CNN History

Jingru Men

March 12, 2024

Contents

1 概述

对于图形所做的数据分析，如识别、分割等，其发展历程在 2012 年有一个史诗级的进步，就是因为卷积神经网络（Convolutional Netural Network，以下简称 CNN）的引入和同时期硬件 GPU 算力的急速发展。但是 CNN 并不是在 2012 年才提出的，也并非只应用于图形的数据分析，CNN 在 1D 数据处理上的应用和传统的 2D 处理图像，乃至如今 3D 处理拥有上下文关系的体积或视频的应用息息相关。Cnn 网络在不同维度和不同场景有着不同的应用形式。如 2d-cnn 的深度模型拥有大量参数，抽象能力高，且对提取到的特征有区域识别的需求。1d-cnn 的深度模型不需要很深，因为计算复杂度低且参数量相对较少，且要求实时性核可部署在计算能力低的设备中。3d-cnn 的参数量更巨，对特征提取的要求更高，也更注重时间维度和空间维度的上下文关系。Cnn 的发展成熟始于 2d，故在 1d 和 3d 应用中有结构的类比关系和针对场景做出的特别改进。本文以 2d-cnn 的发展和改进原理为基础，介绍 cnn 在不同维度的应用差异和模型结构。

结构：本文在 1 中依照时间线介绍 2d-cnn 的早期发展，2 中为 2015 年后以 resnet 系列模型的改进，3 介绍了其他多种模型的改进思路。4，5 中分别介绍 1d-cnn 和 3d-cnn 在不同场景下的应用，着重介绍医疗图像场景下的应用。(3d 部分择文另写)

2 2d-cnn 的改进和发展

2.1 卷积神经网络的初创时期

在卷积神经网络出现之前，是有一个人工神经网络的时期的。神经网络这个概念出现的很早，早在上世纪 40 年代，随着解剖学和脑神经学的发展，人们就开始考虑仿造大脑的基本结构——神经元来构建人工智能的想法了。

人工神经网络是一种模仿生物神经网络的数学抽象，而人工神经元是其基本单位。1943 年，第一个人工神经元模型的提出者是 McCulloch 和 Pitts[1]，他们的 M-P 神经元模型很简单，如图 1 所示，单个神经元有三个树突接口 [2]，其他神经元的输出由树突触接入，输入的信号经过细胞核处的某种运算（这里使用的是加权求和），如果运算结果大于某一阈值，则激活一个脉冲（动作电势）作为输出，输出也会分为许多小分支携带相同信息作为下一个神经元的树突输入接口。整体数学过程如公式 (1) 所示。如此，一个仿生的人工神经元的数学模型就完成了。这是一个“婴儿”神经元，但它却是后面一切深度学习 CNN 的理论基础和数学基础。由人工神经元组成的神经网络就被称为人工神经网络 ANN（Artificial Neural Network）。1949 年 Danald Olding Hebb 的赫布学习率的出现，说明了学习行为与神经元的联系，机器学习行为的可能性所需的一切零件便都具备了。

$$x_k^l = b_k^l + \sum_{i=1}^{N_{l-1}} w_{ik}^{-1} y_i^{l-1} \quad \text{and} \quad y_k^d = f\left(x_k^l\right) \tag{1}$$

为了解决 M-P 模型内权值固定不变的缺陷，1958 年 Frank Rosenblatt 等人将单个仿生神经元内的权值设置为可更新，根据有监督学习，设置训练样本和误差修正来调整实际权值，使神经元拥有可学习的能力，并将其称之为感知机（perceptron）[3]。式 2 为误差修正公式

$$\begin{cases} \mathrm{w}_i \leftarrow w_i + a(r-y)x_i \\ h \leftarrow h - a(r-y) \end{cases} \tag{2}$$

但是感知机模型仍然太简单，它只能解决线性可分问题，对线性不可分问题束手无策。它对稍复杂的函数都无法拟合，比如异或问题，这使得单层感知机模型遭遇了很大质疑。很快，初期的多层感知机（Feedforward Neural Network 前馈网络，FNN）被提出，整体结构为输入层，中间层和输出层，由于前馈网络的误差修正只能单层修正，故只能修改中间层至输出层了参数。虽然如此，多层感知机仍在一定程度上解决了线性不可分的复杂函数拟合问题，如图 2 所示。为了解决前馈网络只能更新单层权值的问题 [4]，BP 反向传播算法的多层感知机又被提出 [5-12]。多层感知机是一个全连接网络，它的模型结合了梯度的连乘效应连接了不同层的权值更新（梯度下降法），使用一个误差修正改善模型内所有神经元的权值更新。如深度学习三大巨头之一的 Bengio 所说，多层感知机的优势是，K 层感知机比 K-1 层所需的神经元（参数）成指数倍减少。根据 Melanie Mitchell 写的复杂一书中系统科学的概念，简单的单仿生神经元成规模地组合在一起时，由于其数量和结构的复杂性，使得整体神经网络有了宏观的认知行为和泛化能力。后人对感知机在更复杂的神经元模型和激活函数等方面进行了改进 [13-16]。尤其是广义运算感知机 [17][18]，它提出的非线性算子：节点操作、池操作、激活操作和异构网络，允许更广泛的神经元活动，这对后续的神经网络发展具有启发意义。

2D-CNN 的视觉学习在同时期也有了萌芽。1959 年，Hubel 和 Wiesel 对猫的视觉皮层系统进行了研究 [19]，并在描述视觉皮层神经元时提出了感受野的概念，他们最早发现了信息层次化的处理机制，这是视觉神经科学的基础，也是 2D-CNN 发展的理论基础。他们的文章发表在《生理学杂志》上 [19-23]。随着视觉认知科学的发展，Fukushima 和 Miyake 在 1982 年提出了一个自组织分层网络，并将其命名为 “Neocognitron” 新认知机 [24]，它可以不受位置影响地识别一些简单的特征（如形状）。它模仿哺乳动物的视觉系统能力，将相似物体分为同一类别。这有力地说明了神经网络的特征提取能力。这是 2D-CNN 的前身。然而，它的限制在于不能有监督地进行学习任务的训练。1986 年 Rumelhart 和 Hinton 重新发明了反向传播（BP）算法，并将其引入网络 [6]，改进了这一缺陷，并将模型称为神经网络，也就是 NN（Netural Network）。BP 算法使用梯度下降

法迭代更新网络参数（权值和偏差），常用的几种随机梯度优化方法有随机梯度下降 (SGD)，动量 SGD[97]，AdaGrad[98]，RMSProp [99]，Adam[100] 及其变种 [101]。当中间层多于 2 层时，称为 DNN（Deep Neural Networks）。1989 年，leCun 孕育许久的 CNN 名作 Le-Net 即将问世 [25]。

2.2 卷积神经网络的朴素时期

1990 年，由于邮票数字识别的需求，Yann LeCun 制作了一个应用了 BP 算法的 CNN 模型——Le-Net[26] 来处理 MNIST 手写体数字数据库 [27]。Lenet 麻雀虽小，深度学习的基本模块：卷积层，池化层，全连接层却是全的。输入为单通道灰度图，经过两次卷积层 + 池化层，再经过全连接层接 softmax 作为输出。其中第一层滤波器为 5*5*1，共 6 个，产生 6 个特征图。如图 5。

Lenet 模型在 1998 年趋向成熟，它的分类能力以极高的准确率宣告了 2D-CNN 的成功 [28] 和一个时代的降临。但是，由于 CNN 的学习能力是站在大数据量和强计算能力的基础上的，如 MNIST 数据集拥有 7 万张图片和对应数量的标签。所以，理论的先行者要蛰伏等待一个大数据量时代和强计算能力的硬件发展的到来，这一等又是 14 年。

在此期间，由于实际应用的困难，和同时期支持向量机（SVM）[29-33] 和贝叶斯网络（BN）[34][35] 的兴起，参数量大且学习能力强的 CNN 干不过参数量小在中小数据集上易优化的 SVM 和 BN，小数据集造成的过拟合和学习不足让 CNN 在 IMAGENET 上屡屡处于下风。

在深度学习提取特征之前，由于计算能力的不足，是有一个人工设计方法提取特征的时期的。传统的手工特征有颜色特征、几何特征、基于特征点的特征描述子等 [36][37]。人工使用线性判别分析或主成分分析的数学方法，从数据中手动筛选和提取特征信息，这种计算常常是次优的，且计算复杂度较高，实际应用并不现实。对于不能应用于实践的模型，工业界的热情是很低的。

终于，1999 年，NVIDIA 首先提出了 GPU（图形处理器）的概念，其性能在接下来的五年里被提升了上千倍。2006 年，CUDA 发布，它将 GPU 应用于通用计算，解放了 GPU 的高性能并行计算能力。2012 年，Krizhevsky 等人提出的 Alex-Net[38] 踩上 GPU 这阵东风，在 ILSVRC 竞赛上的准确率超出第二名 10%，碾压式地证明了 CNN 的学习和泛化能力。AlexNet 使用了 8 个学习层，其中 5 个为卷积层（使用的卷积核为 11*11，5*5 和三个 3*3），3 个全连接层。卷积层后内嵌池化层（3*3）和局部响应归一化（LRN）层用以优化计算。卷积层依仗 GPU 的计算能力暴力提取感受野内的特征以代替传统的手工特征提取。池化层降低计算量。LRN 层的局部神经元竞争机制有缓解过拟合的作用。全连接层负责分类任务，它将卷积层提取到的特征映射到样本空间，可以使目标不受位移影响，依照特征分类。在细节上，它首次使用了 Relu 作为激活函数加速收敛，使用 Dropout 减少计算量且防止过拟合。AlexNet 的成功十分引人注目。学习能力强泛化能力强的 CNN 终于等到了自己的时代。2D-CNN 以其优秀的性能迅速取代了传统的支持向量机和贝叶斯网

