

# 神经网络模型的可视化研究进展

龚少翠<sup>1</sup>, 苗 军<sup>1\*</sup>, 卿来云<sup>2</sup>, 乔元华<sup>3</sup>

<sup>1</sup>北京信息科技大学, 网络文化与数字传播北京市重点实验室, 北京

<sup>2</sup>中国科学院大学, 计算机科学与技术学院, 北京

<sup>3</sup>北京工业大学, 数理学院, 北京

收稿日期: 2022年3月13日; 录用日期: 2022年4月14日; 发布日期: 2022年4月21日

## 摘 要

近年来, 深度学习在计算机视觉、语音识别和自然语言处理等各个领域都取得了巨大的成功。深度学习的主要原理是对数据通过层层特征表达或映射, 使数据在模型最高层线性可分或线性可拟合, 由于此时的模型参数量相对一般浅层模型较少, 从而获得较好的分类或拟合结果。神经网络的应用虽然在许多领域取得了一定的成功, 但仍然有很多问题亟待解决。大部分研究人员仍然不清楚神经网络内部是如何从大规模数据中学习到的有效的特征表示, 神经网络的“黑盒子”特性促进了神经网络可视化技术的发展。可视化技术是以图像可视化的方式对神经网络内部的卷积核以及卷积层所提取到的特征进行分析, 帮助理解神经网络每一层是如何提取特征的, 从而避免在网络训练过程中的盲目调参和试错。神经网络的可视化对于调整参数有着很好的指导作用, 可以使网络结构性能快速达到最优。本文按照以下几方面总结内容: 可视化研究的提起、可视化方法、神经网络模型、可视化工具及可视化应用, 重点关注了可视化神经网络模型的工具; 最后, 对该领域存在的难点及未来研究趋势进行了展望。本文通过论述神经网络模型可视化编程工具的发展与应用, 旨在为神经网络模型的绘制和对其进行更加深入的了解提供参考和根据。

## 关键词

深度学习, 神经网络, 特征表示, 可视化技术

# Advances in Visualization of Neural Network Models

Shaocui Gong<sup>1</sup>, Jun Miao<sup>1\*</sup>, Laiyun Qin<sup>2</sup>, Yuanhua Qiao<sup>3</sup>

<sup>1</sup>Beijing Key Laboratory of Internet Culture and Digital Dissemination Research, Beijing Information Science and Technology University, Beijing

<sup>2</sup>School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing

<sup>3</sup>College of Applied Sciences, Beijing University of Technology, Beijing

Received: Mar. 13<sup>th</sup>, 2022; accepted: Apr. 14<sup>th</sup>, 2022; published: Apr. 21<sup>st</sup>, 2022

\*通讯作者。

文章引用: 龚少翠, 苗军, 卿来云, 乔元华. 神经网络模型的可视化研究进展[J]. 计算机科学与应用, 2022, 12(4): 988-1004. DOI: 10.12677/csa.2022.124102

## Abstract

In recent years, deep learning has achieved great success in various fields such as computer vision, speech recognition and natural language processing. The main principle of deep learning is to express or map the data through layers of features so that the data is linearly separable or linearly fitable at the highest level of the model, which results in better classification or fitting results, because the number of model parameters at this point is relatively small compared to the general shallow model. Although the application of neural networks has been successful in many fields, there are still many problems that need to be solved. Most researchers are still unclear about how neural networks internally learn effective feature representations from large-scale data, and the “black box” nature of neural networks has contributed to the development of neural network visualization techniques. Visualization technique is to analyze the convolutional kernel inside the neural network and the features extracted from the convolutional layers in an image visualization, which helps to understand how features are extracted from each layer of the neural network, thus avoiding blind tuning of parameters and trial and error during the network training process. The visualization of neural networks is a good guide for tuning the parameters, which can make the network structure performance to be optimized quickly. This paper summarizes the content according to the following aspects: the initiation of visual research, visualization methods, neural network models, visualization tools and visualization applications, focusing on the tools of visual neural network models; finally, the difficulties in this field and the future research trends are prospected. By discussing the development and application of visual programming tools for neural network models, this paper aims to provide a reference and basis for the drawing of neural network models and a more in-depth understanding of them.

## Keywords

Deep Learning, Neural Networks, Feature Representation, Visualization Technology

Copyright © 2022 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

深度学习是机器学习的一个新领域，它是以神经网络为架构，对数据进行表征学习的一种算法[1]。传统的机器学习算法采用人工设计的特征集，根据专家经验和相关知识将其组织到机器学习算法中。由于设计人员本身了解这些被定义特征的具体含义，因此，传统机器学习方法在一定程度上是可解释的[2]。相比于传统机器学习算法，以卷积神经网络(Convolutional Neural Network, CNN) [3]为代表的深度学习算法属于特征学习，可对输入数据进行自动特征提取操作，解决了人工设计特征的难题，在性能表现上超过多数传统机器学习算法。然而，CNN 也存在着一定的局限性。一方面，人们至今无法理解 CNN 内部知识表达及其准确的语义含义，难以回答 CNN 到底学习到了哪些特征、特征的具体形式以及不同特征的重要程度度量等问题，这些因素都会导致 CNN 模型的诊断与优化成为经验性、甚至盲目性的反复试探，这不仅会影响模型的性能，还存在一些安全漏洞[4]。另一方面，基于 CNN 模型已被广泛用于各种计算机研究领域，如行人检测[5]，人脸识别[6]，图像检索[7] [8]，场景分割[9]、文本特征的提取[10]等。但是对一些特殊行业，如医疗、金融、军事等领域，可解释性和透明性问题成为其拓展和深入的重大阻碍。

造成这些问题的一个主要原因就是搭建的网络缺乏可解释性，可解释性差会严重阻碍网络各层的鲁棒性评估、网络结构的进一步优化，以及网络对不同应用的适应性和可移植性[11]。

在早期的研究阶段，可视化主要集中在底层特征。随着 CNN 的快速发展和实施，可视化已经扩展到解释 CNN 的整体工作机制。可视化方法是一种展现网络中间层特征的方法，用于理解和解释 CNN 模型所学的特征。文献[12]使用反卷积(Deconvolution)将 CNN 中间层的激活逐层恢复到输入空间，从而观察 CNN 中间层所学特征。文献[13]使用基于梯度的方法，通过网络的反向传播获取输入空间的梯度图，然后对梯度图进行处理，用于表示输入像素对特定输出结果的贡献程度。为了缓解梯度图噪声导致的解释不准确问题，研究者们对基于梯度的可视化方法做了进一步改进，提出了平滑梯度[14]、积分梯度、整流梯度[15]等方法。文献[16]认为，随着 CNN 层数的加深，中间层特征图编码中与决策无关的信息越来越少，因此越往深层目标信息越抽象，语义信息也越丰富。CNN 的最后的卷积层在高层语义信息上达到最佳，其对应的特征图含有最抽象的目标级语义信息，且每个通道检测到目标的不同激活部位。因此，通过对最后的特征图进行通道级加权调整，可生成与特定类别最相关的类激活图。注意力机制在图像分类中应用较为广泛，可以帮助网络在众多信息中聚焦于当前任务最为关键的特征，文献[17]提出一种基于注意力机制的卷积神经网络可视化方法，用于可视化解卷积神经网络中间层所学特征。本文将结合近年来神经网络的相关文献，从神经网络几种重要的模型、神经网络模型结构的可视化工具、可视化应用及编程这三个方面，介绍神经网络模型的设计和可视化编程的研究进展，最后对该领域未来的发展进行一定的讨论和总结。

## 2. 神经网络模型

### 2.1. LeNet-5

LeNet-5 [18]虽然提出的时间比较早，但它是一个非常成功的神经网络模型。LeNet-5 主要是用来进行手写字符的分类和识别，基于 LeNet-5 的手写数字识别系统在 20 世纪 90 年代被美国很多银行使用，用来识别支票上面的手写数字。LeNet-5 的网络结构如图 1 所示。

