　附表2-1（通用）

南京邮电大学

毕业设计(论文)外文资料翻译

|  |  |
| --- | --- |
| 学 院 | 计算机学院、软件学院、网络空间安全学院 |
| 专　　业 | 数据科学与大数据技术 |
| 学生姓名 | 张靖雅 |
| 班级学号 | B21032209 |
| 外文出处 | Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 4700-4708. |

附件：1.外文资料翻译译文；2.外文原文

|  |
| --- |
| 指导教师评价：  1．翻译内容与课题的结合度： ☑ 优 □ 良 □ 中 □ 差  2．翻译内容的准确、流畅： ☑ 优 □ 良 □ 中 □ 差  3．专业词汇翻译的准确性： □ 优 ☑ 良 □ 中 □ 差  4．翻译字符数是否符合规定要求：☑ 符合 □ 不符合  　　　　　　　　　　　　　　　　　　　　　　　指导教师签名：  　　　　　　　　　　　　　　　　　　　　　　　　　　 2025年4月14日 |

**密集连接的神经网络**

黄高（康奈尔大学） 刘庄（清华大学）

Laurens van der Maaten（Facebook 人工智能研究）

Kilian Q. Weinberger（康奈尔大学）

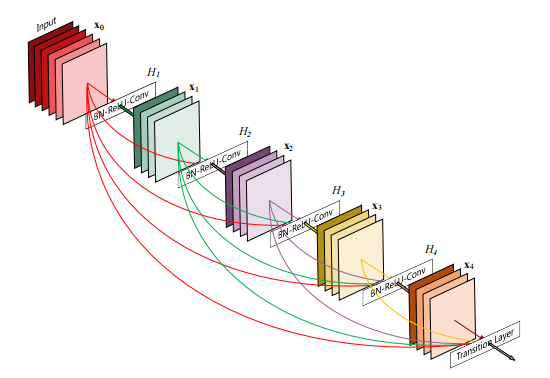
摘要——最近的研究表明，如果卷积网络在靠近输入和接近输出的层之间包含较短的连接，那么卷积网络可以更深入、更准确和有效地训练。在本文中，我们采用这种观测方法，并引入了密集卷积网络（Dense Convolutional Network，简称DenseNet），它以一种前馈的方式将每一层连接到另一层。传统的卷积网络如果有层，则有个连接——每一层与其后续层之间有一个连接。而我们的网络有​个直接连接。对于每个图层，前面所有图层的特征图都作为输入，而它自己的特征图则作为后续所有层的输入。DenseNets具有几个显著的优势：它缓解了梯度消失问题，加强了特征传播，鼓励特征重用，并大幅减少了参数数量。我们在四个竞争激烈的对象识别基准测试任务（CIFAR-10、CIFAR-100、SVHN和ImageNet）上评估了我们提出的架构。DenseNets在大多数方面都显著优于现有的最先进方法，同时需要较少的计算来实现高性能。代码和预训练模型可在以下链接获取：[https://github.com/liuzhuang13/DenseNet](https://github.com/liuzhuang13/DenseNet。)。

**1. 引言**

卷积神经网络（CNNs）已经成为视觉目标识别的主流机器学习方法。尽管它们最初是在 20 多年前引入的[18]，但直到最近计算机硬件和网络结构的改进才使得真正的深度 CNN 的训练成为可能。最初的 LeNet5 [19] 由 5 层组成，VGG 有 19 层 [28]，直到去年 Highway Networks [33] 和 Residual Networks (ResNets) [11] 才突破了 100 层的障碍。

随着CNNs变得越来越深入，一个新的研究问题出现了：当输入或梯度信息通过许多层传递时，它们可能会在到达网络末端（或起始端）时消失并被“冲走”。许多最近的出版物都解决了这个问题或相关问题。 ResNets [11] 和 Highway Networks [33] 通过等值连接将一层的信号旁路到下一层。随机深度（Stochastic Depth）[13]通过在训练期间随机丢弃层来缩短ResNets，从而实现更好的信息和梯度流动。FractalNets [17]将多个并行层序列与不同数量的卷积块重复组合以获得大的标称深度，同时在网络中保持许多短路径。尽管这些不同的方法在网络拓扑和训练过程上有所不同，但它们都有一个关键特征：它们创建从早期层到后期层的短路径。

在本文中，我们提出了一种架构，将这种观点提炼为一种简单的连接模式：为了确保网络中各层之间的最大信息流，我们将所有层（具有匹配的特征图大小）直接相互连接。为了保持前馈性质，每一层从所有前面0的层获得额外的输入，并将其自身的特征图传递给所有后续层。**图1**示意性地展示了这种布局。至关重要的是，与ResNets不同，我们在将特征传递到层之前从不通过求和来组合特征；相反，我们通过连接它们来组合特征。因此，第层有个输入，由所有前面的卷积块的特征图组成。它自身的特征图被传递给所有个后续层。这种连接模式在一个层网络中引入了​个连接，而不是传统架构中的个连接。由于其密集的连接模式，我们称我们的方法为密集卷积网络（Dense Convolutional Network，DenseNet）。