络，成为图像处理的主流工具，走上了高速发展的道路。

2.3 感知机、支持向量机和 CNN 的联系和区别

感知机的可变量是线性式中的权值 w 和偏差 b，数学表达式是 y=wx+b（这条线称为超平面），然后激活函数将低于这个线性式的信息抑制，低于线性式的信息激活，平面可视化为图 3。但是这种分离往往不是最优的，泛化性能不好。一个线性可分数据集的超平面有无数个，如何选取最优的那个呢？SVM 对此做出了改进。为了获得最优泛化性能的超平面，SVM 首先根据 KKT 条件选取具有约束作用的支持向量（靠近边界的学习样本），通过几何间隔最大化的思想，将最优超平面的问题转化为支持向量与超平面欧氏距离最大化，再转化为带有一次项条件的 L2 范数（||w||）的最小化问题，见式 3。SVM 在线性不可分的问题上也给出了自己的回答，它有创见地使用核函数取代映射函数，将学习样本映射在高维特征空间的同时降低计算复杂度，以便在高维求解最优超平面，解决了线性不可分问题。SVM 使用了与传统机器学习（如 PCA、LDA）降维思想完全不同的升维思想，这打开了一个学习空间巨大的新世界，也启发了 CNN 的发展。SVM 是浅层学习优秀的发展，具有坚实的数学理论基础。CNN 则是受到猫的视觉皮层启发，走的是一条深度学习的路子。

我们知道多层感知机在层间是全连接的，如果输入为一张 1000*1000 像素的图片，连接到隐层，每个节点就需要 10^6 个权值参数和 1 个偏置参数参与计算，因为它使用的是全局感受野，每个神经元又相对独立，并不符合视觉图像的习惯（感兴趣区域可能只在一小块区域内），所以多层感知机在处理图片时并不适用。为了摆脱这个困境，人们考虑不用全连接，而是一个区域一个区域地处理图片数据（局部感受野）。参考 SVM 的核函数概念，如果仍是 1000*1000 像素的图片，局部感受野为 10*10，使用一个 10*10 的卷积核与遍历的位置相乘求和（卷积运算），作为提取到的局部特征，那么每个神经元只需要 100 个权值参数和 1 个偏置参数（遍历时权值共享）。每个神经元的二维分布使得局部信息与空间信息同时得以保留。卷积核相当于感知机的权值，在反向传播中可学习更新。在处理多维图像时，如需改变通道数至 n，则取用 n 个与原通道数相同的卷积核，卷积后对应位置相乘求和。如图 4。得知 CNN 的来路，我们可以更坦然地考虑后人对它的改进。

2.4 2d-cnn 的黄金时期

2013 年，Zeiler 和 Fergus 提出了 ZFnet[39]，它在 Alexnet 的基础上使用了更小的卷积核（7*7）和更小的步长以保留更多特征，把 ILSVRC 竞赛的错误率降低到了 11.7%（alexnet 是 15.3%）。它还提出了反卷积将每个卷积层进行可视化，加深了人们对 CNN 学习能力的理解（时至今日 CNN 的学习能力早已超越人们对它的理解方式）。2014 年 ILSVRC 竞赛的冠军是 google 团队的

GoogLenet[40]（L 大写致敬 Lenet）。它考虑的是在 CNN 越来越深的前提下，出现的参数多，计算复杂度大，梯度消失难以优化的问题。它的创新是提出了多尺度卷积的 inception（盗梦空间）模块，如图 5。Inception 的 block 使用了支路结构，支路分别为 1*1，3*3，5*5 的卷积层和 5*5 的 maxpooling 层，输出为各自卷积结果的聚合 concat，相当于对特征图做了多尺度特征融合，解决了模型自动学习使用哪一个卷积核（过滤器）更优的问题。模型在深度和宽度上都得到了提升，而计算成本没有显著上升，这归功于 1*1 的小卷积。模型用 1*1 的卷积层减少了数据的通道数，将参数量压缩到原本的 1/3。模型深度提升有助于非线性表达能力，模型宽度提升提供了更加丰富的特征。为了应对梯度消失的问题，它设计了深浅层 3 个 loss 接口加权求和作为整体 loss。在这些基础上，GoogLenet 做了 22 层，是一个足够深的深网，它把 ILSVRC 竞赛的错误率降低到了 6.7%（人类的错误率为 5.1%）。Inception 模块在其后进行了优化 [40-43](V1-V4)。在 ILSVRC 2014 大赛中，定位任务第一名和识别任务第二名的 VGG 考虑的是网络结构的改善 [44]。它的创新是提出使用小卷积核（3*3）堆叠代替大卷积核的方式可以提升精度，在增加网络深度的同时减少参数量，如式 4。VGG 的深度有两种，16 层的 VGG16 和 19 层的 VGG19。从这里可以看出，追求更深的网络是 CNN 追求更好性能的发展趋势，而深网面对的问题则是参数量大和梯度消失。

GoogLenet 很快在 inception-v2 中采用了 VGG 的想法，Google 将小卷积堆叠代替大卷积引入了 inception，并提出了 BN（Batch Normalization）层 [41]，BN 层标准化了每层的数据分布，容许更大的学习率，使得训练时间大大缩短，取代了 dropout[45][46] 和 LRN 的作用。GoogLenet-v2 在 imagenet 数据集上的成绩达到了 4.9%，超过了人类。有趣的是，GoogLeNetv2 并不是首次在 imagenet 上超越人类的模型。第一个在 imagenet 数据集上超越人类的 CNN 模型 [47] 是微软亚洲研究院（MSRA）的何恺明博士提出的。他做出的改进有二，一个是 PRelu 函数的提出，使得激活函数负半轴也拥有可以学习的函数，二是 kaiming 权值初始化方法。

2.5 权值初始化和 BN

在 BN 出现之前，权值初始化决定了神经网络的初始优化位置，如图 6。正向传播时，神经元的输出会被作为激活函数的输入来进行激活判断。如果神经元的输出不合适，则难以优化（恒为 0 或 1），神经元的输出应当控制在均值为 0，方差为 1 的范围内比较合适。为了使神经元输出控制在这个范围内，如此处神经元输出范围为 N （0，1）正态分布，该神经元的输入有 n 个神经元，则权重矩阵元素初始化就应当为 N （0，1/n），如式 5。在反向传播时，由于梯度的连乘效应（梯度 = 激活层 1 斜率 * 权重 1* 激活层 2 斜率 * 权重。。），权值的过大或过小会造成梯度爆炸或消失。所以梯度应当尽量保持在 1 左右。对求梯度产生影响的有几个因素：激活函数求导（如 sigmoid 函数的梯度最高为 0.25，易梯度消失，而 tanh 函数梯度最高为 1，较为合适）；权重（权重的取用考

虑当前层的输入）。2010 年之前，权值初始化偏向单纯的数学随机选取。常值初始化仅用于偏差 b 的初始化，因为权值的常数初始化会导致神经网络的失效。权值的初始化一般用均匀分布初始化、高斯分布初始化、截断高斯分布初始化随机选取。在此基础上的改进版本为自适应初始化（0 均值，标准差自适应张量 size 的截断高斯分布）。之后，Lecun 提出了 lecun 初始化（期望为 0，标准差自适应前向传播的均匀分布）[48]，如式 6。2010 年，Glorot 和 Bengio 提出的 Xavier 初始化（均值为 0，标准差同时考虑前向和反向传播的正态分布/均匀分布）[49]。他们的背景假设是激活函数在 0 附近近似为恒等映射（线性函数），如 tan 函数（tanh（x）=x）。2012 年 alexnet 中使用非 0 对称 relu 函数后，权值初始化也做出了相应改进，也就是何恺明提出的 he 初始化（均值 0，标准差自适应的高斯分布）[47]，基于 PRelu 激活函数的初始化，如式 6，考虑前向传播和反向传播对方差的放缩系数为常数，故二者等价。PRelu 激活函数下模型收敛较 Xavier 更好。2015 年 Sergey Ioffe 和 Christian Szegedy 提出了批量标准化 BN。BN 考虑，神经网络是由于每层的权值参数不同，导致的每层输入数据分布也会不同，这对学习率和参数初始化的要求就会很高。BN 的原理是，将每层的数据进行以单样本单通道为单位进行小批量规范化，并且为了同时兼顾非线性函数的表示能力，在标准化后使用可单层学习的参数 scale 和 shift 来进行恢复 [41]。在正向传播时，数学表达式如式 7，在反向传播时，对输入 x 梯度计算链式法则，对两参数梯度计算更新参数。BN 减少了数据分布变化引起的内部协变量，又避免了单层规范化可能带来的偏差值爆炸和非线性表达失效的问题。如此，由于数据分布被规范化，正向传播时数据分布移动到激活函数中心非饱和区域加速优化，反向传播时激活函数斜率接近 1 加速收敛。随着 BN 的出现，收敛函数更加平滑，权值初始化设计的问题已经基本被解决。

2.6 大型深网与恒等映射

另一个模型的参数量大的问题，刨析原因，绝大多数的参数和计算量在传统网络的全连接层。所以人们纷纷盯上了这个起分类作用的，要求苛刻的全连接层。首先，一个模型的全连接层的神经元个数是固定的，这限制了模型的输入。为了解决这个问题，何恺明提出了基于金字塔池化的对输入 size 无限制的 SPPNet 网络（该文还提出了卷积复用以加速计算）[50]。但是这样计算量还是下不来，NIN（Network in Network）就直接删掉了全连接层，用 1*1 小卷积作为替代，将参数减少了一个数量级 [51]。

