

一种自学习不确定度的神经网络架构

孙汉卿, 庞彦伟

天津大学电气自动化与信息工程学院, 天津 300072

摘要 在自动驾驶这类多传感器融合任务中, 每个子任务的不确定度对数据融合的策略和结果都有着至关重要的影响, 为使整体系统能够在多工况下稳定运行, 必须要求计算模型以较低的不确定度运行。现有方法仅能在神经网络预测过程中求得不确定度, 很少有方法能够通过自学习的方式降低模型的不确定度。为解决上述问题, 提出了不确定度学习层和不确定度损失项等概念, 设计了一种能够通过自学习的方式降低不确定度的神经网络架构 (ULNN), 从而增强神经网络模型预测的稳健性。在 CIFAR-10 和 CIFAR-100 数据集上的实验表明, ULNN 能够有效地降低模型不确定度, 在两个数据集上分别降低了 26 倍和 12 倍的不确定度。进一步在 CamVid 数据集上的语义分割实验中证明了 ULNN 的通用性。

关键词 光计算; 不确定度; 数据增强; 深度学习

中图分类号 TP183

文献标识码 A

doi: 10.3788/AOS201838.0620002

An Neural Network Framework of Self-Learning Uncertainty

Sun Hanqing, Pang Yanwei

School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China

Abstract In the multi-sensor fusion tasks of automatic drive, the strategy and the results of the data fusion are greatly influenced by the uncertainty of each subtask. To keep the whole system run steadily in multiple circumstances, the calculation model must operate with low uncertainty. The existing methods can only obtain uncertainty in the neural network prediction process, and few methods can reduce the uncertainty of the model in a self-learning method. To address the above problems, the concepts of uncertainty learning layer and uncertainty loss term are proposed, and a neural network architecture (ULNN) which can reduce uncertainty by self-learning method is designed to enhance the robustness of neural network model prediction. Experiments on CIFAR-10 and CIFAR-100 datasets show that ULNN can effectively reduce the model uncertainty and obtain 26 and 12 times lower uncertainty on the two data sets respectively. The universality of ULNN is proved by the experimental results of semantic segmentation on CamVid dataset.

Key words optics in computing; uncertainty; data augmentation; deep learning

OCIS codes 200.4260; 100.4996; 070.5010

1 引 言

深度神经网络已经在雷达数据处理^[1-3]、图像处理^[4-5]、图像识别^[6-9]和多传感器数据融合^[10-13]等方面取得了良好的进展。传统的神经网络方法不能直接获得模型的不确定度, 但能够获得不确定度的方法(如贝叶斯模型等)往往又难以计算^[14]。然而随着神经网络在增强学习、开放世界问题等领域的发

展, 越来越多的文献表明了获得模型不确定度的重要性^[15-17]。例如在自动驾驶系统或高级驾驶辅助系统(ADAS)中, 遇到光照条件不良的情况, 处理摄像头数据的神经网络不确定度较高, 输出结果不可靠, 相反处理雷达数据的神经网络仍能稳定工作, 处理结果具有较低的不确定度。这时, 决策系统就需要根据不确定度调整多传感器融合策略并进行决策。

为了求得不确定度, 需要在神经网络中引入随

收稿日期: 2017-12-29; **收到修改稿日期:** 2018-02-07

基金项目: 国家自然科学基金重点项目(61632081)

作者简介: 孙汉卿(1993—), 男, 硕士研究生, 主要从事模式识别和自动驾驶方面的研究。E-mail: HQSun@tju.edu.cn

导师简介: 庞彦伟(1976—), 男, 博士, 教授, 博士生导师, 主要从事模式识别、深度学习、子空间分析等方面的研究。

E-mail: pyw@tju.edu.cn

机性^[14],这种随机性在主流的神经网络架构中很常见,例如 Dropout^[18]操作和数据增强等都将引入随机性作为一种正则化手段。Dropout^[18]操作在神经网络一层的每个神经元上按伯努利分布随机置零,这种方法计算非常方便且效果显著,能够有效防止神经网络过拟合,已被广泛应用于多种神经网络中^[1-12,19-21]。Dropout有很多变体,例如使用高斯分布替代伯努利分布的乘性高斯噪声方法^[18],随机置零权重而非随机置零神经元的 DropConnect^[22],非随机置零的选择性 Dropout 方法^[23]等等。数据增强则比较适合分类等任务^[19],该方法需要对输入数据做随机的变换,这种变换不影响其预测结果,通过这种方式训练出来的神经网络模型对输入数据的变换更为稳健。

