MLND Capstone 论文
用于端到端自动驾驶的卷积神经网络

目录

1 项目概述

无人驾驶技术是将是改变人类生活的一项技术。近年来，不管是国内还是国外，无人驾驶技术的研究和应用都十分火热，Google、TuSimple、AutoX 等公司的无人驾驶汽车都已经开始在真实的道路上开始了测试。不过，以上公司所在道路上测试的无人驾驶汽车均是基于规则的，即是通过感知环境和确定位置，人为地指定无人驾驶的决策。这样做的工作量非常大，而且很难考虑到所有的情况。本项目将使用端到端深度学习训练模型，利用汽车前置摄像头收集到的视频数据，对相应时间人类驾驶时汽车的转向信号进行拟合。当模型部署在汽车上时，模型能通过当前汽车前置摄像头采集到的图像数据，做出与人类极度相似的转向行为，以此达到无人驾驶的目的。这种方式也被称作为行为克隆（Behavioral cloing）。近年来，基于卷积神经网络的深度学习方法得到了很大的发展。卷积神经网络能够更好地提取图像特征，所以它被大量地应用在计算机视觉领域。

2 问题陈诉

我所要解决的问题便是利用卷积神经网络建立一个强大的特征提取器和利用全链接神经网络对特征进行回归拟合，从而利用端到端的深度学习技术解决自动驾驶转向控制问题。其中，卷积神经网络的特区特征的过程不需要人为的干预和制定，卷积神经网络将自动地学会如何提取图像中有用的特征，再通过全连接神经网络、利用学习到的特征对汽车转向信号的 ground truth 进行拟合。

但是，大量现存的卷积神经网络结构，如 AlexNet、VGG、ResNet 甚至是最近出现的 DenseNet，都是用于图像分类设计的，他们在 ImageNet 数据集上表现出了相当好的效果，但是，它们能直接套用在行为克隆上吗？我想不是的。首先，汽车前置摄像头采集的数据分布，应该远不同于 ImageNet 等其他图形分类器的分布; 其次，我们解决的问题也有所不同：目前热门的卷积神经网络多用于分类问题，而不是我们目前所要解决的回归问题。现有的卷积神经网络在大量计算机视觉问题上表现出的优秀性能不能忽视，但是它不一定是解决我们所遇到的问题的最佳方案。

3 评价指标

模型优劣的评价指标主要是均方误差（mean square error，MSE），即：

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 \tag{3.1}$$

其中，y_i 为数据集第 i 个数据的转向信号值，$\hat{y}_i$ 为神经网络对第 i 个转向信号的预测值，n 为数据集的大小。除了 MSE 之外，对于测试数据，我还需要对驾驶过程进行可

视化，来分析所训练的神经网络输出的转向信号是否合理。

4 分析

4.1 DeepTesla 数据集介绍

DeepTesla 数据集是由 MIT 6.S094 发布的一个用于训练无人驾驶汽车模型的数据集，该数据集包含了行车视频和转向信号数据两个部分。其中，行车视频包含了十个片段，每个片段都是 Tesla 汽车在普通道路上行驶过程中前置摄像头所拍摄到的视频画面，文件格式是 MKV。而转向信号数据则包括了视频中每一帧拍摄时候的时间戳、帧编号和当前帧对应的转向信号的 ground truth。转向信号的 ground truth 由 Tesla 汽车的 AutoPoilot 在行驶期间预测得到的。如图 4.1、4.2 所示是 DeepTesla 数据的视频截图以及各转角信号值出现数量的直方图。

图 4.1: DeepTesla 视频数据截图

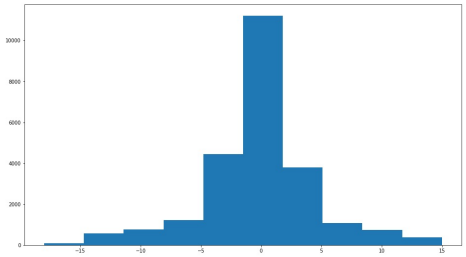

图 4.2: 各转角信号值数量直方图

4.2 算法和技术

4.2.1 监督学习

监督学习的目的是学习一个函数，使得模型对于任意的输入 x，都能给出一个很好的预测输出 y。为了得到这个模型，我们需要确定一个参数随机化的模型，通过算法，利用训练数据对模型进行训练，即是更新参数，使得模型的损失函数最小。以上提到的模型、算法和损失函数，在监督学习中缺一不可。监督学习的模型和算法有很多，比如支持向量机、Logistic 回归、随机森林、决策树、朴素贝叶斯和神经网络等。

