，最后用这个 C*1*1 的 weight 对 Squeeze 之前的 feature map 进行 reweighting，进而 enhance 有用的 features，抑制无用的 features。下图为 SE 模块在 inception 模块和 resnet 模块上的应用：



SE-Resnet50 能抵得上 Resnet101 的效果，SE-Resnet101 能抵得上 Resnet152 的效果。

8. COUPLED ENSEMBLES OF NEURAL NETWORKS

这篇文章提出了一种 multi-branches 的结构，文中以 DenseNet 和 ResNet 为基本的 single-branch，在 CIFAR10、CIFAR100 和 SVHN 数据集上做实验，得到了一下结论：

- 在一定参数量的限制下，将一个大网络（single-branch）分为多个平行的小网络（multi-branches）联合训练要得到更好的结果；
- 在 fuse 多个小网络的结果这一步里，fusion 的操作应该放在 Softmax 或者 Log-Likelihood 这些层的上面要比放在这些层的后面要得到更好的结果，fusion 的操作一般可以用 average 等；
- 这个 multi-branches 的结果可以进一步 ensemble 得到更好的结果。

9. Feature Pyramid Networks for Object Detection

由于像 Faster-RCNN 这些网络做物体检测都是在最后一层 feature map 上做的，很容易对小物体漏检，像 SSD 则是在多层 feature map 上做预测的，但每一层 feature map 的语义是不同的。FAIR Lab 提出一种 Feature Pyramid Network，这是一种 top-down 的网络结构，融合多尺度信息，并对每一层做出预测。

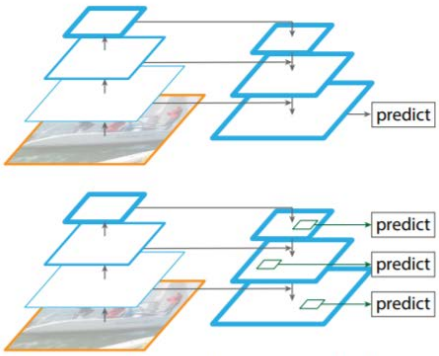


Figure 2. Top: a top-down architecture with skip connections, where predictions are made on the finest level (*e.g.*, [28]). Bottom: our model that has a similar structure but leverages it as a *feature pyramid*, with predictions made independently at all levels.

10.Mask R-CNN

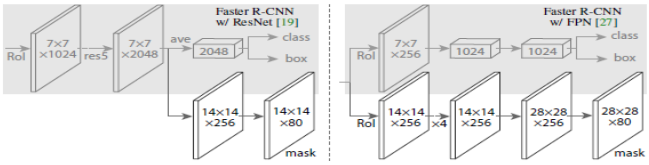


Figure 3. **Head Architecture**: We extend two existing Faster R-CNN heads [19, 27]. Left/Right panels show the heads for the ResNet C4 and FPN backbones, from [19] and [27], respectively, to which a mask branch is added. Numbers denote spatial resolution and channels. Arrows denote either conv, deconv, or *fc* layers as can be inferred from context (conv preserves spatial dimension while deconv increases it). All convs are 3×3, except the output conv which is 1×1, deconvs are 2×2 with stride 2, and we use ReLU [30] in hidden layers. *Left*: 'res5' denotes ResNet's fifth stage, which for simplicity we altered so that the first conv operates on a 7×7 RoI with stride 1 (instead of 14×14 / stride 2 as in [19]). *Right*: '×4' denotes a stack of four consecutive convs.

Mask-RCNN 其实就是在 FPN 网络的后面加了个 instance segmentation 网络而已，但是可以达到很好的效果。

11. Focal Loss for Dense Object Detection

这是 FAIR Lab 何凯明等人对 Dense Object Detection 提出的一种新的 loss。

目前高精度的物体检测基本上都是一种 two-stage 的方法（即一个 object location proposals 后面接一个分类器），one-stage（例如 SSD 或者 YOLO）的方法虽然要更加快也更加简单，但是其精度还是要稍弱于 two-stage 的方法的。这篇文章对这个现象进行了研究，发现在物体检测中前景和背景的样本量严重不平衡是导致这个问题的主要原因，并提出一种 Focal loss 对这个问题进行解决。作者用 Feature Pyramid Network 设计了一种 one-stage 的物体检测方法 RetinaNet，使其既能达到 two-stage 的精度，同时能达到 one-stage 方法的速

度。

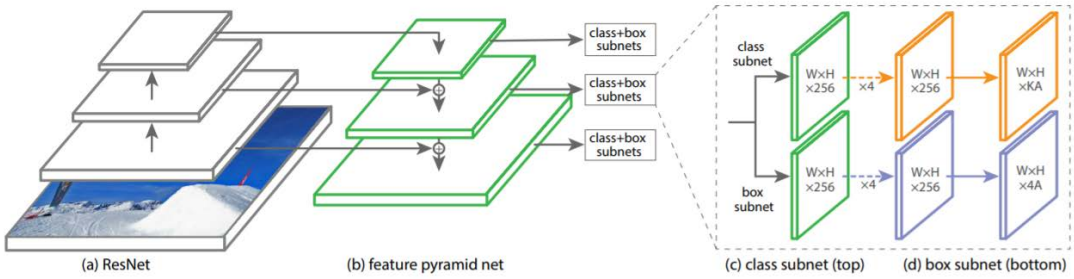


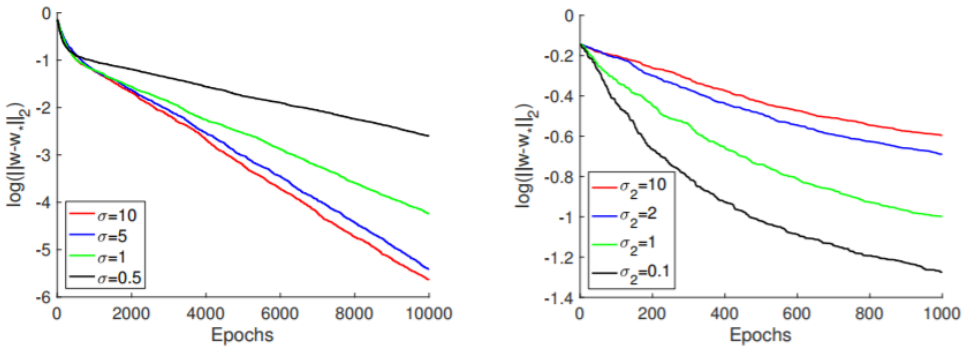
Figure 3. The one-stage **RetinaNet** network architecture uses a Feature Pyramid Network (FPN) [19] backbone on top of a feedforward ResNet architecture [15] (a) to generate a rich, multi-scale convolutional feature pyramid (b). To this backbone RetinaNet attaches two subnetworks, one for classifying anchor boxes (c) and one for regressing from anchor boxes to ground-truth object boxes (d). The network design is intentionally simple, which enables this work to focus on a novel focal loss function that eliminates the accuracy gap between our one-stage detector and state-of-the-art two-stage detectors like Faster R-CNN with FPN [19] while running at faster speeds.