已经有方法利用上述随机性计算不确定度。考虑到 Dropout 在每一个神经元上采用伯努利分布随机置零,Gal Y 等^[14]将 Dropout 视为一种采样操作,给出了一种计算代价较小的求解神经网络不确定度的方式,其将不确定度定义为 Dropout 后多次预测结果的方差。Kendall A 等^[20]将文献^[14]的方法用于语义分割任务,成功地得到了逐像素的不确定度。Isobe S 等^[24]进一步使用文献^[14]的方法,将不确定度较大的部分定义为不确定类别,适合用于对误检比较敏感的任务上。

本文提出一种能够自学习地降低不确定度的神经网络架构(ULNN),其将不确定度作为神经网络训练优化目标的一部分。为了达到这一目标,引入了 ULNN 输入、不确定度学习(UL)层和不确定度

损失项等 ULNN 基本组成部分。作为一种可计算且容易实现的方法,ULNN 可用在诸如自动驾驶的多传感器数据融合等众多需要计算不确定度的任务场景;由于能够通过训练降低不确定度,使用 ULNN 框架训练的神经网络在上述多传感器数据融合等任务中可以提供置信度更高的推断结果。

2 自学习不确定度的神经网络

提出的自学习不确定度的神经网络架构(ULNN)主要组成部分包含 ULNN 输入、UL 层和不确定度损失项,本节将首先介绍 ULNN 的基本组成结构(2.1 小节),并给出 ULNN 输入(2.2 小节)、UL 层(2.3 小节)和不确定度损失项(2.4 小节)等基本组成部分的定义,最后在 2.5 小节中单独说明了这些组成部分引入的新参数。

2.1 自学习不确定度的神经网络的结构

图 1 是本课题组提出的自学习不确定度神经网络的基本结构,其基础是具有随机性的传统神经网络(如图 1 中灰色部分),其与传统神经网络结构的主要区别包括 ULNN 输入、UL 层和不确定度损失项(在图 1 中用蓝色标出)。

ULNN 输入的作用是在神经网络输入中构造关联数据,从而利用神经网络中已经存在的随机性来度量模型的不确定度;随后将构造好的输入数据传入传统神经网络;在传统神经网络中加入提出的 UL 层,该层计算不确定度的量化值,并通过不确定度损失项 βJ_{unc} 来自学习地降低神经网络的不确定度。

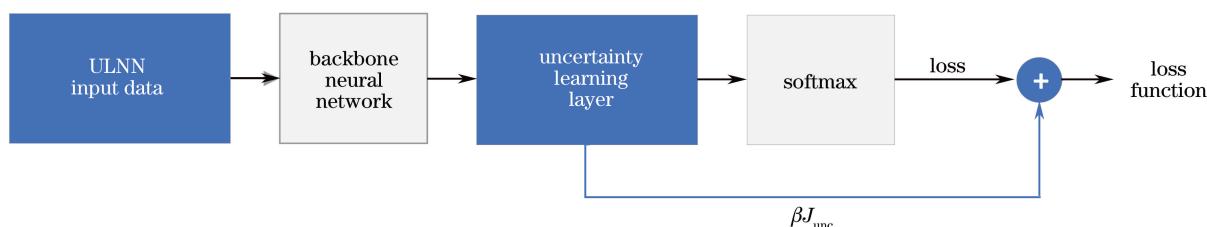


图 1 ULNN 的结构

Fig. 1 Architecture of ULNN

2.2 ULNN 的输入

根据输入数据处理方式不同,ULNN 输入可以分为两种:

第一种称为重复式 ULNN(ULNN-rep),即直接将输入数据重复多次,并传入神经网络中进行训练和预测。由于神经网络中存在随机性,多次输入可以产生多种不同的神经元响应值。