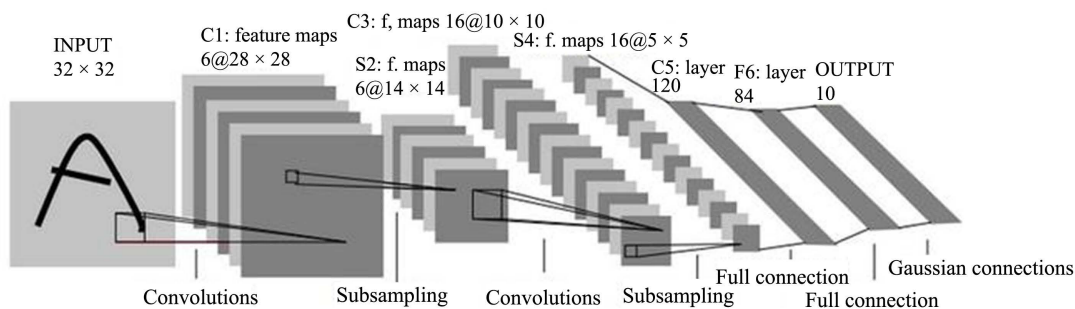


Figure 1. LeNet-5 network structure  
图 1. LeNet-5 网络结构

LeNet-5 最早确立了 CNN 的结构，虽然这个网络比较小，但现在许多网络模型的内容结构在 LeNet-5 的网络中都可以看到，比如卷积层、池化层、全连接层。由于当时缺乏大量的数据进行训练，计算机的性能也较低，因此，LeNet-5 神经网络在处理复杂问题时的效果并不理想，同时在计算过程中也出现了梯度消失的情况。

### 2.2. AlexNet

AlexNet [19]是第一个现代深度卷积网络模型，其首次使用了很多现代深度卷积网络的技术方法，比

如使用 GPU 进行训练，采用了 ReLU 作为非线性激活函数，在一定程度上解决了梯度消失的问题，使用 Dropout 防止过拟合，使用数据增强来提高模型的准确率等。AlexNet 赢得了 2012 年 ImageNet 图像分类竞赛的冠军。

AlexNet 的结构如上图 2 所示，包括 5 个卷积层、3 个全连接层。因为网络的规模超出了当时单个 GPU 的内存限制，AlexNet 将网络拆分为两半，分别放在两个 GPU 上，GPU 之间只在某些层之间进行通信。

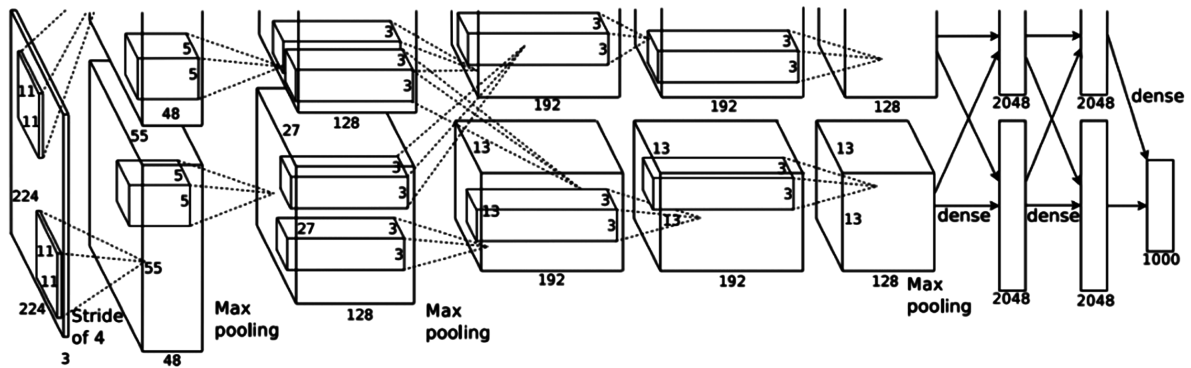


Figure 2. AlexNet network structure  
图 2. AlexNet 网络结构

### 2.3. Inception 网络

在卷积网络中，设置卷积层中卷积核的大小是一个很关键的问题。在 Inception 网络中[20]，一个卷积层由多个不同大小的卷积操作组成，这种结构称为 Inception 模块，Inception 网络是由多个 Inception 模块和少量的池化层堆叠而成。Inception 模块的目的是设计一种优良局部拓扑结构的网络，即对输入图像并行的执行多个卷积运算或池化操作，并将输出结果拼接为一个非常深的特征图。图 3 给出了 v1 版本的 Inception 模块结构，采用了 4 组平行的特征抽取方式，分别是 1×1、3×3、5×5 的卷积和 3×3 最大池化。同时，为了提高计算效率，减少参数数量，Inception 模块在 3×3、5×5 卷积之前、3×3 最大池化之后，进行一次 1×1 的卷积来减少特征映射的深度。如果输入特征映射之间存在冗余信息，1×1 的卷积操作相当于先进行一次特征抽取。

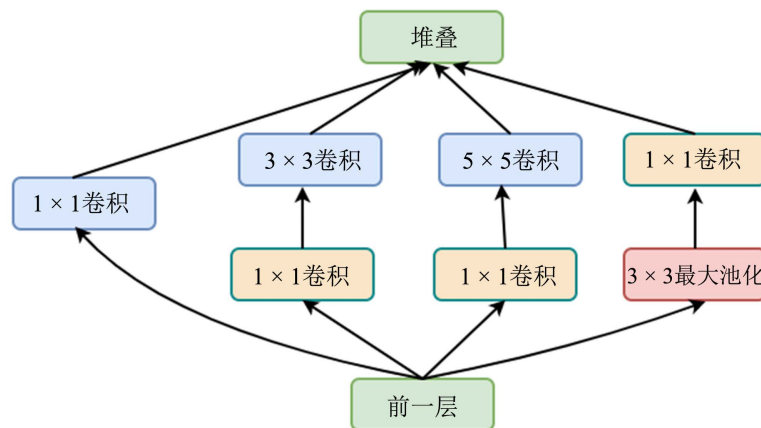


Figure 3. Inception v1 module structure  
图 3. Inception v1 模块结构

Inception 网络有多个版本, 其中最早的是 Inception v1 版本也就是著名的 GoogLeNet。GoogLeNet 赢得了 2014 年 ImageNet 图像分类竞赛的冠军。

GoogLeNet 由 9 个 Inception v1 模块和 5 个池化层以及其他一些卷积层和全连接层组成, 总共为 22 层网络。这是一个深层网络, 它也会遇到梯度消失的问题, 为了解决这个问题, GoogLeNet 在网络的中间层引入两个辅助分类器来加强监督信息。

之后, 研究人员继续对网络进行的改进, 产生了 GoogLeNet 的升级版, 也就是后来的 Inception v2 [21], Inception v3 [22], Inception v4 [23]。其中比较有代表性的就是 Inception v3 网络。Inception v3 网络采用多层的小卷积核来替代大的卷积核, 以减少计算量和参数量, 并保持感受野不变, 具体包括: 使用两层  $3 \times 3$  的卷积来替换 v1 中的  $5 \times 5$  的卷积; 使用连续的  $K \times 1$  和  $1 \times K$  来替换  $K \times K$  的卷积。此外, Inception v3 网络同时也引入了标签平滑以及批量归一化等优化方法进行训练。

## 2.4. ResNet

残差网络(Residual Network, ResNet) [24]通过给非线性的卷积层增加直连边(也称为残差连接)的方式来提高信息的传播效率。传统的卷积网络在信息传递的过程中或多或少会存在信息丢失, 损耗等问题, 同时还会导致梯度消失或者梯度爆炸, 导致很深的网络无法训练。ResNet 网络在一定程度上解决了这个问题, 通过直接将输入信息绕路传到输出, 保护信息的完整性, 简化了学习的目标和难度。

下图 4 给出了一个典型的残差单元示例。残差单元由多个级联的卷积层和一个跨层的直连边组成, 再经过 ReLU 激活后得到输出。

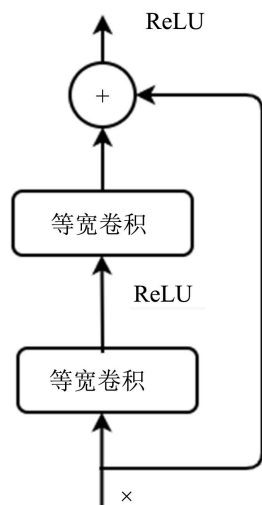


Figure 4. Residual unit structure

图 4. 残差单元结构

残差网络就是将多个残差单元串联起来构成的一个非常深的网络。

目前, 卷积神经网络已经成为计算机视觉领域的主流模型。在残差网络之后, 为了进一步缓解梯度消失的问题, Huang 等人提出 DenseNet 模型[25], 其核心思想在于建立了不同层之间的连接关系, 充分利用了特征, 大幅减少网络的参数量, 进一步减轻了梯度消失问题, 而且训练效果非常好。研究发现, 对于相同的正确率水平, 更小的 CNN 架构可以提供更大的优势, 因此, Iandola 等人提出 SqueezeNet 网络模型[26], 它在 ImageNet 上实现了和 AlexNet 相同的正确率, 但是只使用了 1/50 的参数。更进一步, 使用模型压缩技术, 可以将 SqueezeNet 压缩到 0.5MB, 这是 AlexNet 的 1/510。