**图 1**：增长率为 k=4 的 5 层致密块。每一层都将所有前面的特征图作为输入。

这种密集连接模式的一个可能违反直觉的效果是，它比传统卷积网络需要更少的参数，因为没有必要重新学习冗余的特征图。传统的前馈架构可以被视为一种具有状态的算法，该状态从一层传递到另一层。每一层从其前一层读取状态，并写入到下一层。它改变了状态，但也传递了需要保留的信息。ResNets[11]通过加性恒等变换使信息保存变得明确。 ResNets的最新变化[13]表明，许多层贡献很小，实际上可以在训练过程中随机丢弃。这使得ResNets的状态类似于（展开的）循环神经网络[21]，但ResNets的参数数量要大得多，因为每一层都有自己的权重。我们提出的DenseNet架构明确区分了添加到网络中的信息和保留的信息。DenseNet层非常窄（例如，每层12个滤波器），只向网络的“集体知识”中添加少量特征图，并保持其余特征图不变——最终分类器基于网络中的所有特征图做出决策。

除了更好的参数效率外，DenseNet的一大优势是其改善了整个网络的信息流和梯度，这使得它们易于训练。每层都可以直接访问损失函数和原始输入信号的梯度，从而实现隐式深度监督[20]。这有助于训练更深层的网络架构。此外，我们还观察到密集连接具有正则化效果，可以减少训练集较小的任务的过度拟合。

我们在四个高度竞争的基准数据集（CIFAR-10、CIFAR-100、SVHN和ImageNet）上评估了DenseNets。我们的模型往往需要比现有的具有相当精度的算法少得多的参数，同时在大多数基准任务上显著优于当前的最先进结果。

**2. 相关工作**

自神经网络最初被发现以来，对网络架构的探索一直是神经网络研究的一部分。最近神经网络的重新流行也复兴了这一研究领域。现代网络中层数的增加放大了不同架构之间的差异，并激发了对不同连接模式的探索以及对旧研究思路的重新审视。

类似于我们提出的密集网络布局的级联结构已经在20世纪80年代的神经网络文献中进行了研究[3]。他们的开创性工作侧重于以逐层方式训练的完全连接的多层感知器。最近，提出了用批量梯度下降训练的全连接级联网络[39]。尽管在小数据集上效果显著，但这种方法仅能扩展到包含几百个参数的网络。在[9,23, 30,40]中，通过跳跃连接利用 CNN 中的多级特征已被发现对各种视觉任务有效。与我们的工作并行，[1]推导出了一个纯理论框架，用于具有与我们类似的跨层连接的网络。

Highway Networks [33] 是最早提供有效训练 100 层以上端到端网络方法的架构之一。使用旁路路径和门控单元，可以毫无困难地优化数百层的Highway网络。旁路路径被认为是简化这些非常深的网络训练的关键因素。 ResNets [11]进一步支持了这一点，其中使用纯恒等映射作为旁路路径。 ResNets 在许多具有挑战性的图像识别、定位和检测任务上取得了令人瞩目、创纪录的表现，例如 ImageNet 和 COCO 对象检测 [11]。最近，随机深度被提出作为成功训练 1202 层 ResNets 的一种方法[13]。随机深度通过在训练过程中随机丢弃层来改进深度残差网络的训练。这表明并非所有层都是必需的，并强调深度（残差）网络中存在大量冗余。我们的论文部分受到了这一观察的启发。具有预激活功能的 ResNets 还有助于训练超过 1000 层的最先进网络 [12]。

与使网络更深（例如，借助跳跃连接）的正交方法是增加网络宽度。GoogLeNet[35,36]使用了一个“Inception模块”，该模块连接了由不同大小的滤波器生成的特征图。在[37]中，提出了一种具有宽广义残差块的ResNets变体。事实上，只要深度足够，简单地增加ResNets中每一层的滤波器数量就可以提高其性能[41]。 FractalNets 还使用广泛的网络结构在多个数据集上取得了有竞争力的结果 [17]。

DenseNet 不是从极深或极宽的架构中汲取表征能力，而是通过特征重用来挖掘网络的潜力，产生易于训练且参数效率高的精简模型。连接不同层学习的特征图会增加后续层输入的变化并提高效率。这是DenseNets与ResNets之间的主要区别。与同样连接不同层特征的 Inception 网络 [35,36] 相比，DenseNet 更简单、更高效。

还有其他值得注意的网络架构创新也产生了有竞争力的结果。Network in Network（NIN）[22]结构将微型多层感知器纳入卷积层的滤波器中，以提取更复杂的特征。在深度监督网络（DSN）[20]中，内部层由辅助分类器直接监督，这可以加强早期层接收到的梯度。梯形网络（Ladder Networks） [26,25]在自编码器中引入了横向连接，在半监督学习任务中取得了令人印象深刻的准确率。在[38]中，提出了一种Deeply-Fused Nets（DFNs）来通过组合不同基础网络的中间层来改进信息流动。通过添加最小化重建损失的路径来增强网络也被证明可以改进图像分类模型[42]。

**3. DenseNets**

考虑一张单图像 通过一个卷积网络。该网络包含层，每一层实现一个非线性变换 ，其中表示该层的索引。 是一个复合函数，可能包含批量归一化（Batch Normalization, BN）[14]、修正线性单元（ReLU）[6]、池化（Pooling）[19]或卷积（Convolution）等操作。我们用表示第层的输出。

**ResNets.** 传统的卷积前馈网络将第 层的输出作为第层的输入[16]，这种层间转换可以表示为：。ResNets [11] 添加了一个跳跃连接，通过恒等函数绕过非线性变换：

(1)