在针对图像的监督学习中，图像数据集是作为张量输入的：单张图片则是矩阵，每一个像素点，对应着一个输入的特征。从理论上讲，上述的模型都可以用于解决我们的问题；但是对于图像数据，卷积神经网络有着更好的效果。

4.2.2 神经网络

神经网络是一种多层的感知器，它解决了感知器不能解决的线性不可分问题，是一个比较简单的非线性函数。它通过反向传播算法实现对权值的更新，以此达到模型训练

的目的。

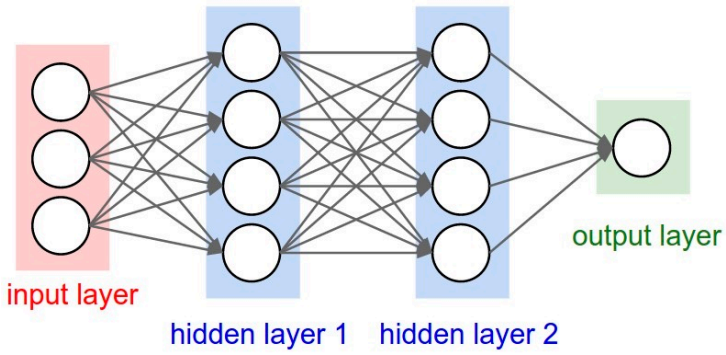

图 4.3: 神经网络

4.2.3 卷积神经网络

卷积神经网络是一种特殊的神经网络，其最大特点就是有用多个卷积层，卷积层通过一个较小卷积核方阵进行卷积运算，并且当前特征图的卷积核权重是共享的。这大大减少了模型中所包含的参数规模，同时也为模型带来了平移不变性。多层的卷积层能逐层自动学习到提取更复杂的特征，比传统的人为设计的特征提取算法更加优秀。同时，卷积神经网路还会包含池化的层，我在之后将详细介绍这一部分。LeNet[1] 第一次讲卷积神经网络用在了图像识别上，在 MNIST 数据集的手写识别中取得了非常好的结果。

4.2.4 NVIDIA 的端到端卷积神经网络模型

NVIDIA 实现了一种卷积神经网络用于行为克隆，在实际的道路测试中取得了非常不错的效果。网络结构如图 4.4 所示该网络包含四个卷积层和三个全连接层组成。但是该网络并没有详细说明用到了什么技术，仅仅只是阐明了基本的网络结构，所以可以把它当做基础模型，在该模型的基础上进行优化。

4.3 基准标准

本文所采用的基准模型是 AlexNet[2]。AlexNet 是经典的卷积神经网络模型，由 SuperVision group 提出，并以 15.3% 的 top-5 错误率获得了 ILSVRC2012 的冠军。AlexNet 使用 ReLU 激活函数，包含了卷积层、池化层和全连接层。