此外，人们发现当深网深到一定程度时，模型的性能逐渐上升至饱和，然后下降，这被称为网络退化问题。网络退化的原因是非线性激活函数造成的不可逆信息损失 [52]，因此深层网络比浅层网络错误率更高，更难以被优化。为了解决这样的问题，需要提出一个能解决“恒等映射”的想法——哪怕更深的层什么都不做，也比变差更好（保留深层提取语义信息的能力和至少不网络退化）。

有了需求，一个满足需求的更大突破到来了。2015 年 ILSVRC 竞赛冠军 Resnet 的成绩为 3.57%（2014 年冠军成绩为 6.7%），模型深度达到了恐怖的 152 层 [53]。如此深网而不造成网络退化的秘诀在于它的数学模型——残差级联，如图 7。如我们所知，“恒等映射” 既然需要什么都不做，那不如直接把网络的浅层跳跃连接到深层。跳跃连接带着浅层的输出与被跳过的层的输出的加和作为深层的输入。在这种短路结构中，需要优化的函数从 H(x) 变成了 F(x)=H(x)-x，也就是所谓残差，残差 F(x) 比输出函数 H(x) 更易优化。如图 7 中的两层残差结构，其数学表达式为式 8。残差结构使用身份映射使跨层连接的维度一致，这在多层剩余函数连接时对网络退化有优势。同时，反向传播时，残差结构对梯度消失问题也有缓解作用，如式 9。残差结构解决了深网的网络退化问题，保证了网络深度带来的抽象理解能力。Resnet 的劣势也显而易见，如果说走跳跃连接 shortcut 更易收敛的话，非 shortcut 之路不传递梯度也是可以的，也就是说，有一些 block 块可能会直接废掉，这一点亟待解决。

3 Resnet 系列网络模型改进

首先是 Resnet 系列，原始的 resnet 网络在输入层有一个 7*7，s=2 的大卷积和 3*3，s=2 的最大池化层，中间层是由 3*3 或 1*1 小卷积组成的残差块的堆叠，输出层是一个全局平均池化和预测类别的全连接。

3.1 resnet-v2

在 resnet-v2 网络中 [54]，何恺明在原版的理论基础上做了组件的位置调换，如图 8。在原有组件分布中，灰色路线的 Relu 在 add 之后，残差块的输出非负，不利于简化优化，所以 Relu 应放在右侧分支层，保持灰色路线 add 后数据分布不变。BN 层应位于 Relu 层之前，因为 mini-batch 中梯度消失几乎不存在，正向传播时 Relu 层可以享受 BN 层的好处。又提出 BN 和 Relu 在权重层（卷积层）之前，作为残差单元的预激活，预激活与后激活效果相当，但 add 之前不应连接 Relu 层，因为 Relu 之后非负会影响数据分布。如此，Resnet-v2 利用组件重组加速了优化。

3.2 Wide ResNet

其后，人们又从增加模型宽度的角度优化残差模型，宽度指特征层的深度，Zagoruyko 在 2016 年提出了 Wide ResNet[55]。它解决了非 shortcut 支路不传递梯度，导致 block 块直接废掉的问题（即 diminishing feature reuse 特征重用减少问题）。解决方法是在每个残差块中添加 dropout 层。

另改进了 CNN 结构，改进有三，一是增加每个残差块的卷积层（非 shortcu 支路卷积层数增加），二是增加卷积层的通道数（通道数 *k，k 是加宽因子），三是增加滤波器尺寸。经过实验，每个残差块的卷积层为两层深度最佳，网络宽度加倍带来的精度改善与网络深度加倍带来的精度改善相当。参数相当的情况下，WRN 在深度缩水 20 多倍（WRN-40-4 与 ResNet1001），仍能保持八倍的训练速度，因为宽度增加更符合 gpu 并行运算的方式。

3.3 ResneXt

同年，Saining 等人提出了 ResneXt 模型 [56]。模型依然是改进残差块，思路与 googlenet 相似，都是走 split-transform-merge（分割-变换-合并）的路线，如图 9，但是与 inception 模块不同的是，它的支路是相同结构的拓扑组成的，并将支路数作为模型深度和宽度外的另一维度，称为“基数”。拓扑作为一个可扩展的维度，减少了人为设计，增加了泛化能力。拓扑结构导致结构的计算复杂度与原 Resnet 几乎相同，但精度提高了，如表 1。滤波器设计借鉴 VGG 的小卷积堆叠思想，继承了 resnet 的瓶颈构架，因为卷积层数大于 3 时拓扑才有效。它还讨论了基数维度的组卷积，即原特征图通道数分为基数维度的组，分别做卷积计算后加和，可以降低计算量。

3.4 Res2Net

2019 年，南开大学、牛津大学和加州大学共同提出了一个 Res2Net 模型 [57]。它解决的是在模型设计的角度上解决多尺度表达能力问题。随着 VGG 提出的小卷积堆叠思想的广泛利用，堆叠过程中的多尺度表达被注意并用于代替架构中特意设计的多尺度特征设计 [58-61]。受架构多尺度特征设计的启发，Res2net 的块结构把 resnet 瓶颈结构内的 3*3 小卷积做了改变，首先参考组卷积 [56] 将特征图依通道数分为 s 层，每层厚度为 w（原通道数 n=s*w），每层特征图使用一个 3*3*w 的小滤波器进行卷积，第一层直连，第二层输出与第三层输入融合后卷积，第三层输出与第四层输入融合后卷积，以此类推。所有层的输出在瓶径口的 1*1 卷积处融合，达到多尺度特征融合的目的，如图 10。Res2net 将 s 层称为一个新维度 “scale”（规模）。Res2net 由于其结构改进，在物体检测和分类方面表现出巨大潜力，模型复杂度也与基线模型大致相等。与分辨率相关的工作还有 [62-67][77]。

3.5 iResnet

2020 年，Cosmin 等人提出了 iResnet，即改进残差网络模型 [68]。它注意到 resnet 网络存在深层信息损失问题，思路是延续 resnet-v2 的进一步优化。它认为 resnet-v2 改善后的结构允许信

息未经处理流入主路，其中 shortcut 的支线信息从未经数据分布的处理和非线性处理，这会限制学习能力。随着 block 叠加，shortcut 的支线信息的数据分布会更加混乱，也不是最优的。对此它做出了三点改进。首先，为了解决信息损失和信息流动问题，iresnet 将网络分段并对段的不同位置提出了开始，中间，结尾三个 block 块的设计，如图 11。其中，开始模块的输入信息已经数据分布处理，开始模块的尾部 BN 在后且不做 relu，可以同时作为中间模块的输入标准化处理，故中间模块的第一个 BN 被消除。结束模块的尾部在主线位置添加了 BN 和 Relu，做为 shortcut 支线信息的数据分布处理和非线性处理，提高学习能力。一个主段结尾的数据分布处理相当于对下一主段的输出做处理。在 Resnet50 中，主要阶段为 4 段，故主线只会经过 4 次 relu，不受深度影响，较小阻碍信号。由于只做了组件上的重组设计，模型复杂度不会增加。第二是 shortcut 的改进。在 resnet 中，shortcut 的维度不匹配可以用身份映射来解决。原始的身份映射可以使用 1*1 小卷积使得维度对齐，但这种对齐是在空间和通道上都匹配的。如果 block 的输入特征图大而需要对齐的空间维度小，例如步长为 2 的小卷积，则会错失四分之三的信息。为了保留信息，iresnet 在 shortcut 上的 1*1 小卷积之前添加了 3*3 的最大池化，使得空间维度上的所有像素都被考虑进去，最大池化的设置应保障卷积的步长为 1，如图 12。在非 shortcut 支路上进行对定位有利的软下采样，在 shortcut 支路上进行对分类有利的硬下采样，形成互补。这个改进也不会使模型复杂度增加。第三是使用分组卷积 [38][56][57][69] 对非 shortcut 支路上的 3*3 卷积进行的改进。分组卷积是将特征图通道分为几层（组），每层（组）使用一个小滤波器进行卷积，减少参数和计算量。由于组卷积的存在，模型得以在 1*1 小卷积后得到通道数更多的特征图，并分配给组卷积操作。通道数的扩长提高了 3*3 卷积的学习能力，如图 13。Iresnet 的多方面改进使得模型性能显著提升，模型深度可达 3002 层且持续优化。

3.6 ResneSt

同年，Zhang 等人引入通道注意力机制，提出了 ResneSt，即分割-注意力网络 [70]。它的块结构在非 shortcut 支路上做了基于组卷积和通道注意力机制的改进。首先输入的特征图为 H*W*C，第一步分组，共 K 组，每组再切片，切 R 片。组内切片的特征图通道数为 C/K/R，做 1*1 小卷积通道数不变，再做 3*3 卷积通道数变为 C/K，把组内所有切片输出作为输入给切片注意力机制，如图 14。切片注意力机制做的工作是，首先将所有切片输出加和，特征图为 H*W*（C/K），求全局平均池化，特征图 1*1*（C/K），连接 DENSE（两个全连接），并做 softmax，将得出的通道权重与原切片通道相乘，如图 15。所有组的输出 concat 后，使用 1*1 小卷积调整通道数到与 block 输入一致，即为非 shortcut 支路上的输出。ResneSt 是对 SENet[71]、SKNet[72]、组卷积 [56] 的有效融合。

3.7 Regnet

可以看出，模型的 block 越来越复杂，设计方案也不可避免地走进了有用就塞进去用的境地，但设计的是否合理却只能用实验来证明。模型深度多深，宽度多宽，该分多少组，都只能靠设计者手工试，模型的设计似乎走进了不合理的瓶颈。