ResNets的一个优势是梯度可以通过恒等函数直接从后面的层流到前面的层。然而，恒等函数和的输出通过求和组合，这可能会阻碍网络中的信息流动。

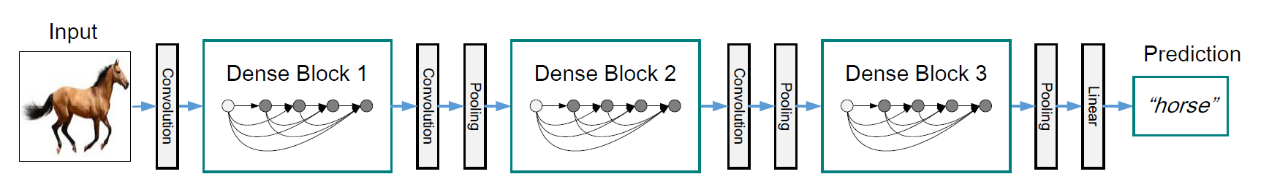
**密集连接（Dense connectivity）.** 为了进一步改善层之间的信息流动，我们提出了一种不同的连接模式：我们引入了从任何一层到所有后续层的直接连接。**图1**示意性地展示了结果DenseNet的布局。因此，第 层接收所有前面层的特征图 ,……, 作为输入：

（2）

其中 表示在第0层到第 层中生成的特征图的连接。由于其密集连接，我们将该网络架构称为密集卷积网络（Dense Convolutional Network，DenseNet）。为了便于实现，我们将等式中的的多个输入连接起来，转化为单个张量如**公式2**。

**复合函数（Composite function）.** 受到[12]的启发，我们将定义为三个连续操作的复合函数：批量归一化（BN）[14]、接着是修正线性单元（ReLU）[6]和最后是 3×3 的卷积操作（Conv）。

**池化层（Pooling layers）.** 当特征图的大小发生变化时，**公式2**中使用的拼接操作就不可行了。然而，卷积网络的一个重要部分是下采样层，它会改变特征图的大小。为了在我们的架构中实现下采样，我们将网络划分为多个密集连接的密集块（dense blocks），如**图2** 所示。我们称块之间的层为过渡层（transition layers），它们执行卷积和池化操作。在我们实验中使用的过渡层由批量归一化层和 1×1 卷积层组成，后面跟着 2×2 的平均池化层。



**图 2**：具有三个密集块的深度 DenseNet。两个相邻块之间的层称为过渡层，并通过卷积和池化改变特征图大小。

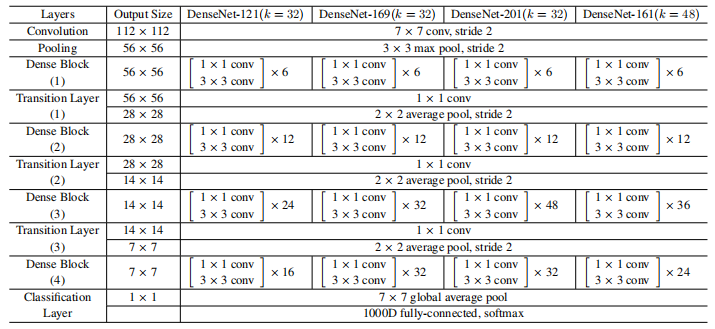
**增长率（Growth rate）.** 如果每个函数生成个特征图，那么第层将有 个输入特征图，其中 是输入层的通道数。DenseNet与现有网络架构的一个重要区别是，DenseNet可以有非常窄的层，例如 。我们将超参数称为网络的增长率。在第4节中，我们展示了在我们测试的数据集上，一个相对较小的增长率就足以获得最优结果。对此的一种解释是，每一层都可以访问其块中所有先前的特征图，因此可以访问网络的“集体知识”。人们可以将特征图视为网络的全局状态。每层将自己的个特征图添加到该状态。增长率调节了每一层为全局状态贡献了多少新信息。一旦写入全局状态，网络内的任何位置都可以访问它，并且与传统网络架构不同，无需逐层复制它。

**瓶颈层（Bottleneck layers）.** 虽然每一层只产生个输出特征图，但它通常有更多的输入。正如在 [36, 11] 中提到的，可以在每个卷积之前引入一个卷积作为瓶颈层，以减少输入特征图的数量，从而提高计算效率。我们发现这种设计对DenseNet特别有效，我们将具有这种瓶颈层的网络，即 的BN-ReLU-Conv(1×1)−BN-ReLU-Conv(3×3)版本称为DenseNet-B。在我们的实验中，我们让每个卷积产生 个特征图。

**压缩（Compression）.** 为了进一步提高模型的紧凑性，我们可以减少过渡层的特征图数量。如果一个密集块包含 m 个特征图，我们让随后的过渡层生成 个输出特征图，其中 被称为压缩因子。当 时，过渡层特征图数量保持不变。我们将压缩因子的DenseNet称为DenseNet-C，并在我们的实验中设置。当同时使用瓶颈层和压缩因子的过渡层时，我们将这种模型称为DenseNet-BC。

**实现细节（Implementation Details）.** 在除了ImageNet之外的所有数据集上，我们在实验中使用的DenseNet包含三个密集块，每个块的层数相等。在进入第一个密集块之前，对输入图像执行一个具有16个输出通道（对于DenseNet-BC，为增长率的两倍）的卷积。对于内核大小为 的卷积层，输入的每一边都用一个像素进行零填充，以保持特征图尺寸不变。我们在两个连续的密集块之间使用 1×1 卷积后接 2×2 平均池化作为过渡层。在最后一个密集块的末尾，执行全局平均池化，然后连接一个softmax分类器。三个密集块中的特征图大小分别为、和。我们实验了配置为{,}、{,} 和 {,}的基本 DenseNet 结构。对于DenseNet-BC，我们评估了配置为 {,}、{,} 和 {,} 的网络。