### 3. 可视化神经网络模型的工具

在深度学习领域, 各种网络模型随处可见, 认识和学习网络模型的最好方法就是通过可视化的方式展现出来。目前, 常用的设计和可视化网络结构的工具主要包括 Draw\_Convnet [27]、NN-SVG [28]、ENNUI [29]、Netscope CNN Analyzer [30]、PlotNeuralNet [31]、Tensorboard [32]、Netron [33]、TensorSpace [34]、Keras.js [35]、CNN explainer [36]和 CNNVis [37]等。

#### 3.1. Draw\_Convnet

Draw\_Convnet 工具, 它是一个用 python 脚本来绘制卷积神经网络结构的工具, 其核心工具是 matplotlib, 绘制出来的神经网络模型虽然很简单, 但布局清晰, 又可以严格控制, 非常适合写论文使用。如下图 5 所示, 是用 Draw\_Convnet 工具绘制的 LeNet-5 网络结构。我们可以通过修改代码严格控制神经网络模型各层卷积核的大小、数量以及文本列表的位置和布局, 使用起来灵活方便, 简单实用。

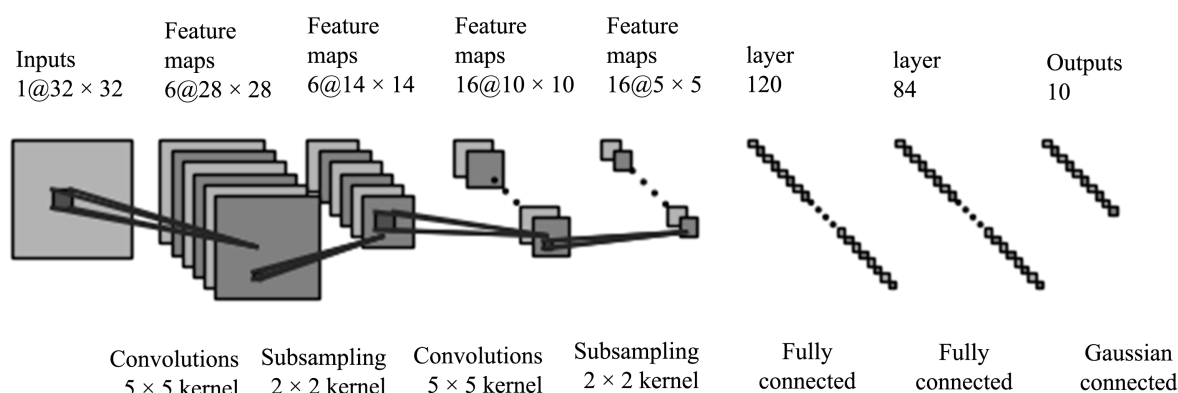


Figure 5. LeNet-5 network structure drawn by Draw\_Convnet

图 5. Draw\_Convnet 绘制的 LeNet-5 网络结构

#### 3.2. NN-SVG

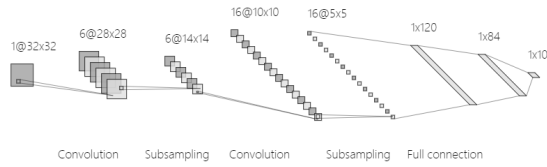
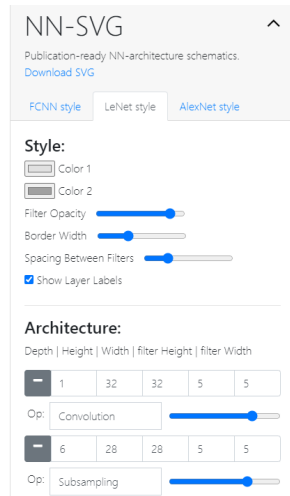
NN-SVG 是一种用于以参数方式而非手动方式创建神经网络架构图的工具。它还具有将这些绘图导出为可缩放矢量图形文件的功能, 适合写学术论文的时候使用。NN-SVG 这个工具可以非常方便的画出各种类型的图。首先, NN-SVG 可以绘制以平铺网络结构的形式展示 LeNet 模型, 如图 6(a)所示, 用二维的方式, 可以查看每一层特征图的大小和通道数目, 添加或删除网络层数也十分方便。其次, NN-SVG 不仅可以绘制 LeNet 模型, 还可以绘制另外两种不同类型的网络, 第一种是可以绘制出包括以节点形式展示的全连接网络, 如图 6(b)所示, 它不仅可以根据更改神经元连线的颜色, 还可以更改每一层的神经元个数, 添加或者删除隐藏层, 比较常见的功能该工具都已包括, 这个特别适合传统的全连接神经网络的绘制。第二种模型是 AlexNet 模型, 如图 6(c)所示, NN-SVG 以三维块的形式展现 AlexNet 的风格, 可以更加真实地展示卷积过程中高维数据的尺度的变化, 目前只支持卷积层和全连接层。

NN-SVG 提供了通过许多大小、颜色和布局参数根据用户的喜好设置图形样式的能力。

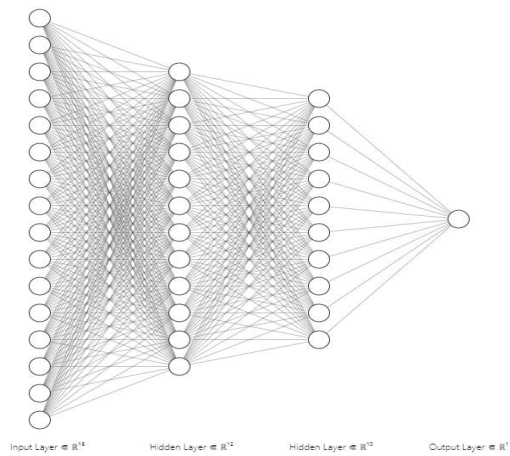
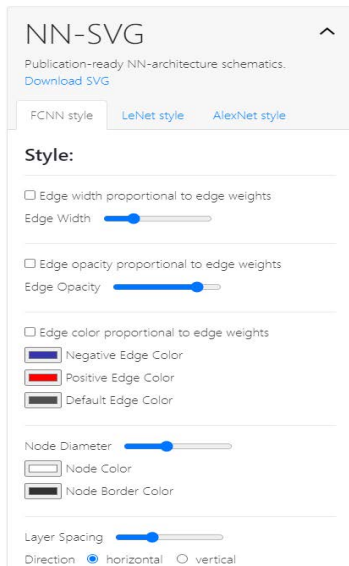
NN-SVG 这个工具十分强大, 作图时只需选择一种类型, 然后在左侧的配置栏里填入自己网络的详细参数, 就可以实现自动作图, 这个工具方便好用, 风格齐全, 功能多样。

#### 3.3. ENNUI

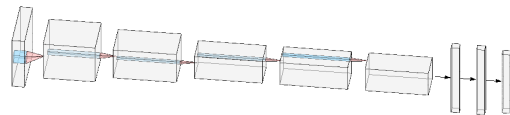
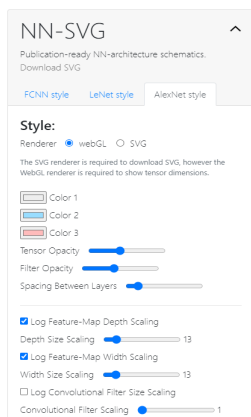
ENNUI 主要通过拖和拽相应的图形框来实现一个网络结构的可视化, 如下图 7 所示。



(a) NN\_SVG 绘制的 LeNet-5 网络结构



(b) NN\_SVG 绘制的 FCNN 网络结构



(c) NN\_SVG 绘制的 AlexNet 网络结构

**Figure 6.** Three different types of network structures drawn by NN-SVG  
**图 6.** NN-SVG 绘制的三种不同类型的网络结构

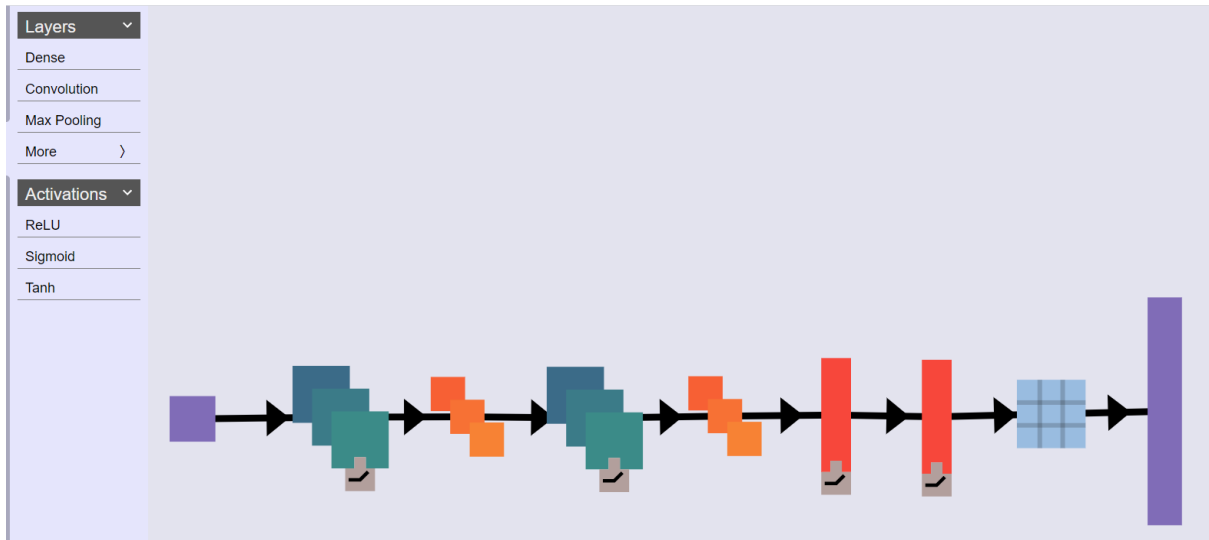


Figure 7. LeNet-5 network structure drawn by ENNUI  
 图 7. ENNUI 绘制的 LeNet-5 网络结构