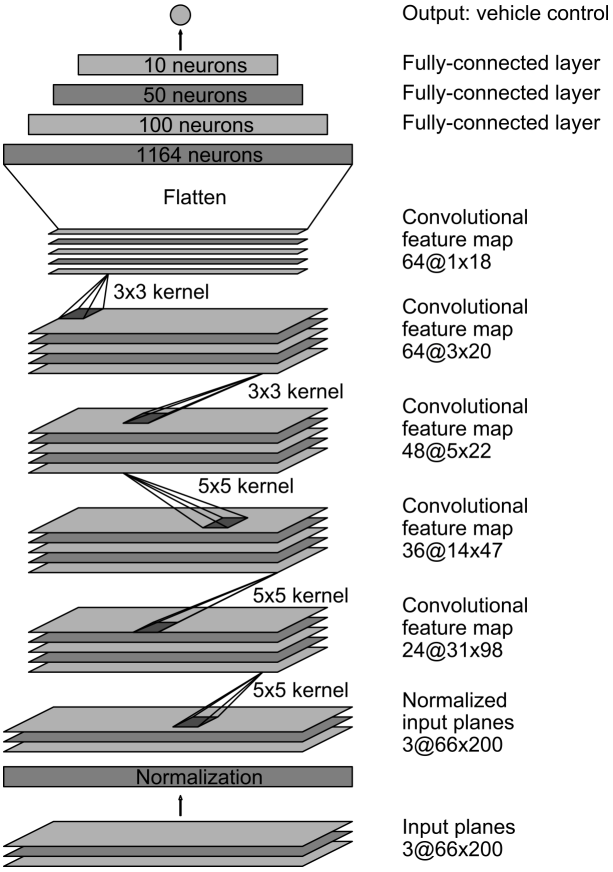

图 4.4: NVIDIA 所使用的模型结构

5 具体方法

5.1 数据预处理

对于 DeepTesla 数据集，需要进行多项预处理才能更好地用于卷积神经网络的训练。原视频是 mkv 文件，所以第一步便是要逐帧提取视频，并将 RGB 图像转化为 YUV 图像，随机打乱图像的顺序，并使之与相应的转向信号对应。

但这远远不够。首先，图像的尺寸比较大，虽然这能包含更多的信息，但是训练这样的模型却非常困难，因为更大的输入意味着模型将要包含更多的参数，将有更大的时间和空间的开销。所以我讲原图像压缩为尺寸为 125x240 的图像。

从示例图片中可以看出，摄像头捕捉的影响的顶端部分和底端部分对于训练模型帮助不大：顶端部分大多是天空，而底端部分则是车辆的引擎盖。我选择将这部分图像裁减掉，以便车辆能够专注于学习路面信息。如图是 RBG 图像经过裁剪的示例。

图 5.1: 原图

图 5.2: 裁剪后

5.2 基础模型

NVIDIA 实现了端到端的卷积神经网络用于行为克隆训练，并且取得了不错的效果。除了输入大小不一样外，我的模型将基本参照这一结构进行训练，并基于这个模型逐步优化，让其能够在 DeepTesla 数据集中也取得很好的效果。接下来，我将详细叙述我对 NVIDIA 模型所做的优化。优化过后的模型结构如图 5.3 所示。

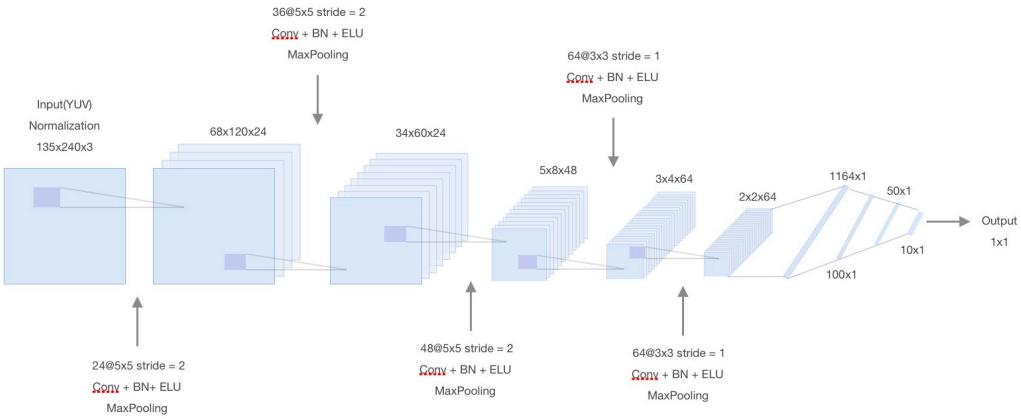

图 5.3: 本文所使用的卷积神经网络模型

5.3 池化

池化（Pooling）是一种非线性的下采样方法。以本文中用到的 Max pooling 为例，则是将图片全部分割为大小相同的子区域方阵，然后计算子区域方阵的最大值。如图 5.4 所示 池化的引入给模型带来了很多优点。池化是一种下采样方法，所以他将缩减模型的参数数量，使得模型训练起来更加容易，并且不容易过拟合。但池化不仅仅只是因为削减参数而被引入的，在计算机视觉中，图像特征的具体位置可能并没有特征的大致位置重要，因为我们在图像处理中还要做到平移不变性（translation invariance）和旋转不变性（rotation invariance），而最大池化的引入能保证小范围的平移和旋转不变性。