在我们对ImageNet的实验中，我们在 224×224 输入图像上使用具有 4 个密集块的 DenseNet-BC 结构。初始卷积层由2k个7×7卷积核（步长为2）构成，其余各层特征图数量均以参数k为基数进行设定‌。我们在**表1**中展示了我们在ImageNet上使用的精确网络配置。



**表 1**：ImageNet 的 Densenet 架构。前 3 个网络的增长率为 k=32，DenseNet-161 的增长率为 k=48。请注意，表中显示的每个“conv”层对应于序列 BN-ReLU-Conv。

**4. 实验**

我们通过在多个基准数据集上的实验验证了DenseNet的有效性，并与最先进的架构（尤其是ResNet及其变体）进行了比较。

**4.1 数据集**

CIFAR. CIFAR的两个数据集[15]由32×32像素的彩色自然图像组成。CIFAR-10（C10）包含来自10个类别的图像，而CIFAR-100（C100）则包含来自100个类别的图像。训练集和测试集分别包含50,000和10,000张图像，我们从训练集中划分出5,000张图像作为验证集。我们采用了广泛用于这两个数据集的标准数据增强方案（镜像/平移）[11, 13, 17, 22, 27, 20, 31, 33]，我们在数据集名称的末尾加上一个“+”标记来表示这种数据增强方案（例如，C10+）。对于预处理，我们使用通道均值和标准差对数据进行归一化。在最终运行中，我们使用所有50,000张训练图像，并在训练结束时报告最终的测试误差。

**SVHN.** Street View House Numbers（SVHN）数据集[24]包含像素的彩色数字图像。训练集有73,257张图像，测试集有26,032张图像，另外还有531,131张图像用于额外训练。按照常见做法[7, 13, 20, 22, 29]，我们使用所有训练数据且不进行任何数据增强，并从训练集中划分出6,000张图像作为验证集。在训练过程中，我们选择验证误差最小的模型，并报告其在测试集上的误差。我们按照[41]的方法，将像素值除以255，使其范围在[0, 1]之间。

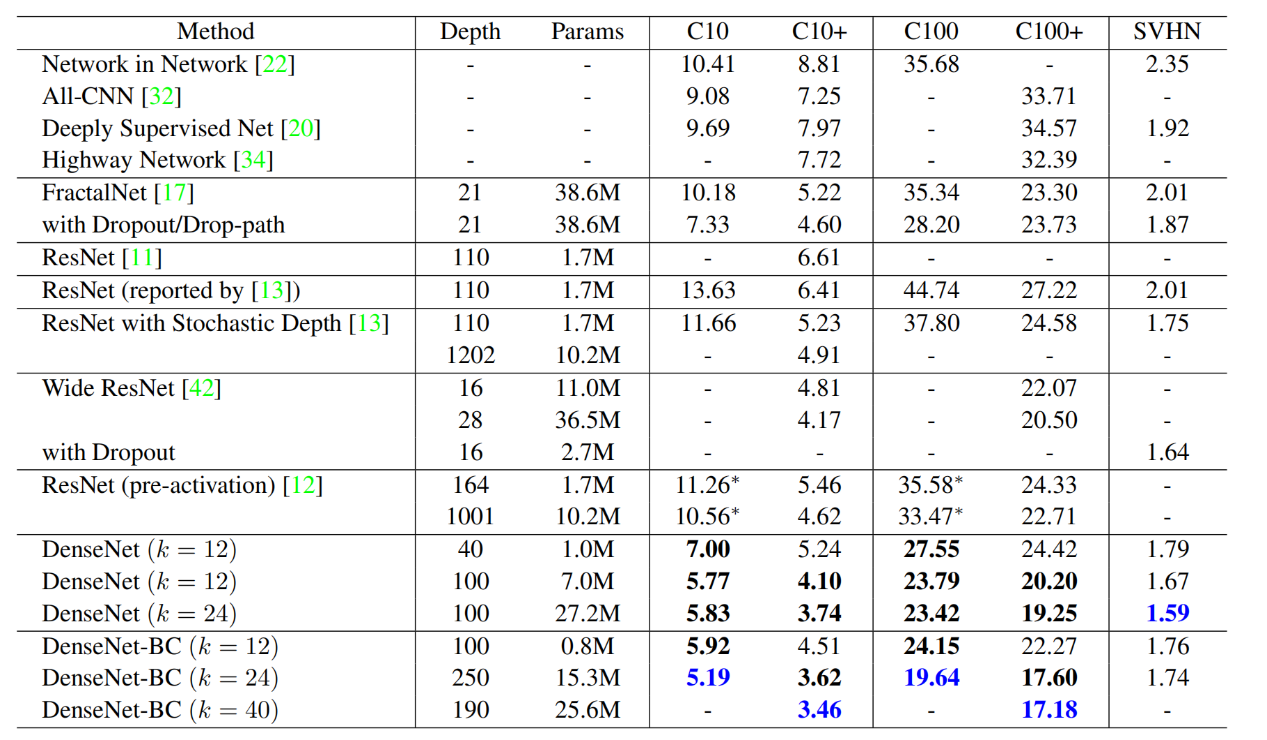
**ImageNet.** ILSVRC 2012分类数据集[2]包含120万张用于训练的图像和50,000张用于验证的图像，分为1000个类别。我们采用了与[8, 11, 12]相同的训练图像数据增强方案，并在测试时应用单裁剪或10裁剪，尺寸为224×224。按照[11, 12, 13]的做法，我们报告验证集上的分类错误率。