第二种称为增强式 ULNN(ULNN-aug),即需

要在训练阶段将输入数据随机进行多种数据增强,将结果作为新的输入数据,传入神经网络中进行训练,在预测阶段则使用与 ULNN-rep 相同的输入构造方式。

图 2 是两种 ULNN 变体输入数据构造方法的示意图:在 ULNN-rep[如图 2(a)所示]中,将原始数据复制多次(图中以 3 次为例,下同),并将得到的三个张量拼接成新的输入数据;而在 ULNN-aug

[如图 2(b)所示]中,将原始数据随机进行多次数据增强[图 2(b)中使用了随机平移、翻转作为示例],并将得到的三个张量拼接成新的输入数据。ULNN-rep 是一种直观的方法,但是由于这种输入等效于重复地将每个输入数据传入多次,容易导致神经网络过拟合,因此,使用 ULNN-aug 训练方法来克服这个问题。

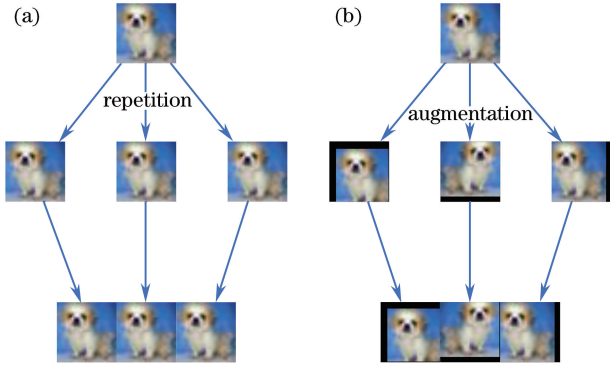


图 2 (a) ULNN-rep 输入数据构造方法;
(b) ULNN-aug 输入数据构造方法

Fig. 2 (a) Input data construction procedure of ULNN-rep;
(b) input data construction procedure of ULNN-aug

构造 ULNN 输入数据的过程可以看作是成倍增大输入数据的数量。在小批量随机梯度下降 (mini-batch SGD) 及其类似方法中,假设每批输入 n_{bat} 个数据,数据为 d 维张量,则输入数据为尺寸 $n_{\text{bat}} \times d$ 的矩阵;按照上述过程,在两种 ULNN 的变体中,输入数据的维度变为 $n_{\text{bat}} \times n_u \times d$,其中 n_u 表示 ULNN 输入重复次数。若令 $n = n_{\text{bat}} \times n_u$,将输入数据维度表示为 $n \times d$,就可以得到与传统神经网络输入数据相似的表达形式,因此 ULNN 网络前向传播过程与传统神经网络兼容,并且训练 ULNN 网络可以使用与训练传统神经网络一样的反向传播和小批量随机梯度下降等方法进行优化。

2.3 UL 层

UL 层是本文提出的 ULNN 架构的重要组成部分之一,UL 层的作用有两个:1) 计算和反向传播不确定度损失,2) 约减 ULNN 输入增加的维度。UL 层可以根据需要放置在神经网络的任意位置,与神经网络的其他层结构相似,UL 层也由输入、计算和输出三部分组成,其中计算过程包括计算不确定度损失和计算输入到输出映射两个部分。图 3 是对 UL 层几个部分的示意图。

根据 2.2 小节对 ULNN 输入数据的描述,ULNN 中 UL 层的输入是形为 $n_u \times d_{\text{feat}}$ 的张量,其中 d_{feat} 表示前层神经网络计算得到的输入特征张量

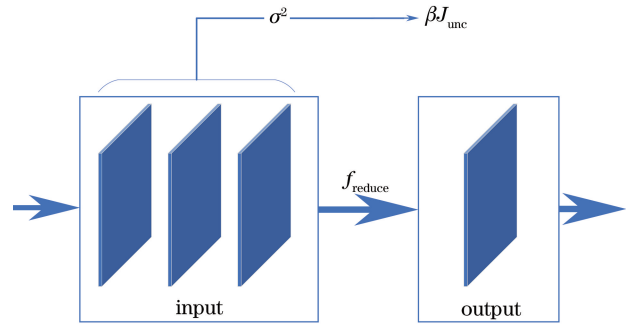


图 3 UL 层的组成

Fig. 3 Components of an UL layer

的维度(为表达简便,忽略了批数量 n_{bat})。而为了得到最终的数据标签,应如传统神经网络一样仅输出一个 $1 \times d_{\text{feat}}$ 的张量,这就需要映射 f_{reduce} 使得

$$\mathbf{O}_{1 \times d_{\text{feat}}} = f_{\text{reduce}}(\mathbf{I}_{n_u \times d_{\text{feat}}}), \quad (1)$$