文中的重点主要在于 Focal Loss 上，其实 Focal loss 也是一个比较容易理解的东西，就是 down-weight 比较容易检测出来的物体的 loss,将网络训练的重心移到 hard examples 上，使得大量容易检测出来的 easy negatives 不会再训练的过程中起到压倒性的作用。

12. When is a Convolutional Filter Easy to Learn?

这也是一篇 FAIR Lab 出的比较偏理论的一篇论文，文中的论点是：

- 如果输入的各个滑动窗有很强的相关性，那么网络的训练将会收敛得非常快；
- 如果输入的分布越平滑，那么网络的训练会收敛得越快，高斯分布式一个特例。



(a) Convergence rates of stochastic gradient descent for one-layer one-neuron model on input distributions with different smoothness. Larger σ is smoother.

(b) Convergence rates of stochastic gradient descent for learning a convolution filter on input distributions with different closeness of patches. Larger σ_2 is smoother.

Figure 3: Experiments on synthesized and real data.

13.PREDICTION OF USERS' FACIAL ATTRACTIVENESS ON AN ONLINE DATING WEBSITE

这篇文章讲了如何在约会网站上提取数据进行人脸吸引力打分，同时也探讨

了人脸颜值与人的年龄的关系。

这篇文章的数据集上由约会网站上注册用户的人脸头像收集的，而其人脸吸引力的分数则是通过男女之间的互动情况进行量化，如下图所示，如果男生对女生发起对话，如果女生会回复，则说明男生的颜值高，本文通过统计这个信息对男生、女生的颜值信息进行统计。

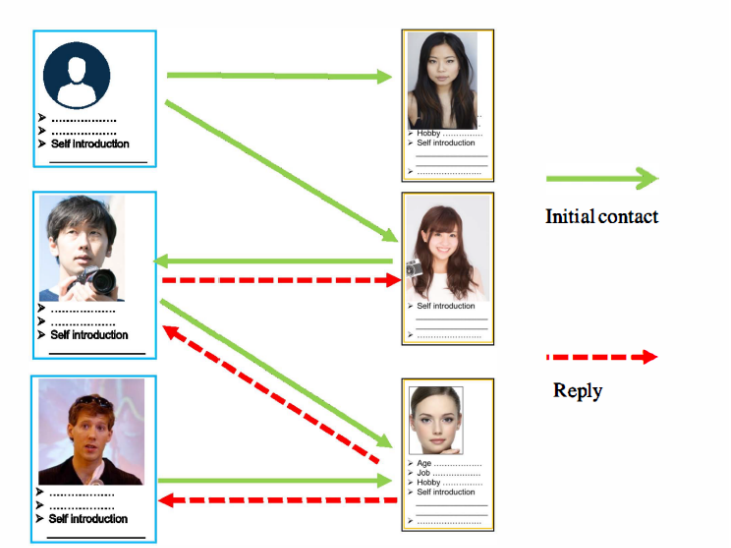


Fig. 1. Illustration of interactions between online dating users. Users contact those whose profiles fit their tastes and people contacted can choose to reply or not to reply.

文中人脸的特征提取主要用了 VGG-FACE 网络进行提取，而回归器则是用了支持向量机以及随机森林，文章的实验是带 RBF 核的支持向量机作为回归器效果要优。但是现在的做法一般用卷积层加全连接层就可以解决了，文中并没有与这种主流的方法进行对比，而且相关系数才 0.4 左右，效果并不是很好，也许与数据集有关吧。

文章还做了一个探讨年龄与颜值的相关性的实验，女生的年龄与颜值的相关性有 0.67 左右，而男生的则为 0.334，这说明了男生在考虑女生颜值的时候会更加注重女生的年龄。

14. RankIQA: Learning from Rankings for No-reference Image Quality Assessment

在进行图像质量评估的时候，为了突破数据集数量的限制，本文通过对清晰的图片进行模糊，用 Siamese 网络来评估模糊前后的 ranking 信息，进一步加强原本网络的训练，如下图所示。

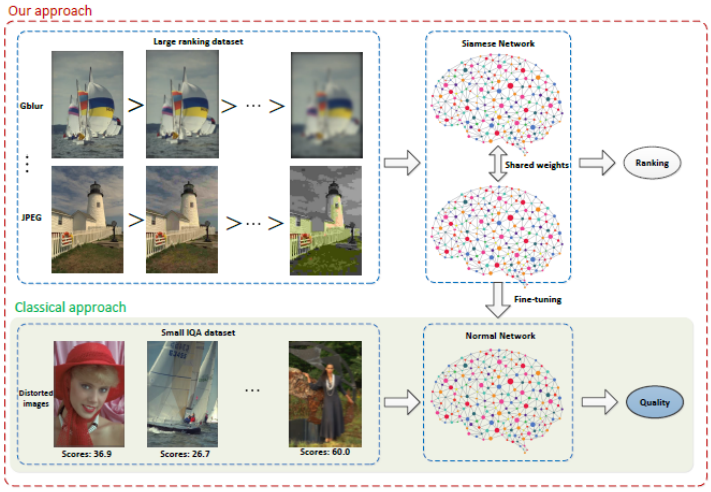


Figure 1. The classical approach trains a deep CNN regressor *directly* on the ground-truth. Our approach trains a network from an image *ranking* dataset. These ranked images can be easily generated by applying distortions of varying intensities. The network parameters are then transferred to the regression network for fine-tuning. This allows for the training of deeper and wider networks.

15. LARGE BATCH TRAINING OF CONVOLUTIONAL NETWORKS

这篇文章讲了如何用大 batch size 来训练 ImageNet。

首先作者在 AlexNet 上做实验，发现用 BN 来代替 LRN 可以使得即使没有 warm-up 也可以用比较大的学习率。然而，将 Batch size 扩大到 8k，精度还是下降了，如下图所示：

(a) Alexnet (warm-up 2.5 epochs)

Batch	Base LR	accuracy,%
512	**0.02**	**58.8**
4096	0.04	53.0
4096	**0.05**	**53.1**
4096	0.06	51.6
4096	0.07	0.1
8192	0.02	29.8
8192	**0.03**	**44.8**
8192	0.04	43.1
8192	0.05	0.1

(b) Alexnet-BN (no warm-up)

Batch	Base LR	accuracy,%
512	**0.02**	**60.2**
4096	0.16	58.1
4096	**0.18**	**58.9**
4096	0.21	58.5
4096	0.30	57.1
8192	0.23	57.6
8192	**0.30**	**58.0**
8192	0.32	57.7
8192	0.41	56.5

Algorithm 1 SGD with LARS. Example with weight decay, momentum and polynomial LR decay.