**4.2 训练**

所有网络均使用随机梯度下降（SGD）进行训练。在CIFAR和SVHN上，我们分别使用64的批量大小训练300个和40个周期。初始学习率设置为0.1，并在总训练周期的50%和75%时将其除以10。在ImageNet上，我们使用256的批量大小训练模型90个周期。初始学习率设置为0.1，并在第30个和第60个周期时降低10倍。由于GPU内存限制，我们最大的模型（DenseNet-161）使用128的小批量大小进行训练。为了补偿较小的批量大小，我们训练该模型100个周期，并在第90个周期时将学习率除以10。

按照[8]的做法，我们使用的权重衰减和0.9的Nesterov动量[34]，不进行阻尼。我们采用[10]提出的权重初始化方法。对于没有数据增强的数据集（即C10、C100和SVHN），我们在每层卷积层（第一层除外）后添加一个Dropout层[32]，并将Dropout率设置为0.2。每个任务和模型设置的测试误差仅评估一次。

**4.3 CIFAR和SVHN的分类结果**



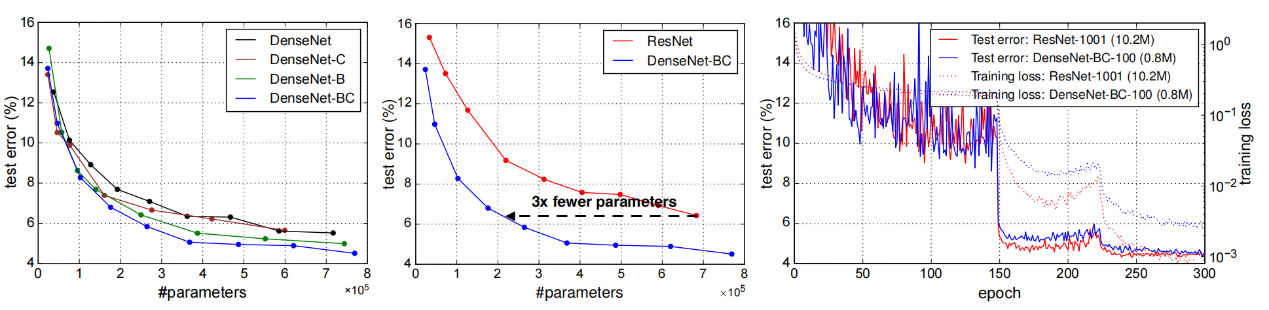
**表 2**：CIFAR 和 SVHN 数据集的错误率 (%)。 k表示网络的增长率。超越所有竞争方法的结果是粗体的，总体最佳结果是蓝色的。 “+”表示标准数据增强（平移和/或镜像）。 \* 表示我们自己运行的结果。所有未使用数据增强的（C10、C100、SVHN）的DenseNets结果都是使用dropout获得的。 Densenets 使用比 ResNet 更少的参数实现了更低的错误率。在没有数据增强的情况下，DenseNet 的性能优势更为显著。

我们训练了不同深度（L）和增长率（k）的DenseNet模型。CIFAR和SVHN的主要实验结果如**表2**所示。为了突出总体趋势，我们将所有优于现有最先进水平的结果用粗体标记，并将总体最佳结果标记为蓝色。

**准确性（Accuracy）.** 最明显的趋势可能来自**表2‌**的底行，该表显示 L=190 和 k=40 的 DenseNet-BC 在所有 CIFAR 数据集上始终优于现有的最先进技术。其在C10+上的错误率为3.46%，在C100+上的错误率为17.18%，明显低于Wide ResNet架构所实现的的错误率[41]。我们在C10和C100（无数据增强）上取得的最佳结果更为令人鼓舞：与采用Drop-path正则化的FractalNet相比，DenseNet-BC的错误率降低了近30%[17]。在SVHN上，使用Dropout的DenseNet（，）也超过了当前Wide ResNet取得的最好成绩。然而，250层的DenseNet-BC并没有比其较短的版本进一步提高性能。这可能是因为SVHN是一个相对简单的任务，极其深的模型可能会过拟合训练集。

**容量（Capacity）.** 在没有压缩或瓶颈层的情况下，DenseNets的性能随着L和k的增加而提高是一种普遍趋势。我们将其主要归因于模型容量的相应增长。C10+ 和 C100+ 列最好地证明了这一点。在C10+上，随着参数数量从1.0M增加到7.0M以上，再到27.2M，错误率从5.24%下降到4.10%，最终下降到3.74%。在C100+上，我们也观察到了类似的趋势。这表明 DenseNet 可以利用更大、更深的模型增强的表示能力。它还表明它们不会遭受过度拟合或残差网络的优化困难[11]。

**参数效率（Parameter Efficiency）.** **表2**中的结果表明，DenseNet比其他架构（尤其是ResNet）更有效地利用参数。具有瓶颈结构和过渡层降维的DenseNet-BC在参数利用方面尤为高效。例如，我们的250层模型只有15.3M参数，但它的性能始终优于 FractalNet 和 Wide ResNets 等具有超过 30M 参数的其他模型。我们还强调，和的 DenseNet-BC 在使用90%更少的参数的情况下，达到了与1001层预激活ResNet相当的性能（例如，在C10+上，错误率分别为4.51%和4.62%，在C100+上，错误率分别为22.27%和22.71%）。**图4**（右图）显示了这两个网络在C10+上的训练损失和测试误差。1001层的深度ResNet收敛到更低的训练损失值，但测试误差相似。我们将在下面更详细地分析这种影响。