为了解决这个问题，同年，Radosavovic 等人提出了 Regnet 模型 [73]。它考虑的是一个良性网络的宽度、深度和组数等参数应当有一个线性关系之类的设计原则，也即设计空间的设计。作者准备衡量的参数有深度，宽度，瓶颈比和组数。实验设计的主体是一个 Anynet（如 resnet），anynet 由三部分构成，分别是主干，身体，和头，我们保持主干和头部固定，探讨身体部分的参数影响。如图 16，身体由 4 个 stage 构成，每个 stage 包含 di（深度）个 block，Block 的宽度为 wi，Block 瓶颈结构的瓶颈比 bi（输入 block 的通道数/第一个 1*1 小卷积后的通道数），Block 在 3*3 卷积部分使用组卷积则组数为 gi。实验的衡量方法是 EDF（经验误差分布函数）。在设计空间采样 500 个模型参数并训练对应的模型，每个模型参数都进行低计算量（400MF）、低 epoch（10）的大规模实验。实验结果得出：瓶颈比和组宽可以简化计算，但几乎不会影响 EDF。瓶颈比 bi<2 时最佳，组宽 gi>1 时最佳，模型深度和宽度呈线性关系，如图 17。根据线性关系，可以提出一个只有 6 个自由度，大小缩减 10 个数量级的 regnet 模型的设计规则。

4 其他网络模型的改进

Resnet 系列的发展，是与同时期提出的其他模型互相学习的结果，除去 resnet 结构的限制，其他模型也提出了很多新颖的想法。

4.1 inception-v3

2015 年，googlenet 的 inception-v3 被提出 [74]。Inception-v3 继承了 v2 的支路结构和小卷积叠加，但讨论了 3*3 小卷积的进一步分解。一是不对称分解，将 3*3 卷积替换为 1*3 叠加 3*1 卷积，也可以泛化至用 n*1 卷积和 1*n 卷积代替 n*n 卷积，这种分解在中等网格尺寸（12-20）效果最好，可以降低 1/3 的计算量，若叠加的分解卷积走高维路线，则更易训练，如图 18。二是提出了一种高维分解，即特征图 d*d*k 要变换至 d/2*d/2*2k，一般需要卷积和池化两步操作，若先卷积则计算量增加，先池化则遭遇描述瓶颈。若两者兼顾可先将特征图依通道二分，一边走卷积，一边走池化，然后 concat，可以在消除描述瓶颈的同时高效降低计算量。另提出了低层辅助分类器低效可去除。还有一种标签平滑的正则化，通过适当调整 label 改善分类问题中的泛化能力。

4.2 inception-v4 和 inception-resnet

2016 年，由于 tensorflow 对计算内存做了优化，googlenet 也在模型上做了更大规模的改进[75]。首先，它做了两个工作，一是对 inception-v3 模型本身进行了优化，称为 inception-v4。二是汲取了 2015 年里程碑模型 resnet 的 shortcut 结构，称为 inception-resnet。Inception-v4 的模型结构如图 19。模型摆脱分区计算的限制，重构了整体结构。首先在模型输入添加了 stem 主干模块做图片预处理，连接了三种使用不同 inception 模块和两种降低空间维度的 reduction 模块串联，不同模块根据输入空间维度的不同使用了不对称卷积、高维的不对称分解和高维分解的不同设计。Inception-resnet 的结构如图 19。可见与 v4 仅 inception 模块不同。在 inception 模块中添加了 shortcut 连接设计，并缩减 BN 的数量。实验发现添加了残差结构的模型训练速度大大提升。另，作者提出了一种残差缩放的方法来解决过滤器数量过多时尾层参数不稳定的问题。

4.3 SqueezeNet

同年，SqueezeNet[76] 考虑了在移动设备上的配置，提出一种压缩的模型架构，同精度参数压缩 50 倍。挤压网的 block 叫 fire，结构是 1*1 卷积瓶颈接 1*1 卷积 concat3*3 卷积。其中瓶颈通道应小于 concat 后的通道数，构成挤压膨胀的结构。模型整体结构的策略是用 1*1 卷积代替 3*3 卷积降参，保留的 3*3 卷积通道减少降参，和下采样后置保留信息。由上而下过滤器数量逐渐增加。另添加了 shortcut 的支路。

4.4 DRN

2017 年，Fisher 等人将 [77] 中的膨胀卷积的概念和 [53] 的 resnet 结合，提出了膨胀剩余网络 DRN[78]。膨胀卷积是使用隔像素取点卷积的方法，扩大感受野的同时又不丢失分辨率。DRN 在 resnet 的结构基础上使用了膨胀卷积，由于其扩大感受野的能力，在靠后的层组中代替了跨步卷积的下采样层，有效保证了高空间分辨率和提高了模型性能。另，使用膨胀卷积会引起网格伪影，是由于膨胀卷积采样问题导致的本不存在的影像被显示的问题。可以使用卷积代替最大池化，和网络尾端添加逐渐降低膨胀的卷积层并去掉此处残差连接的方法来解决网格伪影问题。

4.5 DenseNet

同年，DenseNet 被提出 [79]。它在模型设计上既不深也不宽，而是借鉴 resnet 提出的 shortcut，将连接和特征重应用到极致。它设计的 dense block 内，每一个卷积层都与其他所有卷积层连接，使得每个层都可以直接从 loss 函数和原始输入中得到梯度，梯度训练变得容易。每个 dense block

内的特征图尺寸一致，方便 concat。由于每层的特征图都作为输入被 concat，导致特征图通道数激增，模型变宽，因此设计了两个结构缩减通道数。一个是瓶颈结构，在 dense block 内每个 3*3 卷积前都设计了一个 1*1 卷积，用来缩减通道数，减少参数。另一个是 dense block 之间的过渡层，使用参数 reduction 控制通道数缩小倍数。Densenet 结构如图 21。Chen 等人在 DPN 中揭示了 resnet 和 densenet 的拓扑意义 [70]。

4.6 CondenseNet

同年，CondenseNet 的提出主要是为了解决 densenet 中的冗余问题 [81]。它在 densenet 的 1*1 卷积中使用了自学习分组卷积（分组可降冗余）。由于直接使用分组卷积会导致精度急剧下降，究其原因是特征通道的不合理人工分配，故提出自学习分配通道的分组卷积方法。自学习分组卷积有一个超参压缩率 C，结构为 C-1 个浓缩 stage 和 1 个优化 stage。通道分组后，在浓缩 stage 阶段，先进行组内 lasso 正则化剪 1/C 的枝，然后卷积。优化 stage 负责优化剪枝后（只剩下 1/C 的参数量）的模型。在应用阶段，使用索引层进行特征重选加普通卷积来应用学习到的通道分配规则。Lasso 正则可以保留有用的维度，去冗余（剪枝）后不会太多影响精度。1*1 卷积后接 3*3 的普通组卷积，达到形式与 densenet 的 block 一致，如图 22。另，Huang 等人注意到模型邻近层的依赖关系更强，故添加了更深层更高增长率（呈指数增长）（dense block 输出通道数增加 k 个通道，k 即为增长率）的方式改进。还添加了 block 之间的 shortcut 连接。

4.7 MobileNet 系列

同年，为了便于模型在移动设备上使用，Howard 等人提出了轻量级的模型 MobileNet-v1[82]。Mobilenet 使用深度卷积，即不同通道不同过滤器卷积，和点卷积，即 1*1 对所有通道卷积，的组合代替普通卷积，如图 23。深度卷积可以负责单通道信息的提取，点态卷积可以融合多通道信息。这种卷积方式的计算量和参数量可以减少 8 倍以上而精度基本不变。另，模型还提出了两个参数以应对更小更快的要求。一个是宽度乘数 ，用于按比例减小通道数，一个是分辨率乘数 ，用于按比例减小特征图大小。

MobileNet-v2 在 v1 的基础上考虑了两个问题 [83]。一个是轻量级模型要求了更薄的通道数，但更薄的通道数后接非线性激活函数会导致兴趣流形的坍塌，为了保留兴趣流形，在通道低维处使用线性分类器 linear bottleneck 代替非线性激活函数 Relu。另一个是与 resnet 瓶颈结构相对应的膨胀结构，也称为倒残差，称为倒残差还因为使用了 shortcut 连接。瓶颈结构是用 1*1 卷积降低通道数，而膨胀结构是用 1*1 卷积增加通道数，以达到有效利用特征的目的。如此，mobileNet-v2

的 block 设计为输入经过 1*1 卷积扩通道数后接 relu，然后使用 v1 提出的 3*3 深度卷积后接 relu，再用 1*1 卷积降维后接 linear bottleneck，如图 24。

MnasNet[84] 是受了 NAS 神经结构搜索 [89] 的启发而诞生的。MnasNet 寻找又小又快又要求准确性的模型结构，所以优化指标侧重于准确性和延迟时间两点，将延迟作为硬约束和软约束两种来扩大搜索范围寻找平衡点。寻找的模型结构考虑整体性（在不同深度使用不同 block）而非最优 block 叠加。对于一个采样模型的搜索过程，训练器获取精度，推理机获取延迟，根据优化目标计算奖励值，根据优化策略更新控制器参数，RNN 控制器循环更新以达到搜索目的。搜索的结果如图 25 所示，其中 d 和 f 与 mobilev1 和 v2 的 block 一致。另，MnasNet 在结构中使用了注意力机制 SE，置于深度卷积之后点卷积之前，+SE 版本的 MnasNet 精度更高。

mobilenetv3[85] 为了应对不同适用性要求提出了两个版本，large 版和 small 版。在做模型探索设计时使用了 MnasNet 做整体结构搜索，分层（局部）搜索则使用 Netadapt（每一步产生降低延迟的新提议和增删权重）。模型在起点和终点做了做了不损失精度的改进，在起点减少一半的滤波器并使用 h-swish 代替 Relu，在终点缩减了三个昂贵的层并前置全局池化。如此，mobilenetv3-large 比 mobilenetv2 快了 15%，准确度高了 3.2%。