左边是网络的基本模块选择框，可以根据网络需要直接单击选择然后拖拽到下面，非常简单方便，整个绘制神经网络模型架构的过程也非常的简单，但是绘制出来的神经网络模型整体比较抽象化，不易于新手学习。

### 3.4. Netscope CNN Analyzer

Netscope CNN Analyzer 是一款基于 web 端的卷积神经网络结构可视化和分析工具，绘制出的模型各层的参数情况，非常直观清晰，如下图 8 所示，使用的是 Netscope CNN Analyzer 工具绘制的 AlexNet 神经网络模型，左侧可以编辑网络的详细参数，然后按下 Shift+Enter 键即可自动编译生成模型，可以在右边看到模型的可视化结构，将鼠标放在右侧图上各层时，可以显示出输入输出大小、卷积核大小以及在各层中所进行的操作。

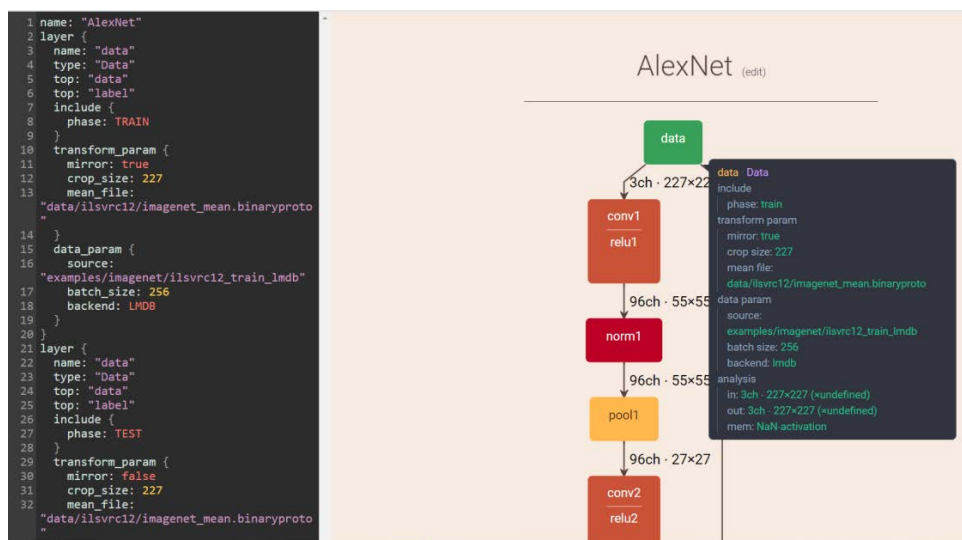


Figure 8. AlexNet network structure drawn by Netscope CNN Analyzer  
 图 8. Netscope CNN Analyzer 绘制的 AlexNet 网络结构



### 3.5. PlotNeuralNet

PlotNeuralNet 目前主要支持的是卷积神经网络，其中卷积层、池化层、Softmax 等常规的层在代码中都有定义，但是缺少 RNN [38]相关的可视化层展示。PlotNeuralNet 这个工具主要是基于 Latex 代码实现的或者使用 Python 脚本编写结构模型。有很多论文中的神经网络模型就是使用这个工具可视化的。它的缺点是上手有点难度。

图 9 是使用 PlotNeuralNet 绘制的 FCN [39]的模型图，可以看到这个可视化工具绘制的模型图是很清晰的，修改模型时只需修改代码即可，可控性较高。PlotNeuralNet 的主要特点是，脚本化，自由度高。

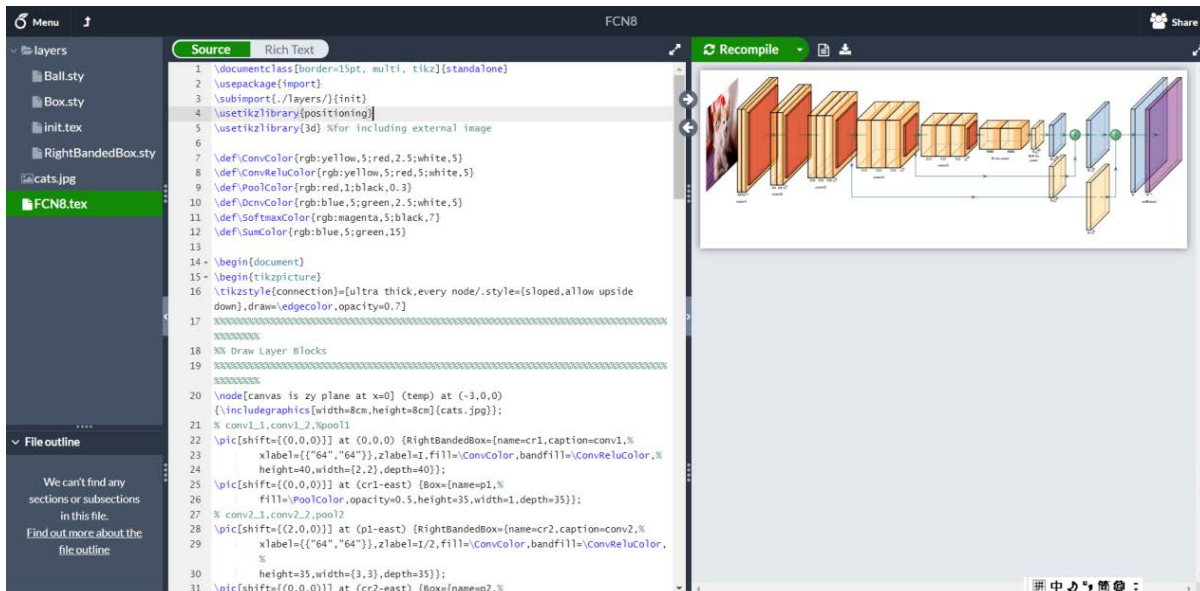


Figure 9. FCN network structure drawn by PlotNeuralNet  
图 9. PlotNeuralNet 绘制的 FCN 网络结构

### 3.6. Tensorboard

Tensorboard 是 tensorflow 内置的一个可视化工具，在安装 tensorflow 的同时 tensorboard 就已经安装好了。我们在构建神经网络模型时，网络开始训练时很多细节对外界来说是不可见的，比如参数如何变化，准确率的变化，这些问题都不清楚。但是，Tensorboard 通过结合 web 应用为我们提供这一功能，它将模型训练过程的细节以图表的形式通过浏览器可视化展现在我们眼前，通过这种方式我们可以清晰感知 weight、bias、accuracy 的变化，把握训练的趋势。具体如下图 10 所示，Abadi [31]等人使用 Tensorboard 可视化模型，有效的展示了卷积神经网络在运行过程中的计算图、各种指标随着时间的变化趋势以及训练中使用到的数据信息，我们还可以根据训练图，对模型的学习情况进行评估考量，比如我们可以看损失图，根据损失训练图决定是否增加训练次数。宗春梅等人[40]将普通神经网络和卷积神经网络相结合的方法应用于推荐系统，在训练神经网络时，通过 Tensorboard 工具实现可视化，通过 Tensorboard 来记录、监测每个 epoch 的梯度值，继而可以根据每次循环后损失率的变化及走势，分析出权值的更新方向，然后对神经网络学习率等参数做出相应的设置和调整。周倩[41]使用 Tensorboard 可视化 CNN 模型结构和参数变化过程，为实际应用中优化模型具体参数提供理论支持，辅助调参工作获得最优模型，提升了模型对图像分类的准确率同时也提升了模型的泛化能力。使用可视化方法展示网络的结构、参数变化以及模型卷积层的学习过程，为实际应用中优化提供理论支持。