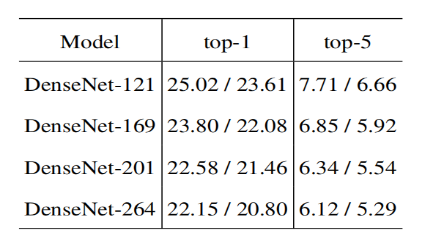


**图4**：左图：DenseNet 变体之间 C10+ 上的参数效率比较。中：DenseNet-bc 和（预激活）ResNet 之间的参数效率比较。 Densenet-bc 需要的参数约为 ResNet 的 1/3 就能达到相当的精度。右：具有超过 10M 参数的 1001 层预激活 ResNet [12] 和仅具有 0.8M 参数的 100 层 DenseNet 的训练和测试曲线。

**过拟合（Overfitting）.** 参数更高效利用的一个积极副作用是，DenseNets往往不太容易过拟合。我们观察到，在没有数据增强的数据集上，DenseNet架构相比之前的工作改进尤为显著。在C10上，错误率从7.33%降低到5.19%，相对减少了29%。在C100上，错误率从28.20%降低到19.64%，减少了约30%。在我们的实验中，我们仅在一种设置下观察到潜在的过拟合现象：在C10上，将k从12增加到24导致参数增加了4倍，但错误率却略有上升，从5.77%增加到5.83%。DenseNet-BC的瓶颈层和压缩层似乎是抵消这一趋势的有效方法。

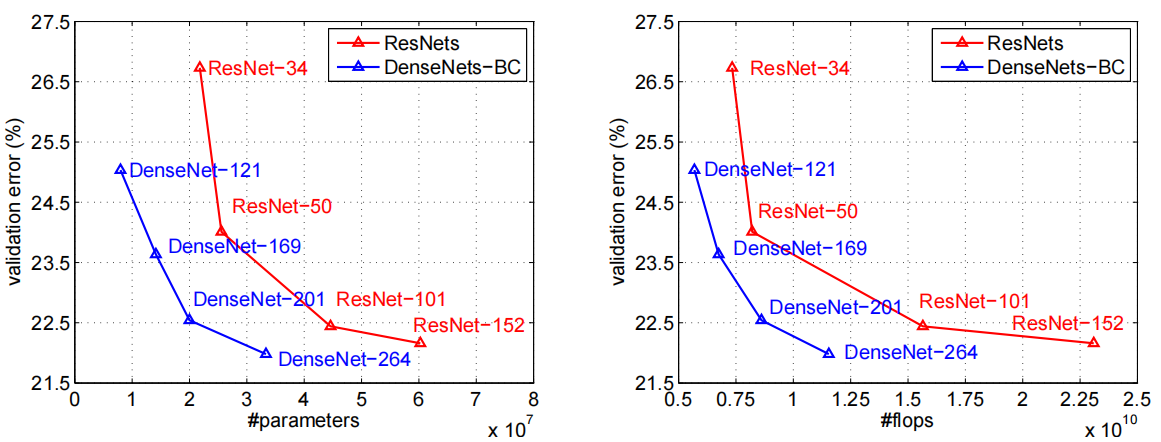
**4.4 ImageNet的分类结果**

我们在ImageNet分类任务上评估了不同深度和增长率的DenseNet-BC，并将其与最先进的ResNet架构进行了比较。为确保两种架构之间的公平比较，我们采用了‌[8]提供的ResNet的Torch公开实现版本，以消除数据预处理和优化设置等其他所有因素上的差异。我们只需将ResNet模型替换为DenseNet-BC网络，并保持所有实验设置与ResNet所使用的设置完全相同。唯一的例外是，由于GPU内存限制，我们最大的DenseNet模型使用了128的小批量大小进行训练；我们训练这个模型100个周期，并在第90个周期时进行第三次学习率降低，以补偿较小的批量大小。



**表 3**：ImageNet 验证集上的 top-1 和 top-5 错误率，采用单次裁剪（10 次裁剪）测试。

**表3**报告了DenseNets在ImageNet上的单裁剪和10裁剪验证错误率。**图3**展示了DenseNets和ResNets的单裁剪Top-1验证错误率与参数数量（左图）和浮点运算次数（FLOPs，右图）的关系。‌图中呈现的结果表明，‌DenseNets的性能与最先进的ResNets相当，但实现相似性能所需的参数和计算量显著减少‌。例如，一个具有20M参数的DenseNet-201模型与一个具有超过40M参数的101层ResNet模型的验证错误率相似。从右图也可以观察到类似的趋势，该图将验证错误率绘制为浮点运算次数的函数：一个计算量与ResNet-50相当的DenseNet，其性能与需要两倍计算量的ResNet-101相当。



**图 3**：ImageNet 验证数据集上的 DenseNets 和 ResNets top-1 错误率（单次裁剪测试）的比较，作为学习参数（左）和测试期间 FLOP（右）的函数。

值得注意的是，我们的实验设置意味着我们使用针对 ResNets 而不是针对 DenseNets 优化的超参数设置。可以想象，更广泛的超参数搜索可能会进一步提高DenseNet在ImageNet上的性能

**5. 讨论**

从表面上看，DenseNet与ResNet非常相似：**公式2**与**公式1**唯一的区别在于输入到 的特征图是通过连接（concatenation）而不是求和（summation）组合的。然而，这一看似微小的修改却导致了两种网络架构截然不同的行为。