4.8 ShuffleNet 系列

同时期，做轻量级网络工作的还有 ShuffleNet。ShuffleNet-v1[86] 的卷积是在组卷积后，分组通道进行重排操作（shuffle）作为下一个组卷积输入。重排方式可学习，重排打破了分组阻塞，可以提高表达能力。Block 结构是加瓶颈的深度卷积为基础，瓶颈 1*1 卷积替换为组卷积加 shuffle，点卷积改为分组点卷积，加 shortcut 结构，如图 26。

ShuffleNet-v2[87] 认为比较运行速度的标准不应当只用计算量 FLOPs 来衡量，还有内存使用量 MAC，并行运算程度等。由此出发，在同计算量情况下，进出通道数为 1：1 时 MAC 最低，速度最快。在同计算量和同输入通道时，组数越高 MAC 占用越多，所以组数不应太多。在一个 block 中使用多个支路（碎片化）如 inception 的多支路会降低并行度，拖慢速度。元素级操作（如 relu，add）MAC 占用率高，会拖慢速度。根据以上对速度新的衡量标准，shufflenetv2 对 v1 的 block 结构做了如下改进。基本块中，使用深度可分卷积加 shortcut 为基础加工，首先输入通道对半劈（channel split，一般设为一半一半），一半走 shortcut，一半走深度可分卷积（分支内通道数 1：1），瓶颈卷积和点卷积均不做组卷积。两分支做 concat，然后做通道混洗。对于下采样 block，输入信息走两个下采样支路，然后 concat 加 shuffle，达到空间下采样而通道数加倍的目的。如图 27。特征重用在临近块之间更强，也符合 condensenet 的建议。

4.9 Xception

2017 年 google 也提出了基于深度可分离卷积的 Xception[88]。Xception 认为普通卷积的过滤器覆盖全通道，深度可分离卷积的过滤器覆盖每一层单通道，而 inception 处于两者之间，输出的特征图通道被三到四个过滤器瓜分。Xception 应用了可分离卷积和剩余连接，设计简洁速度很快，是 inception 系列中的轻量级网络。

4.10 NASNet

同年，google 提出了 NASNet[89]。NASNet（神经结构搜索网）是人为设定整体结构，用以缩小设计空间。RNN 控制器控制超参，超参对应的子模型训练后提供精度，精度为控制器超参提供搜索梯度。设计的最小单位是 block，block 的结构为两个隐藏状态输入、对隐藏状态的运算操作和结合构成，输出为一个隐藏状态。运算操作可选，结合方式可选。B 个 block 组合构成一个 cell。整体结构有两种 cell，一种是输入输出尺寸相同的 cell，另一种是降采样的 cell。Cell 按照人为设定构成整体结构。NASNet 的本质是设计了一个更复杂更合理的 inception 块。NASNet 还提出了一种有效正则化方法，调度路径，即每条路径随训练过程被随机丢弃的概率线性增加。NASNet 使用小数据集训练得到 block 后堆叠移植到大数据集的方法解决 NAS 的大数据集应用问题。

4.11 effnet

2018 年的 effnet 着眼于更小的模型。Freeman 等人认为 shufflenet 和 mobilenet 设计的移动端小网络在更小的模型上面临表现不佳和瓶颈信息损伤两个问题。对更小模型结构优化提出的改进有四点：1 缩小瓶颈参数保留信息。2 将 2*2 最大池化替换为可分离池化置于可分离卷积（1*3，3*1 的可分离卷积）之间。3shufflenet 和 mobilenet 的降采样 block 都是在卷积层做 stride=2 的操作，effnet 使用可分离池化（一维 2*1）用以保护精度，4 不保留第一个层，降低计算量。Effnet 的 block 结构为输入先经过一个 0.5 的瓶颈层，然后做分离的深度卷积的上半部分（1*3），再经过可分离池化的上半部分（1*2），再做分离的深度卷积的下半部分（3*1），点卷积部分用 2*1 卷积核和对应步距代替。Effnet 的 block 设计可以减少计算量的同时保持高于 baseline 的精度。

4.12 Efficientnet

2019 年 Tan 等人提出了 efficientnet。Efficientnet 对模型的深度、宽度、分辨率三个参数进行了线性关系的探索。模型搜索使用了和 MnasNet 相同的准确性和 FLOPs（而非延迟时间，因为不特定硬件设备）双优化指标。为了缩小搜索空间，模型在固定结构的基础上，寻找最佳深度宽度

分辨率组合的基本模型，然后用相同的指数按比例进行扩张（所有层以恒等比例缩放），以满足不同计算资源的需求。其中深度的限制与宽度平方和清晰度平方的限制等量，这是由计算量决定的。efficientnet 在精度提升和参数减少（速度提升）方面成绩都很好。

4.13 SENet 和 SKNet

2019 年的 SENet（最后一届 2017imagenet 冠军）是关注通道间联系，即重视更有用的通道。一般的卷积网络对所有通道无差别对待，而 SENet 的注意力网络是提取每一个通道的全局平均值，对每个值做权值学习（通常为带非线性层的两个全连接），得到的值可以看作反映通道相关性的权值，再与原通道值分别对应相乘，如图 28。达到增强相关性强的通道，抑制相关性弱的通道的目的。取全局平均值的阶段称之为挤压，经过权值学习的数乘回原通道称为激发。整体的计算量增加极其微小，但准确性提升很大，有效且可移植性好。

2020 年 SKNet 在 SENet 的基础上，考虑了卷积核的侧重选择。输入特征图分两路分别做 3*3 和 5*5 卷积（通道数不变），卷积后 add 到一起作为综合特征图，对综合特征图做 SE 处理，将一维结果 softmax 后分别与 3*3 卷积后的特征图和 5*5 卷积后的特征图相乘，然后各自赋予权重 add 到一起作为输出，如图 29。在反向传播时不同卷积核权重可学习。SKNet 在复杂度较低的模型中更有优势。

4.14 RepVGG

2021 年 RepVGG 提出，整体构建只有 3*3 卷积和 Relu 构成，主要工作在多分支拓扑上。根据 shufflenetv2 的建议，多分支模型会导致 MAC（内存使用量）高而拖慢速度，分支融合时内存使用峰值很高，多分支限制了 size 的灵活性也限制了剪枝功能的应用。但多分支模型又会使训练收敛更快。所以 RepVGG 考虑在训练时使用平面拓扑（也可看作单层多分支）结构使之收敛更快，在推理时使用类似 VGG 的单分支结构。推理模型的参数是用训练模型的参数线性组合得到的，因为单层多分支 y=x+g(x)+f(x) 是可以等效于 y=h(x) 的。每个分支 add 之前都做了 BN。线性组合的基本原理是将 1*1 卷积核填充至 3*3 卷积核（x 相当于单位矩阵的 1*1 卷积），三核相加作为复合核，bias 也为三个 bias 加和。如此，多分支结构到单分支结构完成等效表达。RepVGG 在速度和精度上达到了良好权衡。

4.15 Transformer

CNN 是从浅层卷积拿到局部信息，到深层卷积拿到全局信息的逻辑来处理信息的。另外还有一种编码器 Transformer 从一开始就用注意力机制拿到全局信息，同时感受野逐渐扩大（这一点与 CNN 相同）。这种方式的上下文信息能力更强，是一种新的编码器（特征提取器）思路。就像 CNN 一开始没有用全局感受野的道理一样，它难以学习，但现在的数据库和计算能力允许这种尝试。Transformer 有可能成为 CNN 的有力竞争者。

Transformer 最初是在是由 Vaswani 等人在 NLP 中被提出，很快被用在视觉处理。主结构为纯 transformer 的模型如 ViT[95]。

ViT 的想法，为了将 H*W*C 的图片制作为一维序列，故将图片依空间维度分为 P*P*C 的小块，一维序列长度为 H*W/（P*P）。我们称 P*P*C 为 D 维，一维序列长度为 N，也可理解为原图 H*W*C 被转化为 N*D。使用位置 embed 为每个 D 维小块添加编号（一维位置信息）。在编码前位添加一个无语义信息的人为定义的块（可学习）作为编码首位，用来标识整体客观的语义信息，即（N+1）*D。数据有了，将一维数据正则化后送入多头注意力机制，做一个剩余连接，再做正则化后送入多层感知机，再做一个剩余连接，构成一个编码器 block，结构如图 30 右。多个 block 串联，最后做一个多层感知机的分类头。

其中多头注意力机制是由自注意力机制改进而来。自注意力机制可以理解为首先对每一个一维输入乘以权值来表征每个输入的重要性（图中粉色部分），然后乘以三个不同权值作为三个备用（q，k，v）。其中 q 取出与所有输入的 k 做点乘，得到的数做 softmax 后就是权重，乘以 v 得到当前输入的输出 b（这个输出包含了全部输入的信息）。多头注意力机制是将（q，k，v）再分头，各自工作与自注意力机制一样，但得到多个 b，给多个 b 加权求和得到一个 b。多头注意力机制可以关注不同维度的信息，提升性能。在 attention 中不同位置平等，所以乘以权值的输入应当加一个一维位置信息（图中紫色部分），如图 30 左。

关于 transformer 应用于中高端的视觉任务和应用于低端的视觉任务，详见 [96]，此处不做讨论。

5 1D-CNN 的应用

5.1 1D-CNN 在心电图和损伤检测上的应用

Cnn 是将特征提取和分类任务在同一系统中完成，如果将二维信息替换为一维信息，也可以用类似结构保留功能，一维卷积神经网络的发展与二维卷积神经网络的发展息息相关。与二维卷积类