Parameters: base LR γ_0, momentum m, weight decay β, LARS coefficient η, number of steps T
Init: $t = 0, v = 0$. Init weight w_0^l for each layer l
while $t < T$ for each layer l **do**
  $g_t^l \leftarrow \nabla L(w_t^l)$ (obtain a stochastic gradient for the current mini-batch)
  $\gamma_t \leftarrow \gamma_0 * \left(1 - \frac{t}{T}\right)^2$ (compute the global learning rate)
  $\lambda^l \leftarrow \frac{||w_t^l||}{||g_t^l|| + \beta ||w_t^l||}$ (compute the local LR λ^l)
  $v_{t+1}^l \leftarrow m v_t^l + \gamma_{t+1} * \lambda^l * (g_t^l + \beta w_t^l)$ (update the momentum)
  $w_{t+1}^l \leftarrow w_t^l - v_{t+1}^l$ (update the weights)
end while

为了使得提升 Batch size 之后能得到跟小 Batch size 差不多的结果，文章提出了 LARS（Layer wise Adaptive Rate Scaling）算法，算法流程如上图。

传统的 SGD 算法是每一层的学习率都是一样的，这样当学习率很大的时候就会导致权重的梯度变化比权重本身还大，从而导致不收敛。作者发现权重与梯度的比例在 weight 和 bias 都会很不一样，而且在不同的层里也会很不一样，如下图：

Layer	conv1.b	conv1.w	conv2.b	conv2.w	conv3.b	conv3.w	conv4.b	conv4.w
$\|\|w\|\|$	1.86	0.098	5.546	0.16	9.40	0.196	8.15	0.196
$\|\|\nabla L(w)\|\|$	0.22	0.017	0.165	0.002	0.135	0.0015	0.109	0.0013
$\frac{\|\|w\|\|}{\|\|\nabla L(w)\|\|}$	8.48	**5.76**	33.6	83.5	69.9	127	74.6	148
Layer	conv5.b	conv5.w	fc6.b	fc6.w	fc7.b	fc7.w	fc8.b	fc8.w
$\|\|w\|\|$	6.65	0.16	30.7	6.4	20.5	6.4	20.2	0.316
$\|\|\nabla L(w)\|\|$	0.09	0.0002	0.26	0.005	0.30	0.013	0.22	0.016
$\frac{\|\|w\|\|}{\|\|\nabla L(w)\|\|}$	73.6	69	117	**1345**	68	489	93	19

如果学习率比这个比例还要大，那么训练将会很不稳定的，因为作者提出对每一层的学习率均作出自适应的变动。用了 LARS 来训练的超大 Batch size 的 AlexNet 会延缓精度的下降，例如下图中 8k 的 batch size 训练的结果与 0.5k 训练的结果是差不多的。

Table 3: Alexnet and Alexnet-BN: Training with LARS

(a) Alexnet (warm-up for 2 epochs)

Batch	LR	accuracy,%
512	2	58.7
4K	10	58.5
8K	10	58.2
16K	14	55.0
32K	TBD	TBD

(b) Alexnet-BN (warm-up for 5 epochs)

Batch	LR	accuracy,%
512	2	60.2
4K	10	60.4
8K	14	60.1
16K	23	59.3
32K	22	57.8

因此，超大 Batch size 也是可以训练出很好的结果的，这将对加速网络的训练有着很重要的意义。

16. 100-epoch ImageNet Training with AlexNet in 24 Minutes

这篇文章讲的是用百万美元的设备用 LARS 算法在 24 分钟内完成 ImageNet 在 AlexNet 上的训练。借助 LARS（Layer wise Adaptive Rate Scaling）算法可以将 batch size 设置到很大（例如 32k）进行训练，这里用到了 data parallel 的分布式算法，即每个 GPU 负责 batch size 中的一部分，从而保证大 batch size 是能够跑得动的。

同时，为了保证超大 batch size 能够跑出与小 batch size 差不多的精度（起码不要掉），一般会用到两个技巧：

- 当我们将 batch size 从 B 升到 kB，那么学习率也应该提升 k 倍；

- Warmup 的策略：先用小的 lr_rate 进行训练，几个 epoch 之后再恢复到原来的 lr_rate 进行后续的训练。

结合这个技巧以及 LARS 算法，作者在 24 分钟内可以训练出一个 AlexNet 网络，打破世界纪录。

17. Pyramid Scene Parsing Network

港中文提出的 PSPNet，在 PASCAL 以及 Cityscapes 数据集上的 mIoU accuracy 上分别达到 85.4%以及 80.2%，打破了新的纪录。文章的主要思想从下图就可以容易看出：

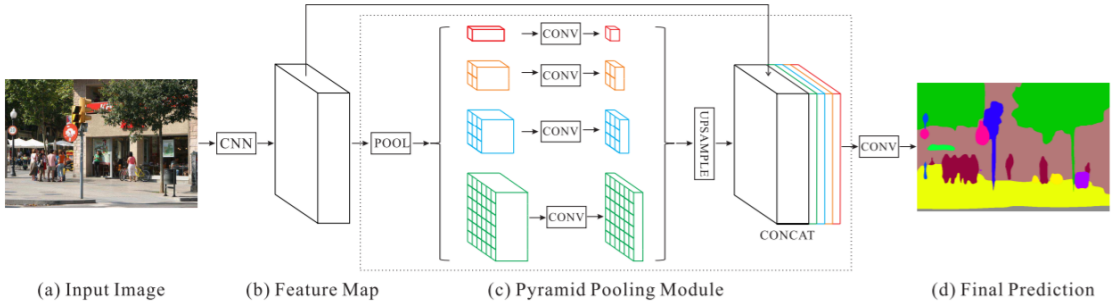


Figure 3. Overview of our proposed PSPNet. Given an input image (a), we first use CNN to get the feature map of the last convolutional layer (b), then a pyramid parsing module is applied to harvest different sub-region representations, followed by upsampling and concatenation layers to form the final feature representation, which carries both local and global context information in (c). Finally, the representation is fed into a convolution layer to get the final per-pixel prediction (d).

主要是用过 pooling 操作得到不同尺度的 feature map 进行进一步的卷积，最后将不同 scale 的 feature map 通过 upsample 进行 concate，最后得到不错的结果。

18. Feature-Fused SSD: Fast Detection for Small Objects

本文的重心主要在用改进后的 SSD 解决小物体的快速检测问题。文中用了两种特征融合的方法使得相对原始 SSD 分别能提 1.6 和 1.7 个点的 mAP，更重要的是在小物体检测上能提 2-3 个点左右，这两种特征融合方法分别是 concate 模块以及 element sum 模块。下图是本文所用到的主要网络结构：

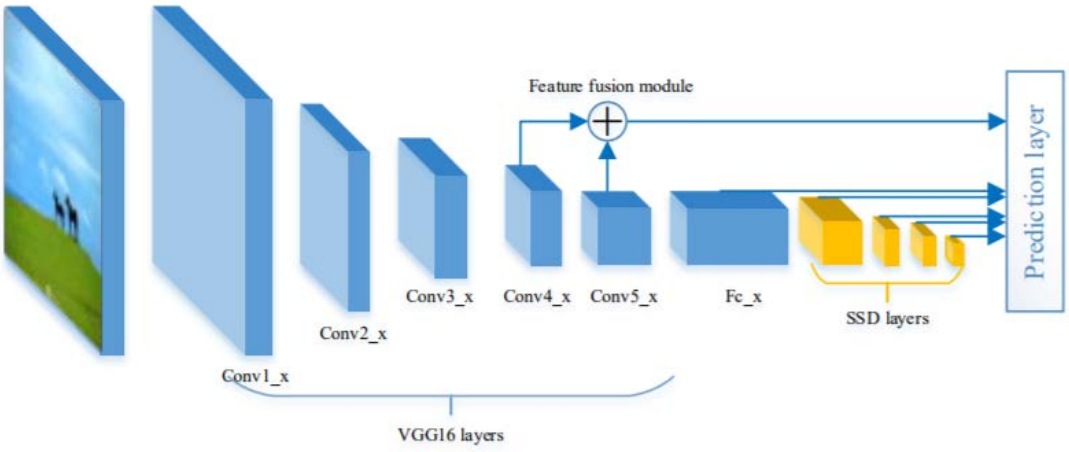


Figure 2. Feature-fused SSD architecture.