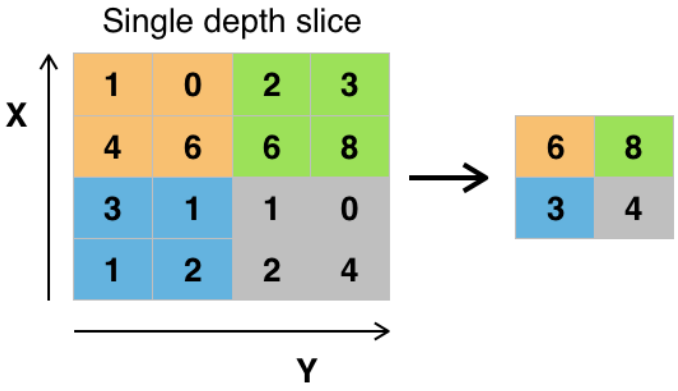

图 5.4: Max pooling 示意图

5.4 Dropout

Dropout[3] 是一种模型的正则化方法，同时可以认为是一种模型的 Bagging 方法。训练 Bagging 通常需要许多的神经网络，而 dropout 则提供了一种廉价的近似实现。在训练过程中，Dropout 会以某个概率 p 随机的移除神经神经元，如图 5.5 所示。

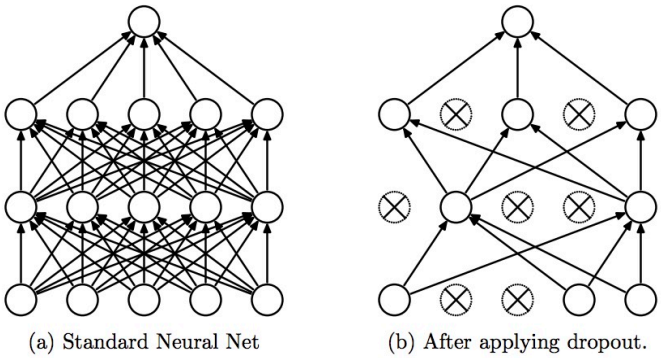

图 5.5: Dropout

当 k 次训练结束后，此时推理的结果可以认为是 2^k 个网络的均值，而仅仅只需要将每个神经元的输出值乘以 p。不仅如此，Dropout 作为一种正则化方法，它的引入可以大大避免过拟合的风险。

5.5 批标准化

批标准化（Batch normalization）是一种重参数化的方法。在神经网络中，参数的不断变化导致后续每一层的输入都在不断变化，这样一来学习率就无法确定，因为某一层神经网络参数的更新效果很大程度上取决于其它层。为了消除这一弊端，可以将神经元输出值得分布满足单位方差和 0 均值，具体操作如图 5.6 所示。

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: γ, β
Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma^2_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma^2_{\mathcal{B}} + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma\widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

图 5.6: 批标准化算法

5.6 ELU

NVIDIA 原本的模型中使用了 ReLU（Rectified Linear Unit）作为激活函数，它被定义为：

$$y = max(0, x) \tag{5.1}$$

相比与 sigmoid 函数，ReLU 在为神经网络带来非线性性质的同时，还能节省更多的时间，并且有着更快的收敛速度。不仅如此，ReLU 相比于 sigmoid 更加仿生。但是 ReLU 的输出并不像标准化的数据那样。0 是处于数据分布中心的，而且当数值小于 0 时，梯度也会是 0，则会导致神经元死亡（dead ReLU），权值永远不会更新。Clevert 等人提出了 ELU[4] 激活函数来克服 ELU 带来的种种缺陷，ELU 被定义为：

$$f(x) = \begin{cases} x & if\ x > 0 \\ \alpha(exp(x) - 1) & if\ x \leq 0 \end{cases} \tag{5.2}$$