比，一维信息的卷积应当是一维卷积核与一维信息的卷积运算，在如何选取卷积信息和卷积核范围方面不同领域有不同选择。在处理一维信息初期提出的思路是简单的将一维信息整形为二维信息，成为震动图像 [102][103]，整形后的矩阵可以直接送入传统的二维 cnn 模型中训练。然而这种简单直接的方法计算复杂度太高无法应用于低内存设备，且这种方法要求大量的标记数据，而大规模的一维数据集标记难以实现。为了解决这个问题，2015 年 Kiranyaz 等人提出了一个 1D-CNN 模型，来解决特定病人心电图信号的问题检测 [104-105]。整个系统有三个卷积层和两个 MLP（多层神经网络）层。它在卷积层中使用滤波器对原始一维数据进行卷积运算得到一维特征图，再通过激活函数和池化层实现特征提取，如图 19。输入层的维度变化通过池化层自适应调整，使用偏置参数作为数据分布的调整。这种 1D-CNN 很快应用于各种一维信号如心电图搏动的早期心律失常检测 [104-106]、结构损伤检测 [107-111]、大功率发动机故障监测 [112] 和大功率电路的实时监测 [113]、轴承的损伤检测 [114-117]。2017 年，随着二维神经网络的发展，张等人针对滚动轴承震动监测的一维信息制作了一个更成熟的端到端 1D-CNN 模型，如图 20。原始信息输入到模型中先进行归一化，然后经过 6 个卷积层 +BN 层 + 池化层进行特征提取，一个全连接层 +BN 层分类，输出层节点数为类别数，进行 softmax 计算。该模型中用 BN 取代了原偏置参数。为了达到抗干扰的目的，第一个大核卷积层进行 dropout 操作，小批量训练和局部最大池化层确保了模型的稳定性。在实际应用中，训练数据可能并不会很多，而且一维信息理论上比二维信息具有更低的计算复杂度，更少的参数和更浅的模型深度。所以 1d-cnn 的结构不会很深。在结构损伤检测中，同样可以使用 1d-cnn[118]。2018 年 Khodabandehlou 等人做了一个类 cnn 结构做混凝土桥梁的损伤等级分类 [120]。2019 年 yu 等人做了一个土木结构损伤检测的 1d-cnn[119]。该模型使用基于震动的方法获得一维损伤信息，输入数据为信号长度 q* 结构层数 n 组成的时序信号矩阵的快速傅里叶变换（频域表示，因为频域特征更明显），通常信号长度 q 很长。为了应对输入信号长度长和鲁棒性的要求，第一层卷积使用大卷积核（1000*1）。模型有三个卷积 +BN+ 激活层 + 池化层，两个全连接层 +BN 连接输出层，如图 22。池化层较小，保留更精细的信息。后层卷积使用小卷积核，计算效率更高。学习层后接 BN 提高收敛效率。针对标记数据量少的问题，Avci 等人在 [121] 和 Abdeljaber 等人在 [122] 中提出了改进方法。

5.2 1d-cnn 在语音识别上的应用

在声学方面 cnn 的应用先驱是 [123-124]，他们使用了时间窗口的卷积运算，并进行了简单的分类，如性别，但非 cnn 结构。[125] 在 LVCSR 工作中提出了多层卷积的可能，并讨论了最优学习层数。真正使用到 1d-cnn 结构的是 [126]，它的工作是自动语音识别（ASR）。它提出一个基于有限权重共享的 cnn 方案，类似于卷积核的权重共享。输入使用的是原始信号的 MFSC 特征，每

帧 40 个对数能量系数。输入信息的结构为原声谱，声谱的一阶导数，声谱的二阶导数三通道组成。输入窗口为 15 帧，MFSC 特征为 40 维，相当于一个通道数为 3 大小为 15*40 的二维图，或 3*15 个长为 40 的一维特征图，即频率轴维度划分，如图 21。在一维特征图上使用一维卷积，每个一维特征图与一个一维卷积核做卷积运算，这个卷积核在所有一维特征图上共享。取滤波器维度为 40，即滤波器大小为核宽 *40。有限权重共享可以提高模型的鲁棒性、提高精度的同时，也带来了巨量的参数量。随着 2d-cnn 的发展，更多 cnn 用于 ASR 的方法被提出 [127-130]。输入一般选用的是频率-时间分布（频谱图）。多层卷积可以得到更长的上下文窗口和更抽象的信息。

6 3d-cnn 的应用

6.1 3d-cnn 的应用

2015 年 Ronneberger 等人提出了 U-net[131]，2016 年 Cicek 等人提出了 3D U-net[132]，是 3D-cnn 的开端。

U-net 是将传统 FCN 全卷积网络添加特征通道构成的，在数据增强方面做了模拟生物的变形增强。3D U-net 主要是将 2d 卷积转换为 3d 卷积，模型思路与 unet 基本一致。

V-Net[133] 是在 unet 基础上添加了剩余链接的 3d 版本。(3d 部分择文另写)

6.2 3d-cnn 在视频中的应用

(3d 部分择文另写)

References

[1] S.M. Warren, P. Walter, A logical calculus of the ideas immanent in nervous activity, Bull. Math. Biophys. 5 (1943) 115–133.

[2] A. Cichoki, R. Unbehauen, Neural Networks for Optimization and Signal Processing, thirrd ed., 1994.

[3] F. Rosenblatt, The perceptron: a probabilistic model for information storage and organization in the brain, Psychol. Rev. 65 (1958) 386.

[4] M. Minsky and S. Papert, “Perceptrons,” MIT Press, Cambridge, 1969.

[5] Paul J. Werbos. Beyond Regression: New Tools for Prediction and analysis in the Behavioral Sciences. PhD thesis, Harvard University, 1974.

[6] Rumelhart D E, Hinton G E, Williams R J. Learning representations by back-propagating errors[J]. nature, 1986, 323(6088): 533.

[7] S. Amari. Backpropagation and stochastic gradient descent method. Neuro-computing, 5(4 - 5):185 –196, 1993. 2.1.5

[8] M.Y. Mashor, Hybrid multilayered perceptron networks, Int. J. Syst. Sci. 31 (2000) 771–785, https://doi.org/10.1080/00207720050030815.

[9] J.P. Resop, A Comparison of Artifificial Neural Networks and Statistical Regression with Biological Resources Applications, 2006.

[10] T. Ince, S. Kiranyaz, J. Pulkkinen, M. Gabbouj, Evaluation of global and local training techniques over feed-forward neural network architecture spaces for computer-aided medical diagnosis, Expert Syst. Appl. 37 (2010) 8450–8461, https://doi.org/10.1016/j.eswa.2010.05.033.

[11] T.W. Rauber, K. Berns, Kernel multilayer perceptron, in: Proc. - 24th SIBGRAPI Conf. Graph. Patterns Images, 2011 pp. 337–343. https://doi.org/ 10.1109/SIBGRAPI.2011.21.

[12] H. Ogai, B. Bhattacharya, Pipe Inspection Robots for Structural Health and Condition Monitoring, 2018. https://doi.org/10.1007/978-81-322-3751-8.

[13] G. Zhou, Y . Zhou, H. Huang, and Z. Tang, "Functional networks and applications: A survey," Neurocomputing, vol. 335, pp. 384–399, 2019.

[14] S. Qian, H. Liu, C. Liu, S. Wu, and H. San Wong, "Adaptive activation functions in convolutional neural networks," Neurocomputing, vol. 272, pp. 204–212, 2018.

[15] X. Jiang, Y . Pang, X. Li, J. Pan, and Y . Xie, "Deep neural networks with elastic rectified linear units for object recognition," Neurocomputing, vol. 275, pp. 1132–1139, 2018.

[16] F. Fan and G. Wang, "Universal approximation with quadratic deep networks," arXiv preprint arXiv:1808.00098, 2018.

[17] S. Kiranyaz, T. Ince, A. Iosifififidis, Moncef Gabbouj, Progressive operational perceptrons, Neurocomputing (2016), https://doi.org/10.1016/j. neucom.2016.10.044.

[18] S. Kiranyaz, T. Ince, A. Iosifififidis, M. Gabbouj, Generalized model of biological neural networks: progressive operational perceptrons, in: Proc. Int. Jt. Conf. Neural Networks. 2017–May (2017) 2477–2485. https://doi.org/10.1109/IJCNN.2017.7966157.

[19] D.H. Hubel, T.N. Wiesel, Receptive fifields of single neurones in the cat' s striate cortex, J. Physiol. 148 (1959) 574–591.

[20] D.H. Hubel, Single unit activity in lateral geniculate body and optic tract of unrestrained cats, J. Physiol. 150 (1960) 91–104, https://doi.org/10.1113/

jphysiol.1960.sp006375.

[21] D.H. Hubel, T.N. Wiesel, Receptive fifields, binocular interaction and functional architecture in the cat' s visual cortex, J. Physiol. 160 (1962) 106–154.

[22] D.H. Hubel, T.N. Wiesel, Receptive fifields of cells in striate cortex of very young, visually inexperienced kittens, J. Neurophysiol. 26 (1963) 994–1002.

[23] D.H. Hubel, T.N. Wiesel, Receptive fifields and functional architecture of monkey striate cortex, J. Physiol. 195 (1968) 215–243.

[24] K. Fukushima, S. Miyake, Neocognitron: a new algorithm for pattern recognition tolerant of deformations and shifts in position, Pattern Recognit. 15

(1982) 455–469, https://doi.org/10.1016/0031-3203(82)90024-3.

[25] Lecun, Y., Boser, B.E., Denker, J.S., et al. (1989) Backpropagation Applied to Handwritten Zip Code Recognition. Neural Computation, 1, 541-551.