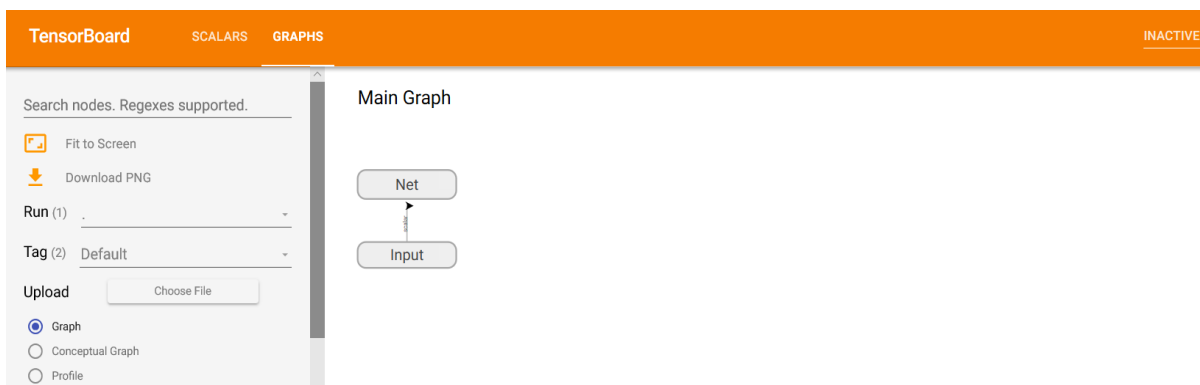


Figure 10. Tensorboard-visualized network structure  
图 10. Tensorboard 可视化网络结构

### 3.7. Netron

Netron 是一个神经网络可视化包，支持绝大多数神经网络操作。该工具操作简单，不需要写代码，只需下载安装一步操作即可。最后，保存快捷，对于可视化的结果，就像保存普通的文件一样，一步到位，保存到自己的电脑上。该 Netron 包可以为不同节点显示不同的颜色，卷积层用蓝色表示，池化层和归一化层用绿色表示，数字操作用黑色显示。在使用方面，可以直接访问网页端，上传模型文件，就可以看到网络结构图，点击结构图中的任意层，右边都会出现该层的具体信息，使用起来很方便。如图 11 所示，使用 Netron 可视化 mobilenet\_v2 网络结构[42]，可以清晰的看到每一层的输入输出参数，网络总体的架构，而且 Netron 支持各种不同网络框架，同时兼容 win、mac、linux 等操纵系统，简单好用。

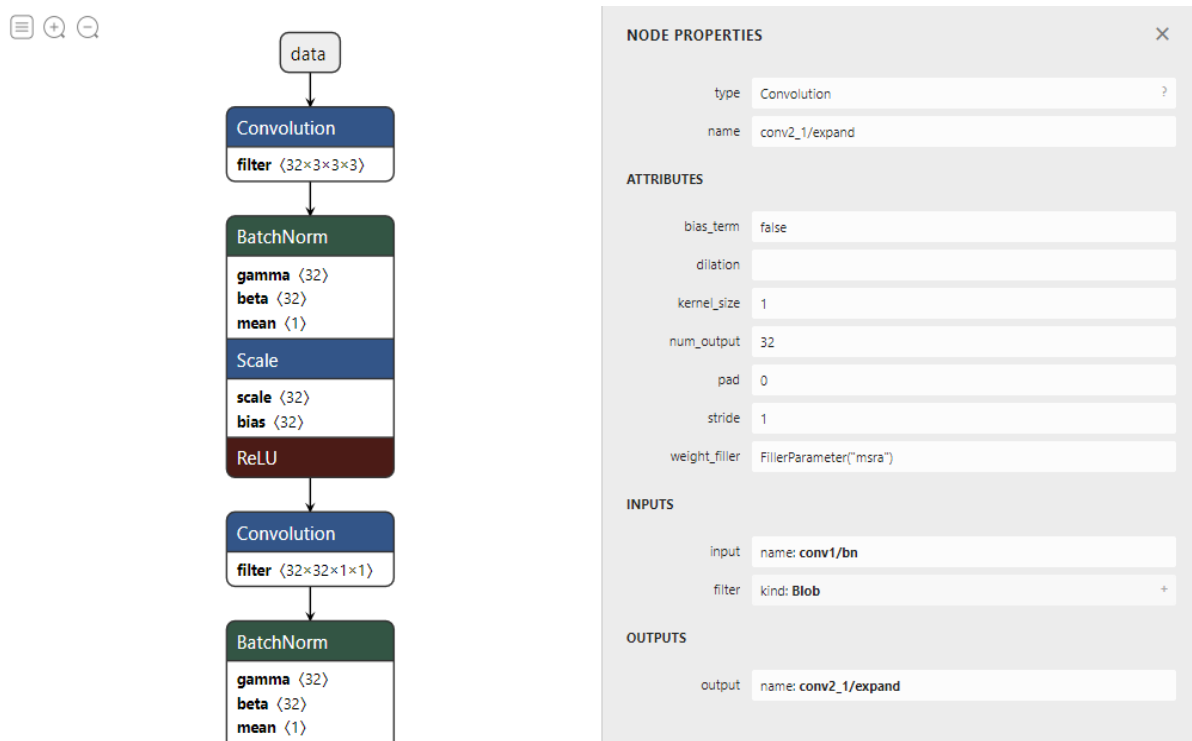


Figure 11. Netron-visualized mobilenet\_v2 network structure  
图 11. Netron 可视化 mobilenet\_v2 网络结构

### 3.8. TensorSpace

TensorSpace 是一个神经网络三维可视化框架。其可以让用户使用浏览器方便地构建一个可交互的神经网络 3D 结构。3D 模型不仅可以直观展示出神经网络的结构特征，比如，哪些层相连，同时用户还可以利用 3D 模型的表意能力特点，在浏览器中进行模型预测，即运行已经训练好的模型看输入输出分别是什么，帮助理解模型。TensorSpace 可以全面地展示层间数据。如图 12 是 TensorSpace 可视化 YOLO 网络结构[43]，通过获取层间输出并渲染到 3D 实体上，使用者可以同时观察到模型结构和层间输出，信息密度更高，更直观。而且可以呈现出每一层对输入数据的直接处理结果，完美了解如何一步一步得到最终预测结果。

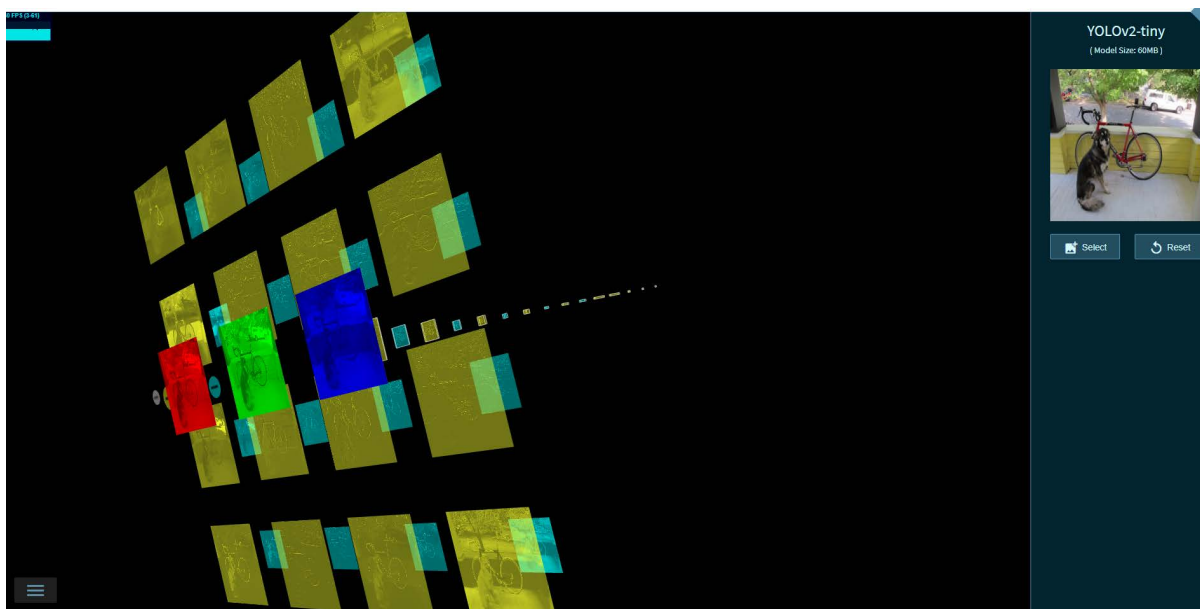


Figure 12. TensorSpace-visualized YOLO network structure

图 12. TensorSpace 可视化 YOLO 网络结构

### 3.9. Keras.js

Keras.js 是一款线上可视化工具，如下图 13 所示，Keras.js 里面包括 ResNet-50、Inception v3 等网络模型架构，界面上方可以输入图片，预测输出结果，界面下方是网络的模型图。我们可以选中一个神经网络结构输入图片测试输出结果，这种方式方便快捷。结构图上每一个卷积层上都有卷积核大小、步长大小和有无填充字样，方便我们快速了解网络的具体结构计算。在 Keras.js 工具的 Basic Convnet 手写一个数字输入之后右边还为我们可视化出每层的处理结果，更加直观的帮助我们了解模型的内部操作。