下图为两种不同的特征融合方式：

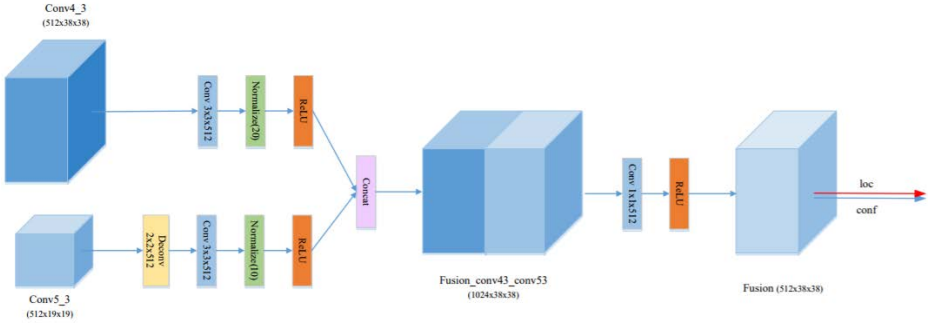


Figure 4. Illustration of the concatenation module.

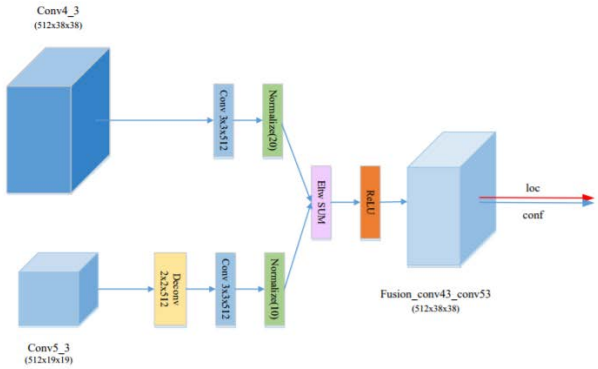


Figure 5. Illustration of the element-sum module.

19. B-CNN: Branch Convolutional Neural Network for Hierarchical Classification

本文建立了一个 Branch-CNN 来进行 coarse-to-fine 多层次的分类，例如下图 1 先将 Cifar10 中的 10 分类粗略的分为两类——交通工具和动物，然后再进一步分类，例如交通工具可以进一步分为水陆空，最后像路这些课进一步分为汽车和卡车等等。这样从粗略到详细进行分类，粗略的分类可在浅层进行，细致的分类可在深层进行，网络结构如下图 2 所示。

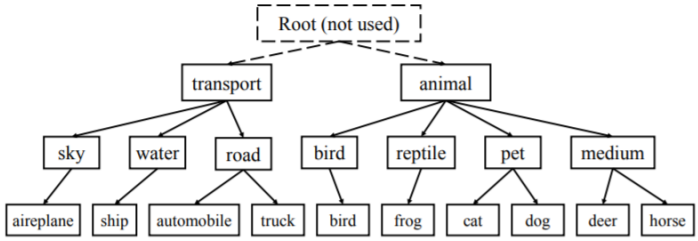


Figure 2: A manually constructed label tree for CIFAR-10 dataset.

图 1

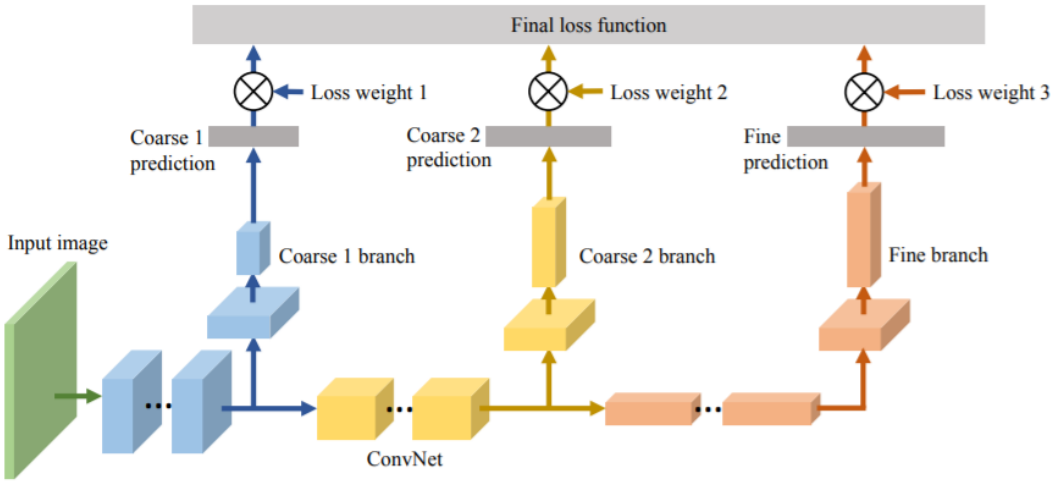


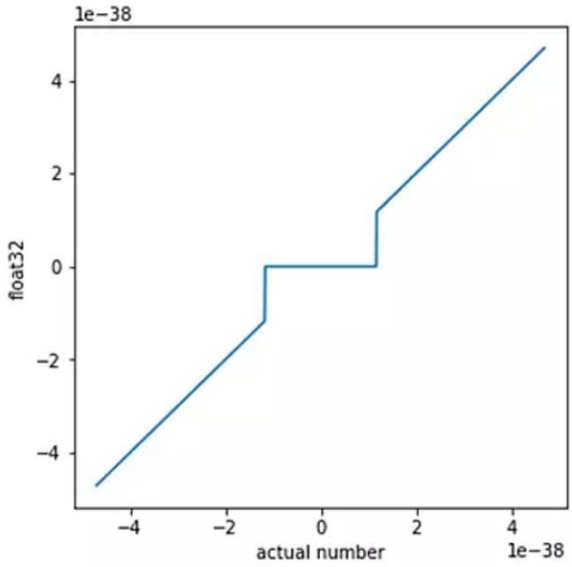
图 2

该网络在 Cifar10、Cifar100、Mnist 上均能使性能得到进一步的提升。

20. Nonlinear-computation-in-linear-networks

https://blog.openai.com/nonlinear-computation-in-linear-networks/

这是 OpenAI 研究员在其博客上介绍的一个有趣的实验，对我们的研究也是具有启发性的，他们发现线性的神经网络其实也可以做非线性的运算，因为浮点数在零附近其实有一个比较大的偏差。在以往的理论中，深度网络是由线性层和非线性层组成的，如果没有非线性层，仅有多个线性层进行叠加，其实就相当于一个线性层而已。然而，他们发现了浮点运算其实也是有非线性的，深层线性网络其实也是可以训练的。