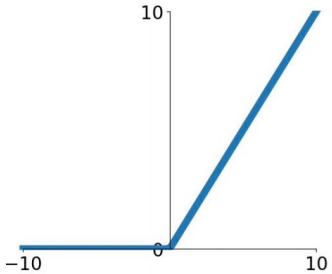

图 5.7: ReLU

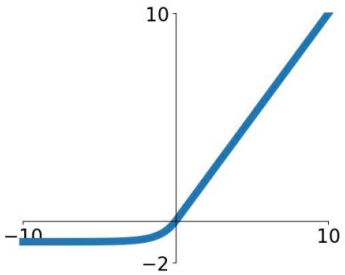

图 5.8: ELU

当输入的值为负数时，ELU 没有简单粗暴地讲负数值直接抹为 0，而是用一指数函数进行运算，得到一个非常接近于 0 的负数值。这样一来，激活函数输出的均值就和 0 非常接近了，并且对于噪声有着良好的鲁棒性。

5.7 He 权重初始化

Kaiming He 提出过一种专门用于当激活函数是 ReLU 时的权重初始化方法，但是在提出 ELU 的论文中使用的权重初始化方式是即是这种，所以我在本文中也沿用这种权重初始化方式。

6 结果与分析

6.1 模型比较

由于基准的 NVIDIA 模型和 AlexNet 长期训练并没有显著的效果提升，所以除这两个模型仅仅训练 10 次外，所有模型均经过了 20 次迭代训练，模型在训练集和验证集上的损失值变化如图 6.1 所示。

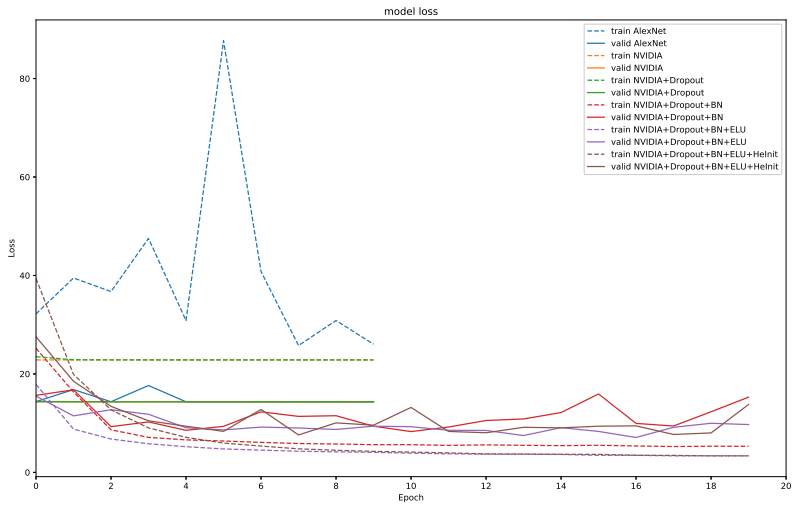

图 6.1: 各个模型在训练过程中损失函数值变化对比

从图中可以看出，我最终使用的优化 NVIDA 模型 +Dropout+ 批标准化 +ELU+He 权重初始化模型在验证集上的损失最小，而且收敛速度也最快；不光如此，该模型在测试集上也能得到最小的损失，所以我认为该模型的所表现的效果最好，我们选取这个模型对测试集的数据进行预测，效果如图所示。由此可见，我基于 NVIDIA 模型优化后的模型有着比较好的效果。