[26] Y. LeCun, B. Boser, J.S. Denker, R.E. Howard, W. Habbard, L.D. Jackel, Handwritten digit recognition with a back-propagation network, Adv. Neural Inf. Process. Syst. (1990) 396–404, https://doi.org/10.1111/dsu.12130.

[27] L. Yann, C. Corinna, B. Chris, MNIST Handwritten Digit Database, New York Univ, 2018.

[28] Y. LeCun, Gradient Based Learning Applied To Document Recognition, 1998, pp. 1–46.

[29] Vapnik, V.N. and Lerner, A.Y., 1963. Recognition of patterns with help of generalized portraits. Avtomat. i Telemekh, 24(6), pp.774-780.

[30] Vapnik, V. and Chervonenkis, A., 1964. A note on class of perceptron. Automation and Remote Control, 24.

[31] Smith, F.W., 1968. Pattern classifier design by linear programming. IEEE Transactions on Computers, 100(4), pp.367-372.

[32] Boser, B.E., Guyon, I.M. and Vapnik, V.N. (1992) A Training Algorithm for Optimal Margin Classifiers. Proceedings of the 5th Annual Workshop on Computational Learning Theory (COLT' 92), Pittsburgh, 27-29 July 1992, 144-152.

[33] Vapnik, V.N. (1995) The Nature of Statistical Learning Theory. Springer-Verlag, New York.

[34] Pearl J. Probabilistic Reasoning in Intelligent Systems [M]. Morgan Kaufinann: Network of Plausible Inference, 1988: 1-86

[35] Cooper G F, Herskovits E.A Bayesian Method for the Induction of Probabilistic Networks from Data[J].Machine Learning, 1992, 9:309-347.

[36] Paul Viola,Michael J. Jones. Robust Real-Time Face Detection[J]. International Journal of Computer Vision,2004,57(2).

[37] Dalal N , Triggs B . Histograms of Oriented Gradients for Human Detection[C]// IEEE Computer Society Conference on Computer Vision & Pattern Recognition. IEEE, 2005.

[38] Krizhevsky, A., Sutskever, I. and Hinton, G.E. (2012) ImageNet Classification with Deep Convolutional Neural Networks. Proceedings of the 25th International Conference on Neural Information Processing Systems, Lake Tahoe, 3-6 December 2012, 1097-1105.

[39] Zeiler, M.D. and Fergus, R. (2013) Visualizing and Understanding Convolutional Networks. arXiv preprint arXiv:1311.2901

[40] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[J]. arXiv preprint arXiv:1409.4842, 2014.

[41] Ioffe S , Szegedy C . Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift[J]. 2015.

[42] Szegedy C, Vanhoucke V, Ioffe S, et al Rethinking the Inception Architecture for Computer Vision[J]. 2015:2818-2826.

[43] Szegedy C, Ioffe S, Vanhoucke V, et al. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning[J]. 2016.

[44] Simonyan K , Zisserman A . Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer Science, 2014.

[45] Hinton G E , Srivastava N , Krizhevsky A , et al. Improving neural networks by preventing co-adaptation of feature detectors[J]. Computer Science, 2012, 3(4):págs. 212-223.

[46] Dropout: A simple way to prevent neural networks from overfitting (2014), N. Srivastava et al.

[47] He K , Zhang X , Ren S , et al. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification[C]// CVPR. IEEE Computer Society, 2015.

[48] Lecun Y , Bottou L , Orr G B , et al. Efficient BackProp[M]. Springer Berlin Heidelberg, 1998.

[49] Glorot X , Bengio Y . Understanding the difficulty of training deep feedforward neural networks[J]. Journal of Machine Learning Research, 2010, 9:249-256.

[50] He K , Zhang X , Ren S , et al. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition[J]. IEEE Transactions on Pattern analysis and Machine Intelligence, 2014, 37(9):1904-16.

[51] Lin M, Chen Q, Yan S. Network in network[J]. arXiv preprint arXiv:1312.4400, 2013.

[52] Sandler M , Howard A , Zhu M , et al. MobileNetV2: Inverted Residuals and Linear Bottlenecks[C]// 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2018.

[53] He K , Zhang X , Ren S , et al. Deep Residual Learning for Image Recognition[C]// IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2016.

[54] He K , Zhang X , Ren S , et al. Identity Mappings in Deep Residual Networks[J]. 2016.

[55] Zagoruyko S , Komodakis N . Wide Residual Networks[J]. 2016.

[56] Xie S , Girshick R , Dollár, Piotr, et al. Aggregated Residual Transformations for Deep Neural Networks[J]. 2016.

[57] Gao Shanghua,Cheng Ming-Ming,Zhao Kai,Zhang Xin-Yu,Yang Ming-Hsuan,Torr Philip H S. Res2Net: A New Multi-scale Backbone Architecture.[J]. IEEE transactions on pattern analysis and machine intelligence,2019.

[58] E. H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt, and J. M. Ogden. Pyramid methods in image processing. RCA engineer, 1984.

[59] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In ECCV. 2014.

[60] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed. SSD: Single shot multibox detector. In ECCV, 2016.

[61] Lin T Y , Dollár, Piotr, Girshick R , et al. Feature Pyramid Networks for Object Detection[J]. 2016.

[62] C.-F. R. Chen, Q. Fan, N. Mallinar, T. Sercu, and R. Feris. Big-Little Net: An Efficient Multi-Scale Feature Representation for Visual and Speech Recognition. In Int. Conf. Mach. Learn., 2019.

[63] Y . Chen, H. Fang, B. Xu, Z. Yan, Y . Kalantidis, M. Rohrbach,S. Yan, and J. Feng. Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution. In Int.Conf. Comput. Vis., 2019.

[64] B. Cheng, R. Xiao, J. Wang, T. Huang, and L. Zhang. High frequencyresidual learning for multi-scale image classification. In Brit. Mach.Vis. Conf., 2019.

[65] K. Sun, B. Xiao, D. Liu, and J. Wang. Deep high-resolution representation learning for human pose estimation. In IEEE Conf. Comput. Vis. Pattern Recog., pages 5693–5703, 2019.

[66] K. Sun, Y . Zhao, B. Jiang, T. Cheng, B. Xiao, D. Liu, Y . Mu, X. Wang,W. Liu, and J. Wang. High-resolution representations for labelingpixels and regions. CoRR, abs/1904.04514, 2019.

[67] F. Y u, D. Wang, E. Shelhamer, and T. Darrell. Deep layer aggregation. In IEEE Conf. Comput. Vis. Pattern Recog., pages 2403–2412, 2018.

[68] Duta I C , Liu L , Zhu F , et al. Improved Residual Networks for Image and Video Recognition[J]. 2020.

[69] Ioannou Y , Robertson D , Cipolla R , et al. Deep Roots: Improving CNN Efficiency with Hierarchical Filter Groups[J]. 2017.

[70] Zhang H , Wu C , Zhang Z , et al. ResNeSt: Split-Attention Networks[J]. 2020.

[71] Jie, Hu, Li,et al. Squeeze-and-Excitation Networks.[J]. IEEE transactions on pattern analysis and machine intelligence, 2019.

[72] Li X , Wang W , Hu X , et al. Selective Kernel Networks[J]. IEEE, 2019.

[73] Radosavovic I , Kosaraju R P , Girshick R , et al. Designing Network Design Spaces[J]. IEEE, 2020.

[74] Szegedy C , Vanhoucke V , Ioffe S , et al. Rethinking the Inception Architecture for Computer Vision[J]. IEEE Computer Society, 2015.

[75] Szegedy C , Ioffe S , Vanhoucke V , et al. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. 2016.

[76] Iandola F N , Han S , Moskewicz M W , et al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size[J]. 2016.

[77] Yu F , Koltun V . Multi-Scale Context Aggregation by Dilated Convolutions[J]. 2015.

[78] Yu F , Koltun V , Funkhouser T . Dilated Residual Networks[J]. IEEE Computer Society, 2017.

[79] Gao H , Zhuang L , Maaten L V D , et al. Densely Connected Convolutional Networks[J]. Computer Era, 2017.

[80] Dual Path Network

[81] Chen Y , Li J , Xiao H , et al. Dual Path Networks. Curran Associates Inc. 2017.

[82] Huang G , Liu S , Laurens V , et al. CondenseNet: An Efficient DenseNet using Learned Group Convolutions. 2017.

[83] Howard A G , Zhu M , Chen B , et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications[J]. 2017.

[84] Sandler M , Howard A , Zhu M , et al. Inverted Residuals and Linear Bottlenecks: Mobile Networks for Classification, Detection and Segmentation[J]. 2018.

[85] Tan M , Chen B , Pang R , et al. MnasNet: Platform-Aware Neural Architecture Search for Mobile[J]. 2018.

[86] Howard A , Sandler M , Chu G , et al. Searching for MobileNetV3[J]. 2019.

[87] Zhang X , Zhou X , Lin M , et al. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices[J]. 2017.

[88] Ma N , Zhang X , Zheng H T , et al. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design[J]. Springer, Cham, 2018.

[89] Chollet F . Xception: Deep Learning with Depthwise Separable Convolutions[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2017.

[90] Zoph B , Vasudevan V , Shlens J , et al. Learning Transferable Architectures for Scalable Image Recognition[J]. 2017.

[91] Freeman I , Roese-Koerner L , Kummert A . EffNet: An Efficient Structure for Convolutional Neural Networks[J]. IEEE, 2018.

[92] Tan M , Le Q V . EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks[J]. 2019.

[93] Squeeze-and-Excitation Networks. IEEE transactions on pattern analysis and machine intelligence, 2019.