在 MNIST 数据集上的训练，一个用反向传播的深层线性网络可以实现 94%的训练精度和 92%的测试精度；而使用进化策略（Evolution Strategies，这是一种不用反向传播的优化方法），可以实现大于 99%的训练精度和 96.7 的测试精度。进化策略平时用在强化学习里比较多，下一篇大体讲一下进化策略。

21. Evolution Strategies as a Scalable Alternative to Reinforcement Learning

——原文链接：https://blog.openai.com/evolution-strategies/

进化策略算法是一种不需要反向传播的优化方法，直观上来讲，这种优化就是一种猜测然后检测的过程，即我们从一些随机参数开始，然后重复执行以下过程：

1）随机对该猜测进行一点调整，如高斯抖动；

2）让我们的猜测向效果更好的方向移动一点。

具体而言，就是在每个步骤输入一个参数向量 w，然后通过高斯噪声对 w 进行抖动来生成一群（比如 100 个）有稍微改变的参数向量 w1，w2……w100。然后我们在环境中分别运行这 100 个候选项所对应的策略网络，从而独立地对这 100 候选项分别进行评估，然后将每个案例中的奖励加起来。然后其更新后的参数就变成了这 100 个向量的加权和，其中每个权重都正比于其总奖励。（即，我们想让更成功的候选项有更高的权重。）在数学上，你也会注意到这就相当于使用有限差分法（finite difference）来估计参数空间中预期奖励的梯度，只是我们是沿着 100 个随机方向来做的。

22. Oriented Response Networks

CNN 网络在处理图像的局部或者全局旋转上还是存在一定的限制的，为了解

决这个问题，中国科学院大学和杜克大学一起提出了 ORNs 网络，其核心在于一种叫 Active Rotating Filters（ARFs）的卷积核，ARFs 就像是一个卷积核及其多个旋转版本的集合。在图像分类等任务中，用 ORNs 网络可以使得输入图像更加具有旋转不变性。在 VGG、ResNet、STN 等网络的滤波器换为 ARFs，可以进一步提升网络性能。

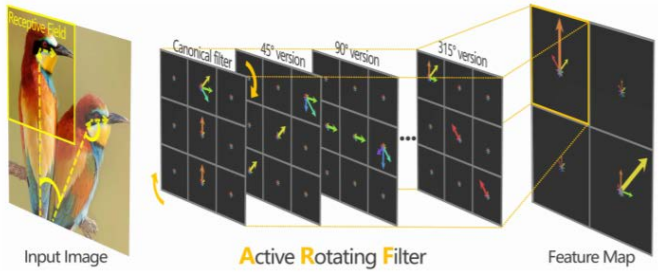


Figure 1. An ARF is a filter of the size $W \times W \times N$, and viewed as N-directional points on a $W \times W$ grid. The form of the ARF enables it to effectively define relative rotations, *e.g.*, the head rotation of a bird about its body. An ARF actively rotates during convolution; thus it acts as a virtual filter bank containing the canonical filter itself and its multiple unmaterialised rotated versions. In this example, the location and orientation of birds in different postures are captured by the ARF and explicitly encoded into a feature map.

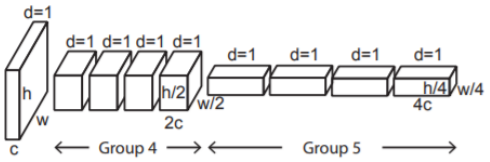
如上图所示，ARFs 卷积核只有第一个卷积核是可学习的，而其他卷积核都是其旋转后的版本。

23. Deep neural networks are more accurate than humans at detecting sexual orientation from facial images

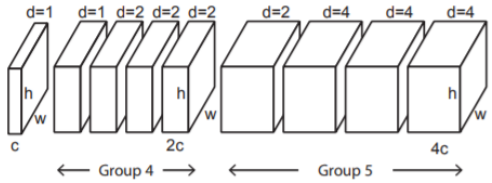
这篇文章是前段时间争议非常大的一篇文章，本身的方法非常简单，用 VGG-FACE 提取特征，然后用 SVD 压缩特征，最后用一个简单的回归模型进行分类，最后男性为同性恋的 AUC 为 0.91。但其重要意义在于连性取向都可以用深度学习来判别，也从侧面印证了“相由心生”这个道理，跟之前那篇通过人脸判别此人是不是犯罪其实也有点类似。

24. Dilated Residual Networks

传统的 CNN 都是随着网络的加深不断地减低 feature map 的分辨率，使其在空域上的结构和细节不再敏锐，从而影响网络的精度，本文在 ResNet 的基础上用了 Dilated 来代替 downsample 的操作，使得每一层的分辨率都能得到保持，如下图所示：



(a) ResNet



(b) DRN

Figure 1: Converting a ResNet into a DRN. The original ResNet is shown in (a), the resulting DRN is shown in (b). Striding in $\mathcal{G}_1^4$ and $\mathcal{G}_1^5$ is removed, bringing the resolution of all layers in $\mathcal{G}^4$ and $\mathcal{G}^5$ to the resolution of $\mathcal{G}^3$. To compensate for the consequent shrinkage of the receptive field, $\mathcal{G}_i^4$ and $\mathcal{G}_1^5$ are dilated by a factor of 2 and $\mathcal{G}_i^5$ are dilated by a factor of 4, for all $i \geq 2$. c, $2c$, and $4c$ denote the number of feature maps in a layer, w and h denote feature map resolution, and d is the dilation factor.

图 1

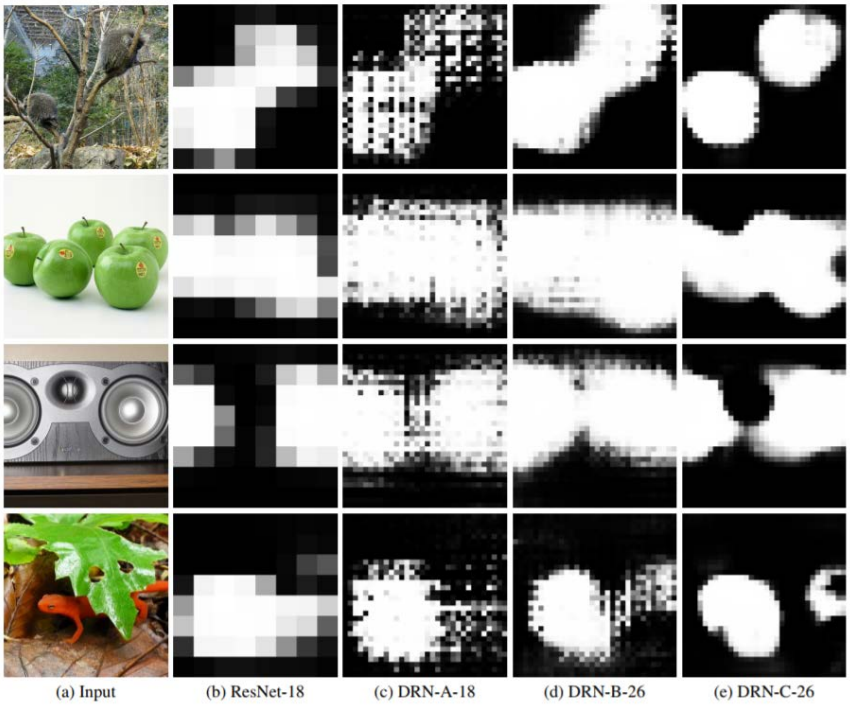


Figure 3: Activation maps of ResNet-18 and corresponding DRNs. A DRN constructed from ResNet-18 as described in Section 2 is referred to as DRN-A-18. The corresponding DRN produced by the degridding scheme described in Section 4 is referred to as DRN-C-26. The DRN-B-26 is an intermediate construction.