6.2 网络状态可视化

人眼在识别图像的时候，并非是观察了图像的每一个细节，而是只关注图像的一些重要的区域，而从这些重要的区域中提取出信息。同理，如果卷积神经网络模型训练得

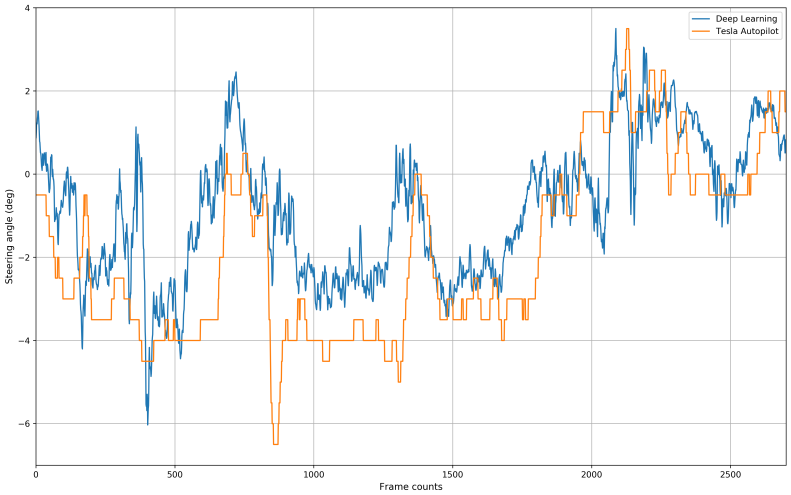

图 6.2: 模型在测试集上的预测结果

当，我们依然有方法可以确定模型所认为的图像的重要区域。我采用 keras-vis[5] 绘制 Saliency map[6] 来可视化网络状态：输入测试图片，返回模型所关注的特征。做到这样的原理比较简单，我们只需要知道当输出随着输入变化的时，输出变化最大的时候，输入变化的区域就可以了。具体公式如 TODO 所示。

$$S = argmax(\frac{\partial output}{\partial input}) \tag{6.1}$$

对于行为克隆的模型来说，好的模型一定会更多地关注到路边线等路面路况信息，而不是路面的颜色。利用 keras-vis 和随机测试图像所做的 Saliency map 如图 6.3 6.4 所示。

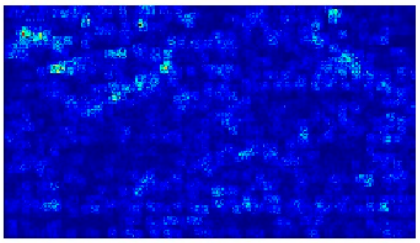

图 6.3: Saliency map

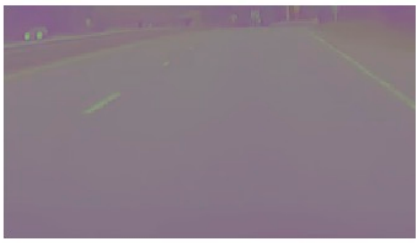

图 6.4: 原图像（YUV）

6.3 在测试集上的可视化

除了计算模型在测试集上每一帧的损失之外，我还用可视化工具对模型在测试集上的表现做了可视化，确保模型所做出的预测合理可靠。在测试集上的结果可视化截图如图 6.5 所示。可见，在大多数情况下，神经网络所预测的转向信号和 Tesla Autopilot 所

图 6.5: 在测试集上测试模型

给出的基本一致，这说明使用该模型训练神经网络实现端到端的自动驾驶是可行的。

6.4 模型缺陷与改进方案

虽然经过改进的 NVIDIA 模型取得了不错的效果，但是他仍然是一种传统的卷积神经网络模型，有着许多缺陷：

- 对于时间缺少长期以来。对于本文中的神经网络而言，每一帧的输入都是独立的；但在人类开车的时候并非这样：我们当前做出的决定，很大程度上是由过去的输入和相应的输出决定的。所以可以将传统的卷积神经网络模型和循环神经网络结合解决这个问题；

- 训练数据过少。DeepTesla 数据集总共只有十个片段，每个片段仅仅只有短短地一分半钟，让模型在十几分钟内学会开车，确实很难。如此少量的数据集会大大降低模型的泛化能力。如图 6.6 所示，当汽车通过天桥时模型预测的结果比较糟糕，我想很大的原因是因为在训练数据中此类数据样本极少甚至没有。

图 6.6: 当汽车通过天桥时

参考文献

[1] LECUN Y, BOTTOU L, BENGIO Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278–2324.

[2] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet Classification with Deep Convolutional Neural Networks[G/OL] //PEREIRA F, BURGES C J C, BOTTOU L, et al. Advances in Neural Information Processing Systems 25. [S.l.]: Curran Associates, Inc., 2012: 1097–1105. http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.

[3] SRIVASTAVA N, HINTON G E, KRIZHEVSKY A, et al. Dropout: a simple way to prevent neural networks from overfitting.[J]. Journal of Machine Learning Research, 2014, 15(1): 1929–1958.

[4] CLEVERT D-A, UNTERTHINER T, HOCHREITER S. Fast and accurate deep network learning by exponential linear units (elus)[J]. arXiv preprint arXiv:1511.07289, 2015.

[5] KOTIKALAPUDI R, CONTRIBUTORS. keras-vis[J], 2017.

[6] SIMONYAN K, VEDALDI A, ZISSERMAN A. Deep inside convolutional networks: Visualising image classification models and saliency maps[J]. arXiv preprint arXiv:1312.6034, 2013.