### 3.10. CNN Explainer

CNN explainer 是一个交互式可视化系统，其总结了 CNN 的结构，并按需、动态的可视解释视图，帮助用户了解 CNN 的底层组件。通过跨抽象级别的平滑转换，CNN explainer 使用户能够检查低级数学操作和高级模型结构之间的相互作用。如下图 14 所示在训练神经网络的过程中，我们可以通过单击并将鼠标停留在上面的可视化的各个部分，可以很清晰的看到 CNN 的内部运作，使用起来非常方便。CNN explainer 使用现代网络技术开发，无需安装或专门的硬件就能在用户的浏览器中运行，拓宽了公众接受现代深度学习技术教育的渠道。

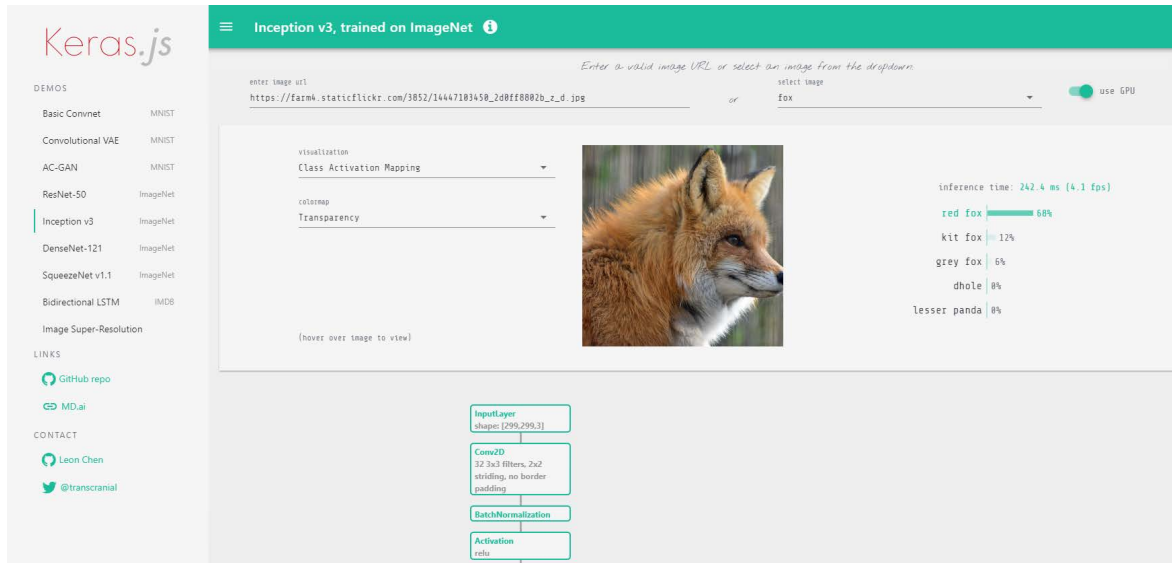


Figure 13. Keras.js-visualized Inceptionv3 network structure

图 13. Keras.js 可视化 Inception v3 网络结构

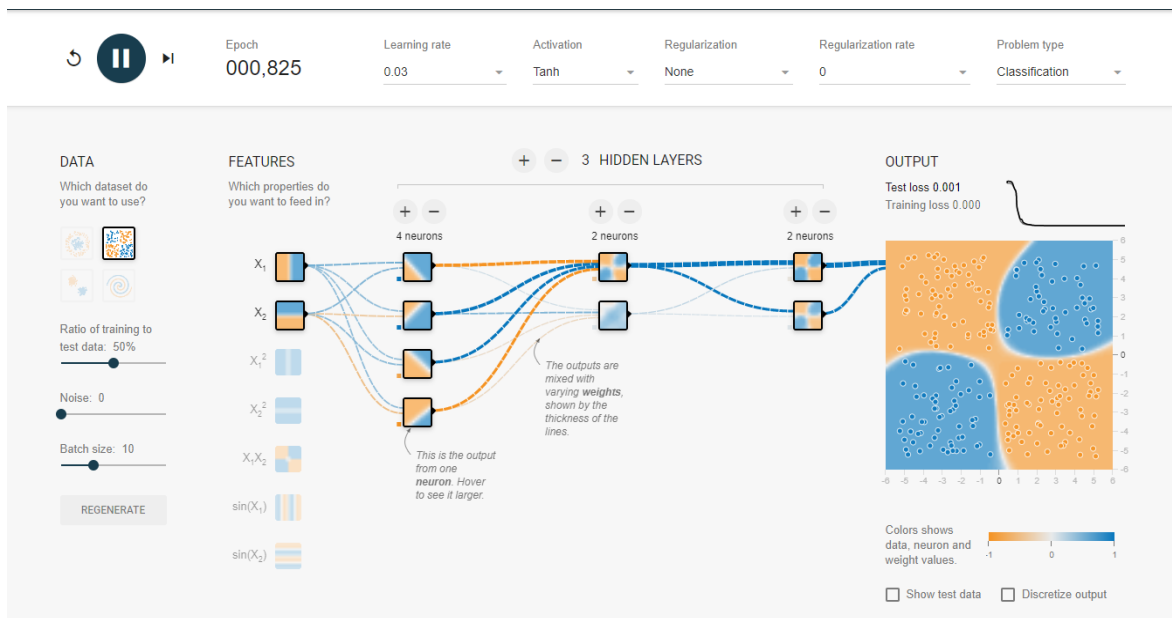


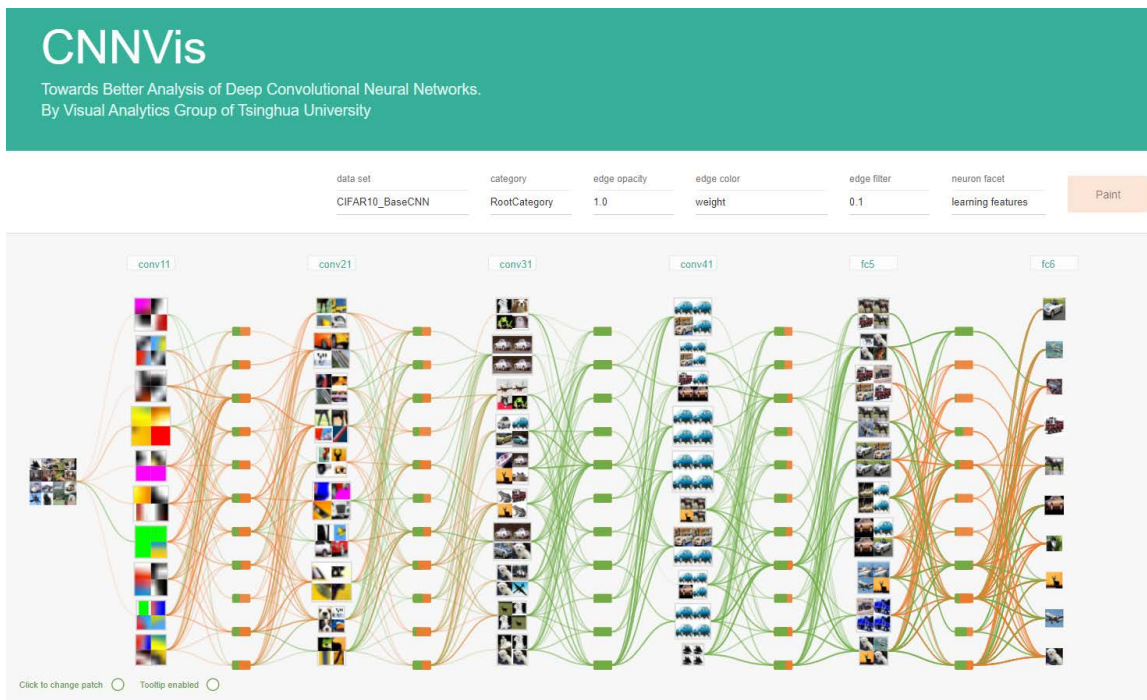
Figure 14. CNN explainer visual classification questions