图 2

但是这样做会导致 gridding artifacts 的效果，如下图 2 中的 C 图，会出现许多小格子，为了去除 gridding artifacts，作者做了从 DRN-A 到 DRN-B 到 DRN-C 的改进，如下图：

Figure 5: Changing the DRN architecture to remove gridding artifacts from the output activation maps. Each rectangle is a Conv-BN-ReLU group and the numbers specify the filter size and the number of channels in that layer. The bold green lines represent downsampling by stride 2. The networks are divided into levels, such that all layers within a given level have the same dilation and spatial resolution. (a) DRN-A dilates the ResNet model directly, as described in Section 2. (b) DRN-B replaces an early max pooling layer by residual blocks and adds residual blocks at the end of the network. (c) DRN-C removes residual connections from some of the added blocks. The rationale for each step is described in the text.

这些改进总结来讲就是三点：

- 去除 max pooling；
- 在最后加几层 Dilated 比较小的卷积层；
- 去除最后两层的残差连接。

改进之后最后一层的激活图如图 2 中的 D 和 E 所示，gridding artifacts 被抑制了。

这个网络不仅在图像分类中有好处，同时在物体检测，图像分割上均有好处。

25. Deep Convolutional Neural Network for Age Estimation based on VGG-Face Model

这篇文章是用 VGG-FACE 模型做人脸年龄预测的，作者 claims 他们的贡献在于：

- 在做年龄预测之前，先将该模型在人脸识别任务上进行训练，然后把该模型拿来做年龄预测会得到更好的结果；
- 在大量人脸识别数据库上先进行预训练可以避免过拟合的风险；
- 预训练做得好不好也会对模型的性能造成很大的影响。

这其实就是迁移学习嘛，我现在做的是直接拿 ImageNet 上预训练好的模型来用。

26. Learning Two-Branch Neural Networks for Image-Text Matching Tasks

这篇论文是用 two-branch 网络来做文本图片匹配。文中提了两种 two-branh 网络，分别为 Embedding Network 和 Similarity Network，如下图所示。Embedding Network 的效果要好一些，并在 Flickr30k 和 MSCOCO 中的 region-to-phase 和 image-to-sentence 的任务中取得 state-of-the-art 的结果。

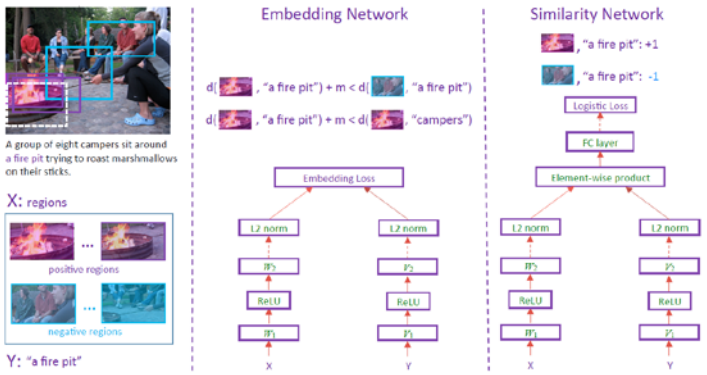


Fig. 1. Taking the phrase localization task as an example, we show the architectures of the two-branch networks used in this paper. Left column: given the phrase "a fire pit" from the image caption, sets of positive regions (purple) and negative regions (blue) are extracted from the training image. The positive regions are defined as ones that have a sufficiently high overlap with the ground truth (dashed white rectangle). X and Y denote the feature vectors describing image regions and phrases, respectively. In this paper, X are features extracted from pre-trained VGG networks, and Y are orderless Fisher Vector text features [2]. Middle: the embedding network. Each branch consists of fully connected (FC) layers with ReLU nonlinearities between them, followed by L2 normalization at the end. We train this network with a maximum-margin triplet ranking loss that pushes positive pairs closer to each other and negative pairs farther (Section 3.2). Right: the similarity network. As in the embedding network, the branches consist of two fully connected layers followed by L2 normalization. Element-wise product is used to aggregate features from two branches, followed by several additional fully connected (FC) layers. The similarity network is trained with the logistic regression loss function, with positive and negative image-text pairs receiving labels of "+1" and "-1" respectively (Section 3.3).

27. Facial beauty analysis based on features prediction and beautification models

这篇文章主要做了两件事情，分别为人脸美丽值的打分和人脸美化。

在人脸打分方面，文章主要用了几何特征和纹理特征来打分。

下面的表为作者提取的 42 个几何特征，然后用 KNN 的方法来判别哪个几何特征对人脸分数预测是有用的。下面也列出了选取几何特征的方法。

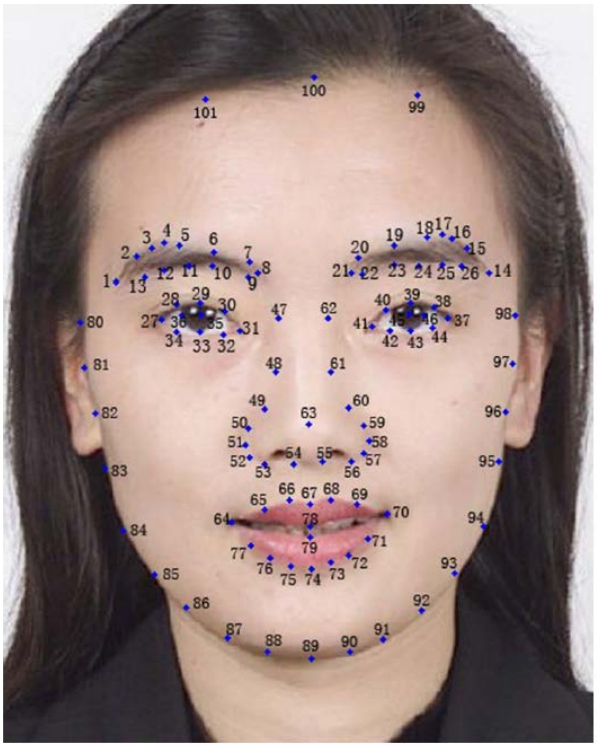


Fig. 6 Number of each landmark point

图 1

表 1

Description	**1**:X8-X1	**2**:X21- X8	**3**:X14-X21	**4**:X31-X27	**5**:X41-X31	**6**:X35-X36
Description	**7**:X37-X41	**8**:X45-X35	**9**:X27-X80	**10**:X98-X37	**11**:X62-X47	**12**:X55-X54
Description	**13**:X61-X48	**14**:X60-X49	**15**:X46-X45	**16**:X58-X51	**17**:X97-X81	**18**:X70-X64
Description	**19**:X94-X84	**20**:X95-X83	**21**:X98-X80	**22**:X93-X85	**23**:Y12-Y14	**24**:Y10-Y6
Description	**25**:Y25-Y17	**26**:Y23-Y19	**27**:Y54-Y47	**28**:Y56-Y45	**29**:Y33-Y29	**30**:Y43-Y39
Description	**31**:Y8-Y100	**32**:Y54-Y8	**33**:Y70-Y45	**34**:Y67-Y55	**35**:Y78-Y67	**36**:Y89-Y70
Description	**37**:Y89-Y54	**38**:Y84-Y35	**39**:Y83-Y80	**40**:Y74-Y79	**41**:Y89-Y74	**42**:Y63-Y100