[94] Li X , Wang W , Hu X , et al. Selective Kernel Networks[C]// 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2020.

[95] Ding X , Zhang X , Ma N , et al. RepVGG: Making VGG-style ConvNets Great Again[J]. 2021.

[96] Dosovitskiy A , Beyer L , Kolesnikov A , et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale[J]. 2020.

[97] Han K , Wang Y , Chen H , et al. A Survey on Visual Transformer[J]. 2020.

[98] N. Qian, On the momentum term in gradient descent learning algorithms, Neural Networks 12 (1999) 145–151, https://doi.org/10.1016/S0893-6080(98)00116-6.

[99] J. Duchi, E. Hazan, Y. Singer, Adaptive subgradient methods for online learning and stochastic optimization, COLT 2010 - 23rd Conf. Learn. Theory(2010) 257–269.

[100] T. Tieleman, G. Hinton, Lecture 6.5 - RMSProp, Neural Networks for Machine Learning | Coursera, (n.d.).

[101] K. Diederik, J.L. Ba, ADAM: a method for stochastic optimization, AIP Conf. Proc. 1631 (2014) 58–62, https://doi.org/10.1063/1.4902458.

[102] S. Ruder, An overview of gradient descent optimization algorithms, 2016.

[103] O. Janssens, V. Slavkovikj, B. Vervisch, K. Stockman, M. Loccufier, S. Verstockt, R. Van de Walle, S. Van Hoecke, Convolutional neural network based fault detection for rotating machinery, J. Sound Vib. 377 (2016) 331–345, https://doi.org/10.1016/j.jsv.2016.05.027.

[104] Z. Wei, P. Gaoliang, L. Chuanhao, Bearings Fault Diagnosis Based on Convolutional Neural Networks with 2- D Representation of Vibration Signals as Input, 13001 (2017) 1–5.

[105] S. Kiranyaz, T. Ince, R. Hamila, M. Gabbouj, Convolutional Neural Networks for patient-specific ECG classification, in: Proc. Annu. Int. Conf. IEEE Eng.

Med. Biol. Soc. EMBS, 2015. https://doi.org/10.1109/EMBC.2015.7318926.

[106] S. Kiranyaz, T. Ince, M. Gabbouj, Real-time patient-specific ECG classification by 1-D convolutional neural networks, IEEE Trans. Biomed. Eng. 63

(2016) 664–675, https://doi.org/10.1109/TBME.2015.2468589.

[107] S. Kiranyaz, T. Ince, M. Gabbouj, Personalized monitoring and advance warning system for cardiac arrhythmias, Sci. Rep. 7 (2017), https://doi.org/10.1038/s41598-017-09544-z.

[108] O. Avci, O. Abdeljaber, S. Kiranyaz, M. Hussein, D.J. Inman, Wireless and real-time structural damage detection: a novel decentralized method for wireless sensor networks, J. Sound Vib. (2018).

[109] O. Avci, O. Abdeljaber, S. Kiranyaz, D. Inman, Structural damage detection in real time: implementation of 1D convolutional neural networks for SHM applications, in: C. Niezrecki (Ed.), Struct. Heal. Monit. Damage Detect. Vol. 7 Proc. 35th IMAC, A Conf. Expo. Struct. Dyn. 2017, Springer International Publishing, Cham, 2017, pp. 49–54.

[110] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, D.J. Inman, Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks, J. Sound Vib. 388 (2017), https://doi.org/10.1016/j.jsv.2016.10.043.

[111] O. Avci, O. Abdeljaber, S. Kiranyaz, B. Boashash, H. Sodano, D.J. Inman, Efficiency Validation of One Dimensional Convolutional Neural Networks for Structural Damage Detection Using a SHM Benchmark Data, 2018.

[112] O. Abdeljaber, O. Avci, M.S. Kiranyaz, B. Boashash, H. Sodano, D.J. Inman, 1-D CNNs for structural damage detection: verification on a structural health monitoring benchmark data,Neurocomputing(2017),https://doi.org/10.1016/j.neucom.2017.09.069.

[113] T. Ince, S. Kiranyaz, L. Eren, M. Askar, M. Gabbouj, Real-time motor fault detection by 1-D convolutional neural networks, IEEE Trans. Ind. Electron. 63(2016)7067–7075, https://doi.org/10.1109/TIE.2016.2582729.

[114] S. Kiranyaz, A. Gastli, L. Ben-Brahim, N. Alemadi, M. Gabbouj, Real-time fault detection and identification for MMC using 1D convolutional neural

networks, IEEE Trans. Ind. Electron. (2018), https://doi.org/10.1109/TIE.2018.2833045.

[115] O. Abdeljaber, S. Sassi, O. Avci, S. Kiranyaz, I. Abulrahman, M. Gabbouj, Fault detection and severity identification of ball bearings by online condition

monitoring, IEEE Trans. Ind. Electron. (2018).

[116] L. Eren, T. Ince, S. Kiranyaz, A generic intelligent bearing fault diagnosis system using compact adaptive 1D CNN classifier, J. Signal Process. Syst. 91(2019) 179–189, https://doi.org/10.1007/s11265-018-1378-3.

[117] L. Eren, Bearing fault detection by one-dimensional convolutional neural networks, Math. Probl. Eng. 2017 (2017), https://doi.org/10.1155/2017/8617315.

[118] W. Zhang, C. Li, G. Peng, Y. Chen, Z. Zhang, A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load, Mech. Syst. Signal Process. 100 (2018) 439–453, https://doi.org/10.1016/j.ymssp.2017.06.022.

[119] O Avci, O Abdeljaber, S Kiranyaz, M Hussein, M Gabbouj, D Inman, A Review of Vibration-Based Damage Detection in Civil Structures: From Traditional Methods to Machine Learning and Deep Learning Applications, Mechanical Systems and Signal Processing 147 (2021) 107077, https://doi.org/10.1016/j.ymssp.2020.107077.

[120] Y. Yu, C. Wang, X. Gu, J. Li, A novel deep learning-based method for damage identification of smart building structures, Struct. Heal. Monit. 18 (2019)143–163, https://doi.org/10.1177/1475921718804132.

[121] H. Khodabandehlou, G. Pekcan, M.S. Fadali, Vibration-based structural condition assessment using convolution neural networks, Struct. Control Heal.Monit. (2018), https://doi.org/10.1002/stc.2308.

[122] O. Avci, O. Abdeljaber, S. Kiranyaz, B. Boashash, H. Sodano, D.J. Inman, Efficiency Validation of One Dimensional Convolutional Neural Networks for Structural Damage Detection Using a SHM Benchmark Data, 2018.

[123] O. Abdeljaber, O. Avci, M.S. Kiranyaz, B. Boashash, H. Sodano, D.J. Inman, 1-D CNNs for structural damage detection: verification on a structural health monitoring benchmark data, Neurocomputing (2017), https://doi.org/10.1016/j.neucom.2017.09.069.

[124] H. Lee, L. Yan, P. Pham, A.Y. Ng, Unsupervised feature learning for audio classification using convolutional deep belief networks, in: Adv. Neural Inf.

Process. Syst. 22 - Proc. 2009 Conf., 2009, pp. 1096–1104.

[125] D. Hau, K. Chen, Exploring hierarchical speech representations with a deep convolutional neural network, in: Proc. UKCI' 11, 2011.

[126] T. Sainath, A.-R. Mohamed, B. Kingsbury, and B. Ramabhad-ran, "Deep convolutional neural networks for lvcsr," in Acous-tics, Speech and Signal Processing (ICASSP), 2013 IEEE Inter-

national Conference on, May 2013, pp. 8614–8618.

[127] O. Abdel-Hamid, A.R. Mohamed, H. Jiang, L. Deng, G. Penn, D. Yu, Convolutional neural networks for speech recognition, IEEE Trans. Audio, Speech

Lang. Process. 22 (2014) 1533–1545, https://doi.org/10.1109/TASLP.2014.2339736.

[128] M. Bi, Y. Qian, K. Yu, Very deep convolutional neural networks for LVCSR, in: Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH. 2015–Janua,2015, pp. 3259–3263.

[129] T. Sercu, C. Puhrsch, B. Kingsbury, Y. Lecun, Very deep multilingual convolutional neural networks for LVCSR, ICASSP, in: IEEE Int. Conf. Acoust. Speech Signal Process. - Proc. 2016–May, 2016, pp. 4955–4959. https://doi.org/10.1109/ICASSP.2016.7472620.

[130] D. Yu, W. Xiong, J. Droppo, A. Stolcke, G. Ye, J. Li, G. Zweig, Deep convolutional neural networks with layer-wise context expansion and attention, in:Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH. 08–12–Sept, 2016, pp. 17–21. https://doi.org/10.21437/Interspeech.2016-251.

[131] T. Zhao, Y. Zhao, X. Chen, Time-frequency kernel-based CNN for speech recognition, in: Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH.2015–Janua, 2015, pp. 1888–1892.

[132] Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Springer, Cham.: Munich, Germany, 2015; Vol. 9351, pp. 234–241.

[133] Çiçek, Ö.; Abdulkadir, A.; Lienkamp, S.S.; Brox, T.; Ronneberger, O. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention; Springer, Cham.: Athens, Greece, 2016; Vol. 9901 LNCS, pp. 424–432.

[134] F Milletari, Navab N , Ahmadi S A . V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation[C]// 2016 Fourth International Conference on 3D Vision (3DV). IEEE, 2016.

[135] Return of the devil in the details: delving deep into convolutional nets (2014), K. Chatfield et al.

[136] ASK, BOA, COA, et al. 1D convolutional neural networks and applications: A survey - ScienceDirect[J]. Mechanical Systems and Signal Processing, 151.