图 14. CNN explainer 可视化分类问题

### 3.11. CNNVis

对于理解和分析深度卷积神经网络而言，面临主要挑战是：CNN 可能是由几十或几百的层组成、每层有几千个神经元，它们之间错综复杂，不好理解。

为了解决这个问题，提出了一个交互式视觉分析系统，称作 CNNVis，如下图 15 该系统是为了帮助机器学习专家更好的理解，分析与改进 CNNs，基于深度 CNN 的结构，构思出 DAG，其中每个节点代表一个神经元，每个边代表神经元对之间的连接；基于 DAG 表示，提出了混合可视化并设计了矩阵重新排序算法等。



**Figure 15.** CNNVis visual root category questions  
**图 15.** CNNVis 可视化分类问题

以上这些可视化工具的输出结果各有不同，既有直接打印的，也有黑白图、彩色图可视化结果，以及三维可视化结果，这些可视化工具基本都可以支持目前主流的深度学习框架，可以根据需求和使用的框架来进行选择，能够满足大部分人对可视化网络结构的需求。将文中的各种可视化工具进行比较，结果如表 1 所示。

**Table 1.** A comparison of visual neural network model tools  
**表 1.** 可视化神经网络模型工具的对比

工具对比	构图方式	展示维度	优点	缺点	适用范围	是否支持 在线使用
draw_convnet	Python 脚本	二维	布局清晰，可严格控制	编程构图，不合适初学者	绝大多数网络	不支持
NN-SVG	设置参数	二维/三维	方便，各层之间界面控制增减；选择空间大	功能不全面；各层连接不美观	FCNN、LeNet、AlexNet	支持
ENNUI	拖拽图形框	二维	操作简单方便	图形比较抽象化，不易于新手学习	绝大多数网络	支持
Netscope CNN Analyzer	代码编辑	二维	清晰看到各层的参数情况	编程构图，不合适初学者	卷积神经网络(有向无环图)	支持
PlotNeuralNet	Latex 代码或者 python 脚本	三维	脚本化，自由度高	上手有难度，不适合新手	卷积神经网络	支持
Tensorboard	Tensorflow 结合 Web	二维	清晰感知参数变化，把握训练趋势	操作复杂，不适合新手	绝大多数网络	Tensorflow 自带可视化工具

Continued

Netron	上传模型文件, 生成网络结构	二维	操作简单, 方便使用	模型较大时, 时间会比较长	绝大多数网络	支持
TensorSpace	TensorFlow.js、Three.js 和 Tween.js 构建	三维	3D 形式更加直观清晰的描述模型	缺乏模型细节的展示, 参数变化未知	绝大多数网络	支持
Keras.js	Python 脚本	二维	方便快捷, 可以直接得出预测结果	创建新模型时较复杂	绝大多数网络	支持
CNN explainer	设置参数	二维	动态可解释视图, 清晰的看到模型的内部运作, 使用起来非常方便	有局限性, 数据集较少, 网络类型较单一	分类、回归	支持
CNNVis	BaseCNN 设计	二维	可视化分析系统帮助机器学习专家更好的理解、分析、设计深度卷积神经网络。	有局限性, 不能自己设计神经网络	BaseCNN 及其变体	支持

#### 4. 可视化的应用

首先在诊断和优化网络方面, CNN 学习效果的诊断和结构优化上, 基于卷积的可视化能够观察任意层的神经元的激活, 从而分析 CNN 的学习率、卷积核尺寸及步长等重要参数的设计是否达到最优; 其次在理解和解释模型方面, 可视化是理解 CNN 模型的一种重要途径, 在图像领域广泛应用, 常见于图像分类、场景识别等任务的可视化解释; 除了对 CNN 本身的理解和诊断, 可视化方法在其他任务也有扩展与延伸, 在实际领域方面, 可视化方法能够提升对推荐系统决策结果的理解, 以及与知识图谱的结合来实现可解释的推荐算法[44]。对于自动驾驶[45] [46]以及智能医疗等领域, 由于这些领域对于决策风险的承受能力较低, 可视化方法对这些领域应用的落地至关重要。

#### 5. 可视化编程

可视化编程的思想为“所见即所得”, 即随时可以看到结果, 程序与结果的调整同步。和传统的编程方式相比, 可视化编程相对来说更为简单, 只需要通过简单的操作, 就可以设计出程序界面。但是, 可视化编程大大降低了代码的自由程度和可控性, 我们需要按照可视化工具所给出的标准代码模板进行模型的构建, 这在一定程度上降低了代码的自由度。可视化编程可以分为两大类: 基于块的可视化编程和基于流的可视化编程。在基于块的可视化编程环境中, 不需要输入文本代码, 直接可以将预先定义好的块通过拖拽放到脚本区域, 也不需要打字, 操作起来简单方便, 比如可视化工具中的 ENNUI, 建立神经网络模型非常方便快速, 唯一的缺点就是比较抽象化, 不利于初学者学习; 基于流的可视化编程实际上是基于节点的编程。它们通常使用流程图来表示状态、逻辑或数据的变化。同样的效果, 使用可视化编程工具操作, 只需根据需求在工作区里面简单地拖动几个控件, 并且在它们之间绘画箭头来完成神经网络的绘制。而在非可视化编程工具里, 你得思考如何输入各种命令, 与可视化编程工具比起来, 要复杂很多。

#### 6. 存在的难点及发展趋势

近年来, 可视化 CNN 相关研究越来越多, 研究者们提出了多种多样的可视化方法和工具, 极大推动

了该领域的进展,但仍存在一些难点问题有待解决,本节对其进行了归纳,并分析了未来可能的研究趋势。

1) 对于可视化方法,仍存在稳定性、解释能力有限等问题。

不同图像的可视化效果不尽相同,有些图像可能直接导致可视化方法的失效,失效的原因尚不清楚,仍有待进一步的探究。此外,面对复杂背景条件的图像、多目标场景、小目标图像等,受限于模型本身的性能约束,可视化方法的解释效果并不一定好。未来可能的研究趋势是将可视化方法与其它解释方法的结合,从不同侧面不同角度解释模型,从而缓解可视化方法解释效果受限的问题。

2) 对于可视化神经网络工具,仍存在一些缺陷。

目前为止,虽然研究者开发了很多可视化神经网络的工具,但还是存在着一些缺陷。有些可视化工具操作复杂,使用受限,创建模型时较复杂;有些缺乏模型细节的展示,参数变化未知,功能不全面。因此,开发出一款简单好用、功能全面的可视化工具,能够实现对神经网络更好的解释效果。

## 7. 总结与展望

本文主要围绕可视化神经网络模型这个话题展开研究,总结了几个重要的神经网络模型以及可视化神经网络模型的工具,并对这些常用的可视化工具进行了介绍与对比。随着深度学习越来越多地应用于我们的日常生活中,更加详细的了解神经网络的内部机制对我们日常的学习也愈加重要。在实际项目中,我们会遇到各种网络模型,如果单纯的去查看模型文件,我们很难直观的浮现网络的架构。目前很多可视化开发工具让我们体会到了可视化开发带来的便利,但在神经网络的可视化自动编程方面,技术的发展还远远没有成熟,使之在神经网络的设计上没有显示出应有的开发便利。不过,现在越来越多的开发人员尝试使用可视化工具,并且在这一领域中积极探索,可以预见,未来神经网络模型的可视化和自动编程技术会使我们对神经网络内在的工作原理和创新神经网络的构造设计产生更加积极的作用。

## 基金项目

北京市自然科学基金项目(资助号:4202025),国家自然科学基金项目(资助号:61771341、41776204)。

## 参考文献

- [1] 王浩滢. 深度学习及其发展趋势研究综述[J]. 电子制作, 2021(10): 92-95.
- [2] 司念文, 张文林, 屈丹, 罗向阳, 常禾雨, 牛铜. 卷积神经网络表征可视化研究综述[J/OL]. 自动化学报: 1-31. <https://doi.org/10.16383/j.aas.c200554>, 2021-05-18.
- [3] LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W. and Jackel, L. (1990) Handwritten Digit Recognition with a Back-Propagation Network. *Neural Information Processing Systems* 1989, Denver, 27-30 November 1989, 396-404.
- [4] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. and Fergus, R. (2013) Intriguing properties of neural networks, arXiv Preprint, arXiv: 1312.6199.
- [5] Huang, X., Ge, Z., Jie, Z. and Yoshie, O. (2020) NMS by Representative Region: Towards Crowded Pedestrian Detection by Proposal Pairing. 2020 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, 13-19 June 2020, 10747-10756. <https://doi.org/10.1109/CVPR42600.2020.01076>
- [6] Huang, Z., Zhang, J. and Shan, H. (2021) When Age-Invariant Face Recognition Meets Face Age Synthesis: A Multi-Task Learning Framework. 2021 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Nashville, 20-25 June 2021, 7278-7287. <https://doi.org/10.1109/CVPR46437.2021.00720>
- [7] Brown, A., Xie, W., Kalogeiton, V. and Zisserman, A. (2020) Smooth-AP: Smoothing the Path Towards Large-Scale Image Retrieval. *European Conference on Computer Vision* 2020, Glasgow, 23-28 August 2020, 677-694. [https://doi.org/10.1007/978-3-030-58545-7\\_39](https://doi.org/10.1007/978-3-030-58545-7_39)
- [8] Chen, W., Liu, Y., Wang, W., Bakker, E., Georgiou, T., Liu, L., et al. (2021) Deep Image Retrieval: A Survey. arXiv:2101.11282.