式中下标表示张量维度, \mathbf{I} 为输入张量, \mathbf{O} 为输出张量。

在本文的实验中,选择了均值作为 UL 层输入到输出的映射关系,即

$$\mathbf{O}_{(1,k)} = \frac{1}{n_u} \sum_{i=1}^{n_u} \mathbf{I}_{(i,k)}, \quad (2)$$

式中括号下标 (\cdot, k) 表示张量元素的索引, $1 \leq k \leq d_{\text{feat}}$, n_u 为 ULNN 输入重复次数。

除了均值这种线性组合的形式,本文还进行了将 f_{reduce} 设置为随机映射和 Maxout^[25-26] 映射实验:随机映射即随机选择 UL 层输入数据中的一个作为输出;Maxout 映射是逐个神经元选取输入层中最大的元素构成输出。结果表明选择不同的映射对本文的 ULNN 结果并没有显著的影响,在实际使用中可以根据神经网络特性和需要选择合适的映射函数。

2.4 不确定度损失项

将不确定度作为优化目标的一部分是本文提出的 ULNN 的最主要特征,为了实现这一目标,本文将不确定度损失项 J_{unc} 引入传统神经网络的损失函数中,这样就能使用与传统神经网络一致的训练方法来降低不确定度。

按照 Gal Y 等^[14] 和 Kendall A 等^[20] 提供的方法,本文选择了 UL 层输入沿 ULNN 输入重复维度(即 n_u 维度)的方差的无偏估计量来度量神经网络的不确定度 \mathbf{U} ,其数学表达式为

$$\mathbf{U}_{(1,k)} = \frac{1}{n_u - 1} \sum_{i=1}^{n_u} \left[\mathbf{I}_{(i,k)} - \frac{1}{n_u} \sum_{j=1}^{n_u} \mathbf{I}_{(j,k)} \right]^2, \quad (3)$$

定义不确定度损失项 J_{unc} 为输入张量 \mathbf{I} 所有元素不确定度的均值,从而有

$$J_{\text{unc}} = \frac{1}{d_{\text{feat}}} \sum_{k=1}^{d_{\text{feat}}} \mathbf{U}_{(1,k)} = \frac{1}{d_{\text{feat}}} \sum_{k=1}^{d_{\text{feat}}} \frac{1}{n_u - 1} \left[\sum_{i=1}^{n_u} \mathbf{I}_{(i,k)} - \frac{1}{n_u} \sum_{j=1}^{n_u} \mathbf{I}_{(j,k)} \right]^2. \quad (4)$$

最终, ULNN 的损失函数可表示为

$$J(p, q; \theta) = H(p, q; \theta) + \beta J_{\text{unc}}(q; \theta) + \lambda \|\theta\|_2, \quad (5)$$

式中, β 称为 UL 权重(其作用将在 2.5 小节中详细说明), p 为训练集的概率分布, 是神经网络中待拟合的分布, q 为用来拟合 p 的概率分布, 是神经网络的训练结果, θ 表示神经网络的参数, $H(p, q; \theta)$ 为传统神经网络常用的交叉熵损失项, $\|\theta\|_2$ 为 θ 的二范数, 亦称权重衰减项, λ 为权重衰减系数。

2.5 自学习不确定度的神经网络的参数

上文提到, ULNN 包含两个参数: 输入重复次数 n_u 和 UL 权重 β 。ULNN 的一组输入数据是相关的, 输入重复次数 n_u 即表示输入层中相关数据的个数, 具体地, 在 ULNN-rep 中, n_u 表示每个输入数据重复的次数, 在 ULNN-aug 中, n_u 则表示每个输入数据增强的次数。 n_u 的设置需要考虑计算代价和网络特性, 这一参数对 ULNN 的影响将在实验部分给出(见表 2)。