The number in bold stands for the no. of the feature

算法 1 选取几何特征的方法：

(1) Let A be a set composed of the 42 initial geometric features. If we use {a1, a2, a3,…, a42} to denote the 42 initial geometric features, then A = {a1, a2, a3,…, a42}. B is used to store selected features and is initially set to B = 0;

(2) For each feature in set A, we use the KNN to learn a machine rating for each testing image and then use the Pearson correlation to evaluate the result. If feature a1 obtains the best result, i.e., the highest correlation, then A = {a2, a3,…, a42} and B = {a1};

(3) Choosing each feature of A to combine with all the features in B to perform testing. If feature a3 obtains the best result, then A = {a2, a4,…, a42} and B = {a1, a3};

(4) Repeating Steps 2 and 3 until A = 0 and B contains the 42 initial geometric features. We then take the combination of features that achieves the best testing result to represent the facial beauty.

而纹理特征则用的是 LBP 特征，并用了 BLBP 模型和 PCA-Net 来进行人脸颜值分析。下图为 BLBP 模型。

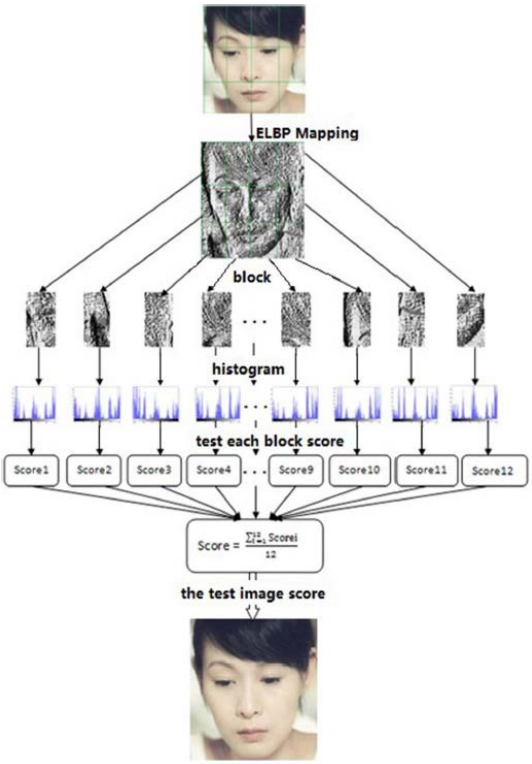


图 2 BLBP 模型

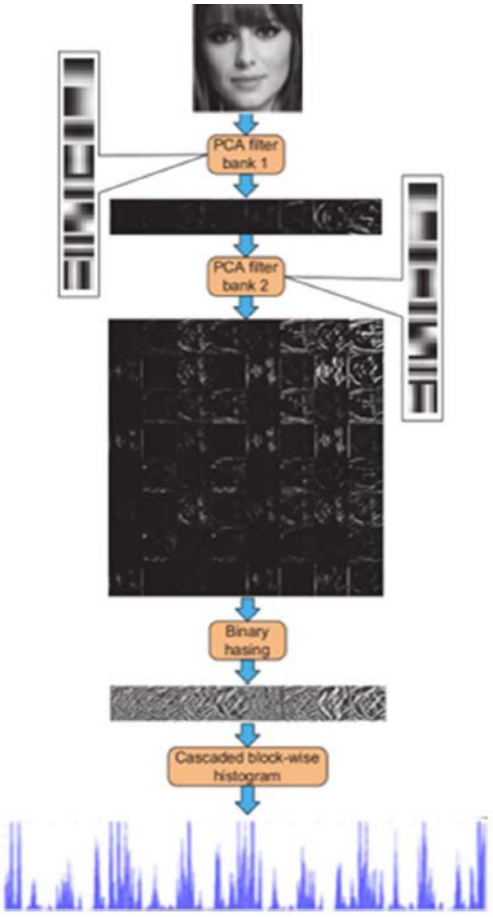


图 3 PCANet

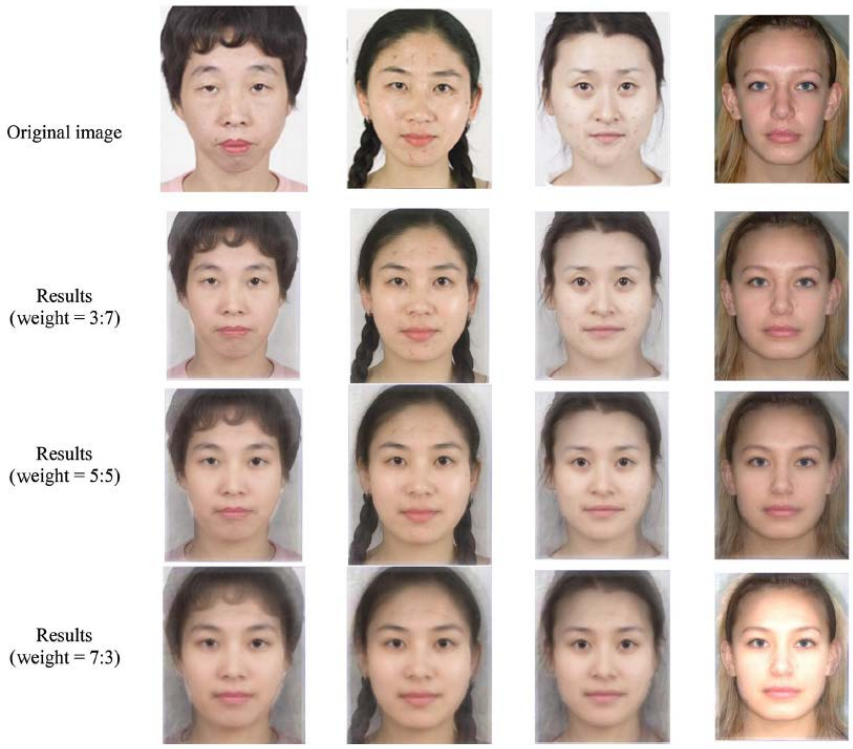
文章最后是选取了几何特征融合 BLBP 特征和 PCANet 进行了人脸颜值预测，在他们自己的数据集上跑出了相关系数 0.889 的成绩。

Table 4 Experimental results of facial beauty prediction using different learning models

	Pearson correlation	Running time (s)	Learning model
Geo	0.762	<2	*KNN*
BLBP	0.852	<35	HM
PCANet	0.858	<2	SVR
Average face model	0.452	<1	DM
Golden ratio	0.323	<1	DM
Three court five eyes	0.441	<1	DM
Geo + BLBP	0.864	<36	*KNN* + HM
BLBP + *PCANet*	0.882	<37	HM + SVR
Geo + *PCANet*	0.869	<3	*KNN* + SVR
PCANet + *Geo* + BLBP	0.889	<38	SVR + *KNN* + HM