- [9] Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., *et al.* (2020) RandLA-Net: Efficient Semantic Segmentation of Large-Scale Point Clouds. 2020 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, 13-19 June 2020, 11105-11114. <https://doi.org/10.1109/CVPR42600.2020.01112>
- [10] 刘鼎立. 基于深度学习文本特征提取的聚类算法研究[D]: [硕士学位论文]. 桂林: 桂林电子科技大学, 2021. <https://doi.org/10.27049/d.cnki.gglc.2021.000866>
- [11] Shin, H., Roth, H.R., Gao, M., Lu, L., Xu, Z., Nogues, I., *et al.* (2016) Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning. *IEEE Transactions on Medical Imaging*, **35**, 1285-1298. <https://doi.org/10.1109/TMI.2016.2528162>
- [12] Zeiler, M.D. and Fergus, R. (2014) Visualizing and Understanding Convolutional Networks. *European Conference on Computer Vision 2014*, Zurich, 6-12 September 2014, 818-833. [https://doi.org/10.1007/978-3-319-10590-1\\_53](https://doi.org/10.1007/978-3-319-10590-1_53)
- [13] Simonyan, K., Vedaldi, A. and Zisserman, A. (2013) Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. arXiv:1312.6034.
- [14] Smilkov, D., Thorat, N., Kim, B., Viégas, F. and Wattenberg, M. (2017) SmoothGrad: Removing noise by Adding Noise. arXiv:1706.03825.
- [15] Kim, B., Seo, J., Jeon, S.H., Koo, J., Choe, J. and Jeon, T. (2019) Why are Saliency Maps Noisy? Cause of and Solution to Noisy Saliency Maps. 2019 *IEEE/CVF International Conference on Computer Vision Workshop*, Seoul, 27-28 October 2019, 4149-4157. <https://doi.org/10.1109/ICCVW.2019.00510>
- [16] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A. and Torralba, A. (2016) Learning Deep Features for Discriminative Localization. 2016 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, 27-30 June 2016, 2921-2929. <https://doi.org/10.1109/CVPR.2016.319>
- [17] 司念文, 常禾雨, 张文林, 屈丹. 基于注意力机制的卷积神经网络可视化方法[J]. 信息工程大学学报, 2021, 22(3): 257-263.
- [18] Lecun, Y., Bottou, L., Bengio, Y. and Haffner, P. (1998) Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, **86**, 2278-2324. <https://doi.org/10.1109/5.726791>
- [19] Technicolor, T., Related, S., Technicolor, T., *et al.* (2012) ImageNet Classification with Deep Convolutional Neural Networks. *Communications of the ACM*, **60**, 84-90.
- [20] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., *et al.* (2014) Going Deeper with Convolutions. 2015 *IEEE Conference on Computer Vision and Pattern Recognition*, Boston, 7-12 June 2015, 1-9. <https://doi.org/10.1109/CVPR.2015.7298594>
- [21] Ioffe, S. and Szegedy, C. (2015) Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *Proceedings of the 32nd International Conference on Machine Learning*, Vol. 37, Lille, 6-11 July 2015, 448-456.
- [22] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. and Wojna, Z. (2016) Rethinking the Inception Architecture for Computer Vision. 2016 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, 27-30 June 2016, 2818-2826. <https://doi.org/10.1109/CVPR.2016.308>
- [23] Szegedy, C., Ioffe, S., Vanhoucke, V. and Alemi, A.A. (2016) Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, San Francisco, 4-9 February 2017, 4278-4284.
- [24] He, K., Zhang, X., Ren, S. and Sun, J. (2016) Deep Residual Learning for Image Recognition. 2016 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, 27-30 June 2016, 770-778. <https://doi.org/10.1109/CVPR.2016.90>
- [25] Huang, G., Liu, Z., Van Der Maaten, L. and Weinberger, K.Q. (2017) Densely Connected Convolutional Networks. 2017 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, 21-26 July 2017, 2261-2269. <https://doi.org/10.1109/CVPR.2017.243>
- [26] Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J. and Keutzer, K. (2016) SqueezeNet: AlexNet-Level Accuracy with 50x Fewer Parameters and <0.5MB Model Size. arXiv:1602.07360.
- [27] Ding, G.W. (2017) Draw Convnet. [https://raw.githubusercontent.com/gwding/draw\\_convnet/master/draw\\_convnet.py](https://raw.githubusercontent.com/gwding/draw_convnet/master/draw_convnet.py)
- [28] LeNail, A. (2019) NN-SVG: Publication-Ready Neural Network Architecture Schematics. *Journal of Open Source Software*, **4**, Article No. 747. <https://doi.org/10.21105/joss.00747>
- [29] Grosser, S. (2018) ENNUI. <https://github.com/martinjm97/ENNUI>
- [30] Gschwend, D. (2017) Netscope. <https://github.com/cwlacewe/netscope/tree/master/presets>
- [31] Iqbal, H. (2018) PlotNeuralNet. <https://github.com/HarisIqbal88/PlotNeuralNet/tree/master/examples>
- [32] Tensor Flow: Large-Scale Machine Learning on Heterogeneous Distributed Systems (2016).



- 
- [33] Roeder, L. (2016) Netron. <https://github.com/lutzroeder/netron>
- [34] Zhu, C. (2018) TensorSpace. <https://github.com/tensorspace-team/tensorspace>
- [35] Chen, L. (2017) Keras-js. <https://github.com/transcranial/keras-js>
- [36] Wang, Z.J., Turko, R., Shaikh, O., Park, H., Das, N., Hohman, F., *et al.* (2021) CNN Explainer: Learning Convolutional Neural Networks with Interactive Visualization. *IEEE Transactions on Visualization and Computer Graphics*, **27**, 1396-1406. <https://doi.org/10.1109/TVCG.2020.3030418>
- [37] Liu, M., Shi, J., Li, Z., Li, C., Zhu, J. and Liu, S. (2017) Towards Better Analysis of Deep Convolutional Neural Networks. *IEEE Transactions on Visualization and Computer Graphics*, **23**, 91-100. <https://doi.org/10.1109/TVCG.2016.2598831>
- [38] Zaremba, W., Sutskever, I. and Vinyals, O. (2014) Recurrent Neural Network Regularization. arXiv:1409.2329.
- [39] Shelhamer, E., Long, J. and Darrell, T. (2017) Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **39**, 640-651. <https://doi.org/10.1109/TPAMI.2016.2572683>
- [40] 宗春梅, 张月琴, 赵青杉, 郝耀军, 郭玥鑫. 可视化支持下 CNN 在个性化推荐算法中的应用[J]. 计算机系统应用, 2020, 29(6): 204-210. <http://www.c-s-a.org.cn/1003-3254/7451.html>.
- [41] 周倩. 基于卷积神经网络的参数选择及可视化研究[D]: [学位论文]. 哈尔滨: 哈尔滨工业大学, 2019.
- [42] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A. and Chen, L. (2018) MobileNetV2: Inverted Residuals and Linear Bottlenecks. 2018 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, 18-23 June 2018, 4510-4520. <https://doi.org/10.1109/CVPR.2018.00474>
- [43] Zhu, X., Lyu, S., Wang, X. and Zhao, Q. (2021) TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-captured Scenarios. 2021 *IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, Montreal, 11-17 October 2021, 2778-2788. <https://doi.org/10.1109/ICCVW54120.2021.00312>
- [44] Zhang, Y. and Chen, X. (2020) Explainable Recommendation: A Survey and New Perspectives. Now Foundations and Trends. <https://doi.org/10.1561/9781680836592>
- [45] Fan, R., Wang, L., Bocus, M.J. and Pitas, I. (2020) Computer Stereo Vision for Autonomous Driving. arXiv:2012.03194.
- [46] Zhang, S., Song, L., Liu, S., Ge, Z., Li, Z., He, X., *et al.* (2021) Workshop on Autonomous Driving at CVPR 2021: Technical Report for Streaming Perception Challenge. arXiv:2108.04230.