UL 权重 β 与具体任务相关, 其反映了当前任务对不确定度的要求。对某一特定 ULNN 结构而言, β 越大, 则不确定度降低越快, 最终训练得到的模型不确定度越小(如图 4)。在本文的实验中, β 的默认值取 1, 不同 β 取值对模型不确定度影响也将在实验部分给出(见表 3)。

3 实验及结果

本文基于 DenseNet^[19] 进行不确定度自学习改进。DenseNet 是一种在 CIFAR^[27] 数据集上表现良好的卷积神经网络架构, 其主要特点是在一个基本组成块内, 每个卷积层的输出都是其后所有卷积层的输入, 多个输入使用张量连接操作合并。

DenseNet^[19] 结构有多种可选的配置, 本文重点在于用自学习方法降低模型的不确定度, 而非在数据集上取得最高的正确率, 因此使用了较简单的配置, 该配置有 1×10^6 个参数, 由三个块组成, 这一结构在 CIFAR-10 和 CIFAR-100 上分类正确率分别为 93.0% 和 72.5%, 在增强的 CIFAR-10 和 CIFAR-100 上分类正确率分别为 94.8% 和 75.6%。为对比 ULNN 的 UL 效果, 本文实验参数配置与 DenseNet 一致: 共训练 300 个循环, 初始学习率 0.1, 在第 150 个(50%)、第 225 个(75%)循环处学习率降低 10 倍, 使

用带动量的 Nesterov^[28] 优化方法, 动量为 0.9。由于本文的主要贡献在于用自学习方法降低不确定度, 因此没有专门根据本文的网络特点对训练参数进行调优。在本部分, 默认取 ULNN 重复次数为 3($n_u=3$), 不确定度权重为 1($\beta=1$), ULNN-aug 方法在 CIFAR-10 和 CIFAR-100 上分类正确率分别为 95.1% 和 75.9%, 此时不确定度分别为 0.0579 和 0.0753, 比训练得到的 DenseNet 原始不确定度分别下降了 26.2 倍和 12.4 倍。

为了证明所提 ULNN 的通用性, 本文也基于 Bayesian SegNet^[20,29] 神经网络架构进行了语义分割实验, 验证了所提出的 ULNN 框架能够有效地迁移到多种神经网络架构和多种任务中, 且能够自学习地降低不确定度。

3.1 ULNN 切片分析结果

为研究本文的 ULNN 各个部分的作用, 首先给出 ULNN 切片分析实验, 主要从三个角度进行分析: ULNN-aug 相对于 ULNN-rep 的优势; 输入重复次数 n_u 和 UL 权重 β 两个参数对不确定度的影响; 不同的 Dropout 比例对准确率的影响。

2.2 小节中提到, ULNN-rep 方法可能存在过拟合的问题, 于是本文提出 ULNN-aug 来克服这一问题。表 1 是使用重复输入但不使用不确定度损失项重新训练 DenseNet(即 $\beta=0$, 表 1 中 DenseNet-rep)、带不确定度损失项训练的 ULNN-rep($\beta=1$) 和 ULNN-aug($\beta=1$) 训练的结果, 全部取 Dropout 比例为 0.2(此处为了与 ULNN-rep 对比, 在数据增强的数据集上也使用了 Dropout, 但 DenseNet 原始结构用于数据增强的数据集时并未 Dropout, 所以表 1 中正确率略低于 DenseNet)。结果表明: 本文的不确定度惩罚项作为一种正则化方法, 能够一定程度上避免数据重复输入带来的性能下降; 而所提出的 ULNN-aug 方法则基本不会带来这种性能下降。

表 1 ULNN-rep 与 ULNN-aug 的对比

Table 1 Comparison between ULNN-rep and ULNN-aug

	CIFAR-10		CIFAR-100	
	Accuracy	Uncertainty	Accuracy	Uncertainty
DenseNet-rep	91.6%	0.0356	68.3%	0.1166
ULNN-rep	92.4%	0.0010	69.3%	0.0066
ULNN-aug	94.3%	0.0828	74.2%	0.1480

输入重复次数 n_u 是影响 ULNN 计算复杂度的重要参数, 为研究该参数对 ULNN 框架结果的影响, 进行了多组实验($\beta=1$, Dropout 比例为 0.05),