**模型紧凑性（Model compactness）.** 由于输入的连接操作，DenseNet中任何一层学到的特征图都可以被所有后续层访问。这鼓励了特征在整个网络中的重用，并导致了模型更加紧凑。

**图4**的左图和中图展示了一项实验结果，该实验旨在比较所有DenseNet变体（左图）以及与类似的ResNet架构（中图）的参数效率。我们在C10+数据集上训练了多个不同深度的小型网络，并绘制了它们的测试准确率与网络参数数量的关系图。与其他流行的网络架构（如AlexNet[16]或VGG-net[28]）相比，带有预激活的ResNet在通常实现更好结果的同时使用了更少的参数[12]。因此，我们将DenseNet（）与这种架构进行了比较。DenseNet 的训练设置与上一节保持相同。

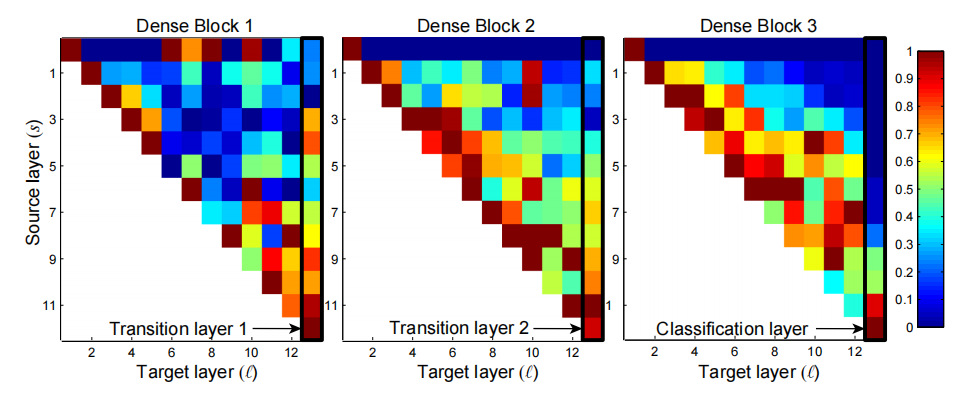
该图显示，DenseNet-BC 始终是 DenseNet 参数效率最高的变体。此外，为了达到相同水平的精度，DenseNet-BC仅需要ResNet约1/3的参数（中图）。这一结果与我们在**图3**中展示的ImageNet结果一致。**图4**的右图显示，一个仅有0.8M可训练参数的DenseNet-BC，其准确率与具有10.2M参数的1001层（预激活）ResNet相当。

**隐式深度监督（Implicit Deep Supervision）.** DenseNet准确率的提高的一个可能解释是，每一层通过较短的连接从损失函数中获得额外的监督。我们可以将DenseNets理解为执行了一种“深度监督”。 深度监督的好处之前已经在深度监督网络（DSN）[20]中得到了体现，这种网络在每个隐藏层都附加了分类器，从而促使中间层学习具有区分性的特征。

DenseNet 以隐式方式执行类似的深度监督：网络顶部的单个分类器通过最多两个或三个过渡层为所有层提供直接监督。然而，DenseNet 的损失函数和梯度要简单得多，因为所有层之间共享相同的损失函数。

**随机连接与确定性连接（Stochastic vs. deterministic connection）.** DenseNet与残差网络的随机深度正则化之间存在一个有趣的联系[13]。在随机深度中，残差网络中的层被随机丢弃，从而在周围层之间创建了直接连接。由于池化层永远不会被丢弃，因此网络最终形成了与DenseNet类似的连接模式：如果所有中间层都被随机丢弃，在相同池化层之间的任意两层之间，存在一个较小的概率直接连接。尽管这两种方法最终存在很大差异，但从DenseNet的角度解释随机深度可能会为我们理解这种正则化方法的成功提供见解。

**特征重用（Feature Reuse）.** 按照设计，DenseNet允许每一层访问其前面所有层的特征图（尽管有时通过过渡层）。我们进行了一项实验，以调查训练后的网络是否利用了这一机会。我们首先在C10+上训练了一个DenseNet，其深度为，增长率为。对于块中的每个卷积层 ，我们计算了与层的连接所分配的平均（绝对）权重。**图5**显示了所有三个密集块的热图。平均绝对权重可以作为卷积层对其前序层依赖性的替代指标。位置为的红点表明，层平均而言大量使用了之前层生成的特征图。从图中可以得出以下几点观察结果：



**图 5**：经过训练的 DenseNet 中卷积层的平均绝对滤波器权重。像素的颜色表示密集块内连接卷积层到的权重的平均范数（通过输入特征图的数量归一化）。由黑色矩形高亮显示的三列对应于两个过渡层和分类层。第一行表示连接到密集块输入层的权重。

1. 所有层在其所在的块内将权重分散到许多输入上。这表明很早的层提取的特征确实被同一密集块中的深层直接使用。
2. 过渡层的权重也分散在其前面密集块的所有层上，表明信息通过很少的间接路径从DenseNet的第一层流向最后一层。
3. 第二和第三个密集块中的层始终对过渡层的输出赋予最少的权重（三角形的最上面一行），表明过渡层输出了许多冗余特征（平均权重较低）。这与DenseNet-BC的强劲结果一致，其中正是这些输出被压缩。
4. 尽管最终分类层（显示在最右侧）也使用了整个密集块的权重，但似乎有向最终特征图集中的趋势，表明网络的后期可能产生了一些更高级别的特征。