文章下半部分还做了人脸美化的工作，包括了下面三步：

- 通过上面的几何特征，找到与需要美化的图片最相似的 10 张图片，这 10 张图片的颜值同时必须必目标图片高，最后用 moving least squares (MLS) 进行人脸的轮廓美化。
- 通过人脸 landmarks 找到人脸皮肤区域，然后用 multi-level 的中值滤波进行皮肤美化。
- 平均脸美化。用基于人脸 landmarks 的 ASM 算法进行人脸的三角对齐，构造平均脸，平均脸主要有颜值较高的多张脸组合而成。但是从实验结果来看，平均脸美化会有肉眼可见的失真，如下图。

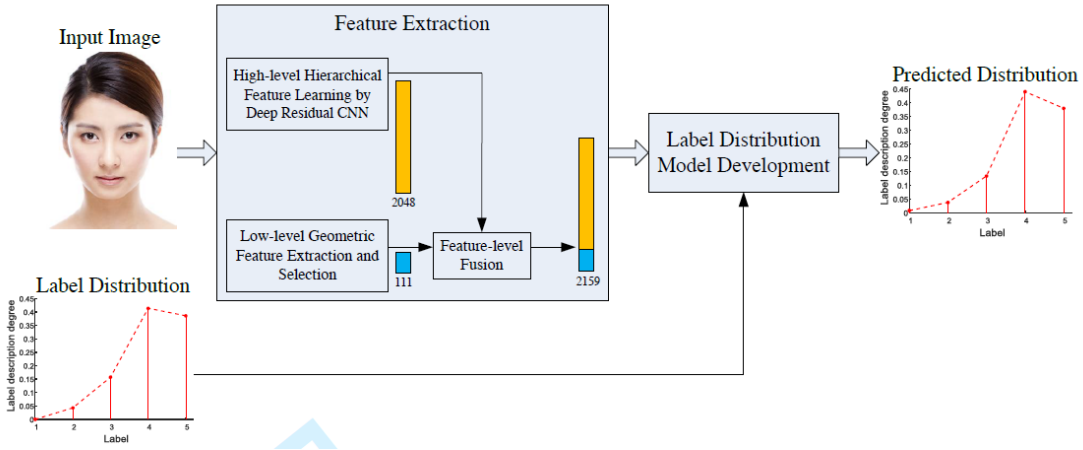


28. Label Distribution Based Facial Attractiveness Computation by Deep Residual Learning

这篇文章主要的工作为一下三个点：

- 用 ResNet 提取人脸特征；
- 把人脸打分的问题当成标签分布的问题来解决；
- 结合 low-level 的人脸几何特征进一步增强 ResNet 提取出来的高层语义特征，进一步提升网络的 performance。

这篇文章所用到的模型框架如下图所示：



一般的 ground-true 标签取自多位打分者为每张人脸打分的平均值。而本文的重点——标签分布，简单来讲就是统计多位打分者为每张人脸打分的分布，如

下面两条式子，y 为多个分数的集合，D 为每个分数的概率的集合。

$$\mathbf{y} = \{y_1, y_2, \ldots, y_c\}$$

$$\mathbf{D} = \{d_{\mathbf{x}}^{y_1}, d_{\mathbf{x}}^{y_2}, \ldots, d_{\mathbf{x}}^{y_c}\}$$

COMPARISON WITH THE STATE-OF-THE-ART*

Method	PC	MAE	RMSE
Xie et al. [21]	0.6482	0.3931	0.5149
Liu et al. [37]	0.6938	0.4441	0.5768
Liu et al. [36]	0.7335	0.4109	0.5465
Wang et al. [19]	0.6406	0.9316	1.1233
Xu et al. [22]	0.8800	-	-
Ours	**0.9301**	**0.2127**	**0.2781**

* The first three rows are the results using low-level features; the following two rows are using deep features from shallow networks. Xie et al. [21] also obtained a correlation of 0.8187 with deep features and further improved to 0.88 in [22].

每张人脸即使他们的分数的均值是一样的，但是他们的分布有可能是不一样的，这样就为网络的训练提供更多的信息，进而提高网络 performance，这篇文章最后在 SCUT-FBP 数据集上可以达到相关系数 0.93 的结果。

29. Sense Beauty by Label Distribution Learning

这篇文章是今年 IJCAI 上的一篇论文，也是用标签分布来做人脸打分的。这篇文章跟上篇文章不一样的在于两个点：

- 提出一种方法可以将两两对比的打分方法得到的结果转化为标签分布；
- 用 structure SVM 来预测人脸分数分布情况。

人们用来打分有两种方法，一种是直接打分，这种要转化为分数分布跟上篇文章一样；还有一种方法是进行两两比较来打分，没有一个绝对的分数，这种情况下如何转化为分数的分布，文中提出可以用最大似然估计来解决，如下列公式，即求两个不同样本的联合概率分布，使得集合内 P（xi,xj）最大，P（xi,xj）为 xi 大于 xj 的概率。

$$
\begin{aligned}
\underset{\boldsymbol{d}_n, n=1,\cdots,N}{\arg\max} \quad & \sum_{(i,j)\in S} \log P(\boldsymbol{x}_i \succ \boldsymbol{x}_j) - \lambda \sum_n v(\boldsymbol{d}_n) \\
= & \sum_{(i,j)\in S} \log \sum_{y_p > y_q} P(y_p|\boldsymbol{x}_i) P(y_q|\boldsymbol{x}_j) \\
& -\lambda \sum_n \sum_l P(y_l|\boldsymbol{x}_n)(y_l - \sum_{l'} y_{l'} P(y_{l'}|\boldsymbol{x}_n))^2 \\
= & \sum_{(i,j)\in S} \log \sum_{y_p > y_q} d_{\boldsymbol{x}_i}^{y_p} d_{\boldsymbol{x}_j}^{y_q} \\
& -\lambda \sum_n \sum_l d_{\boldsymbol{x}_n}^{y_l} (y_l - \sum_{l'} y_{l'} d_{\boldsymbol{x}_n}^{y_{l'}})^2 \\
\text{s.t.} \quad & \forall l = 1, \cdots, c, \forall n = 1, \cdots, N \\
& 0 < d_{\boldsymbol{x}_n}^{y_l} < 1, \\
& \sum_l d_{\boldsymbol{x}_n}^{y_l} = 1.
\end{aligned}
$$

当然，基于大多数人的审美都是相似的原则，这条式子里面还加了一个正则项，使得 c 个人对图片的打分的偏差不至于过大，即使得分数的分布平滑。

本文最后用 structure SVM 在 SCUT-FBP 和 M²B 数据集上进行人脸分数分布的预测，在 SCUT-FBP 上的结果如下所示：

Algorithm	MAE	RMSE
sDFAT	0.4065	0.5647
SVR	0.3549	0.4643
k-NN	0.3920	0.5116
SLDL	**0.3015**	**0.4076**

Table 2: The experimental results of the predicting task on SCUT-FBP dataset

这篇文章没有做相关性分析，单从 MAE 和 RMSE 这两个指标来看的话，performance 没有上一篇好。