**6. 结论**

我们提出了一个新的卷积网络架构，称为**密集卷积网络（Dense Convolutional Network，DenseNet）**。它引入了具有相同特征图大小的任意两层之间的直接连接。我们证明了DenseNet可以自然地扩展到数百层，而不会出现优化困难。在我们的实验中，DenseNet 往往会随着参数数量的增加而不断提高准确性，而不会出现任何性能下降或过度拟合的迹象。在多种设置下，DenseNet在多个极具竞争力的数据集上均取得了最先进的结果。此外，DenseNet在实现最先进的性能时需要的参数和计算量显著更少。由于我们在研究中采用了针对残差网络优化的超参数设置，我们相信通过更细致地调整超参数和学习率计划，DenseNet的准确性可能会进一步提高。

尽管遵循着简单的连接规则，但DenseNet自然地整合了恒等映射、深度监督和多样化深度的特性。它们允许在整个网络中重用特征，因此能够学习更紧凑的模型，并且根据我们的实验，这些模型也更准确。由于其紧凑的内部表示和减少的特征冗余，DenseNet可能成为各种基于卷积特征的计算机视觉任务（例如[4, 5]）的良好特征提取器。我们计划在未来的研究中探索DenseNet的这种特征迁移能力。

**致谢**

论文的部分作者得到了国家科学基金会（NSF）III-1618134、III-1526012、IIS-1149882以及比尔和梅琳达·盖茨基金会的资助。Gao Huang获得了中国博士后科学基金会国际博士后交流计划（International Postdoctoral Exchange Fellowship Program of China Postdoctoral Council）的资助，编号为No.20150015。Zhuang Liu得到了国家重点研发计划（2011CBA00300、2011CBA00301）和国家自然科学基金（61361136003）的支持。我们还要感谢Daniel Sedra、Geoff Pleiss和Yu Sun的许多富有洞察力的讨论。

**参考书目**

[1] C. Cortes, X. Gonzalvo, V. Kuznetsov, M. Mohri, and S. Yang. Adanet: Adaptive structural learning of artiﬁcial neural networks. *arXiv preprint arXiv:1607.01097*, 2016.

[2] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. FeiFei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[3] S. E. Fahlman and C. Lebiere. The cascade-correlation learning architecture. In *NIPS*, 1989.

[4] J. R. Gardner, M. J. Kusner, Y. Li, P. Upchurch, K. Q. Weinberger, and J. E. Hopcroft. Deep manifold traversal: Changing labels with convolutional features. *arXiv preprint arXiv:1511.06421*, 2015.

[5] L. Gatys, A. Ecker, and M. Bethge. A neural algorithm of artistic style. *Nature Communications*, 2015.

[6] X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. In *AISTATS*, 2011.

[7] I. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio. Maxout networks. In *ICML*, 2013.

[8] S. Gross and M. Wilber. Training and investigating residual nets, 2016.

[9] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, 2015.

[10] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*, 2015.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[12] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *ECCV*, 2016.

[13] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger. Deep networks with stochastic depth. In *ECCV*, 2016.

[14] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[15] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. *Tech Report*, 2009.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[17] G. Larsson, M. Maire, and G. Shakhnarovich. Fractalnet: Ultra-deep neural networks without residuals. *arXiv preprint arXiv:1605.07648*, 2016.

[18] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[19] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[20] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu. Deeply-supervised nets. In *AISTATS*, 2015.

[21] Q. Liao and T. Poggio. Bridging the gaps between residual learning, recurrent neural networks and visual cortex. *arXiv preprint arXiv:1604.03640*, 2016.

[22] M. Lin, Q. Chen, and S. Yan. Network in network. In *ICLR*, 2014.

[23] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[24] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning, 2011. In *NIPS Workshop*, 2011.

[25] M. Pezeshki, L. Fan, P. Brakel, A. Courville, and Y. Bengio. Deconstructing the ladder network architecture. In *ICML*, 2016.

[26] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko. Semi-supervised learning with ladder networks. In *NIPS*, 2015.

[27] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio. Fitnets: Hints for thin deep nets. In *ICLR*, 2015.

[28] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*.

[29] P. Sermanet, S. Chintala, and Y. LeCun. Convolutional neural networks applied to house numbers digit classification. In *ICPR*, pages 3288–3291. IEEE, 2012.

[30] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun. Pedestrian detection with unsupervised multi-stage feature learning. In *CVPR*, 2013.

[31] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

[32] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 2014.

[33] R. K. Srivastava, K. Greff, and J. Schmidhuber. Training very deep networks. In *NIPS*, 2015.

[34] I. Sutskever, J. Martens, G. Dahl, and G. Hinton. On the importance of initialization and momentum in deep learning. In *ICML*, 2013.

[35] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

[36] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016.

[37] S. Targ, D. Almeida, and K. Lyman. Resnet in

resnet: Generalizing residual architectures. *arXiv preprint arXiv:1603.08029*, 2016.

[38] J. Wang, Z. Wei, T. Zhang, and W. Zeng. Deeply-fused nets. *arXiv preprint arXiv:1605.07716*, 2016.

[39] B. M. Wilamowski and H. Yu. Neural network learning without backpropagation. *IEEE Transactions on Neural Networks*, 21(11):1793–1803, 2010.

[40] S. Yang and D. Ramanan. Multi-scale recognition with dagcnns. In *ICCV*, 2015.

[41] S. Zagoruyko and N. Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[42] Y. Zhang, K. Lee, and H. Lee. Augmenting supervised neural networks with unsupervised objectives for large-scale image classification. In *ICML*, 2016.