实验结果(如表 2 所示)表明重复次数并不是越大越好,在实际使用中,应当根据实际计算代价和不确定度量需求合理配置该参数,建议取 n_u 为 2 或 3 来控制计算复杂度。

表 2 输入重复次数 n_u 的影响

Table 2 Impact of input repetition number n_u

n_u	CIFAR-10		CIFAR-100	
	Accuracy	Uncertainty	Accuracy	Uncertainty
2	94.8%	0.0579	75.9%	0.0753
3	94.9%	0.0659	75.6%	0.0762
4	95.0%	0.0717	75.7%	0.0743
5	94.8%	0.0761	75.6%	0.0740

ULNN 的另一个参数是 UL 权重 β ,正如 2.3 小节中介绍的, β 影响最终模型的不确定度,因此需要根据不同的任务和需求进行设置。图 4 给出在 ULNN 训练过程中, β 对不确定度降低和收敛速度的影响(Dropout 比例为 0.2,为使曲线清晰,纵轴使用了以 10 为底的对数坐标);表 3 给出了训练的数值结果($n_u=3$)。需要说明的是,由于不确定度惩罚项 J_{unc} 对 UL 层输入的偏导包含系数 $2/n_u$ (推导请参考第 4 节),因此表 3 中使用了 $\beta=n_u/2=1.5$ 的权重作为一组对照。但在实际使用中,只要 n_u 确定,该系数就为一确定常数,所以本文没有单独强调这一系数,而是将其统一作为 UL 权重 β 的一部分来处理。

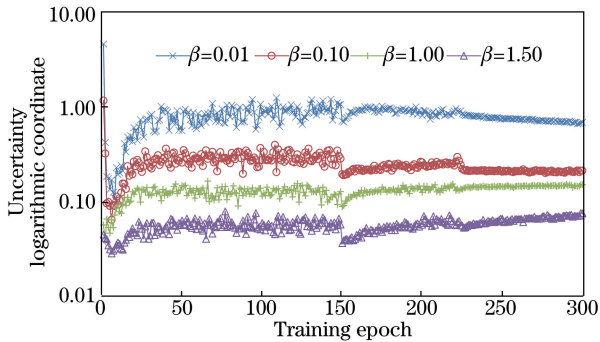


图 4 不同 β 的不确定度学习曲线 (CIFAR-10,纵轴为对数坐标)

Fig. 4 Curves of uncertainty learning with different β s (on CIFAR-10, vertical-axis is a logarithmic coordinate)

表 3 UL 权重 β 的影响

Table 3 Impact of UL weight β

β	CIFAR-10	CIFAR-100
$1.5(=n_u/2)$	0.0718	0.1076
1	0.0828	0.1480
0.1	0.2107	0.8285
0.01	0.6936	2.4774

另外,在 ULNN-aug 中,由于在随机数据增强

时就已经引入了随机性,因此在 ULNN-aug 方法中可以根据实际情况减小 Dropout 比例,表 4 是以 DenseNet 为基础的 ULNN-aug 方法在不同 Dropout 比例下的结果。

表 4 Dropout 对 ULNN-aug 方法的影响

Table 4 Impact of Dropout on ULNN-aug

Dropout ratio	CIFAR-10		CIFAR-100	
	Accuracy	Uncertainty	Accuracy	Uncertainty
0.2	94.8%	0.0757	75.2%	0.1417
0.1	94.9%	0.0687	75.6%	0.1077
0.05	95.1%	0.0579	75.9%	0.0753

3.2 ULNN 的总体性能

根据切片分析的结果,本文选择 ULNN-aug,并设置 ULNN 重复次数为 3($n_u=3$),不确定度权重为 1($\beta=1$),其他参数与 DenseNet^[19]一致,得到了如表 5 所示的结果(在测试时使用 Dropout 比例 0.05 以计算不确定度),在 CIFAR-10 和 CIFAR-100 上分类正确率略有提高的条件下,分别将不确定度降低了 26.2 倍和 12.4 倍。进一步结合 DenseNet 和 ULNN 在 CIFAR-10 上的不确定度曲线(如图 5),可以明显看到 ULNN-aug 方法不确定度远低于使用数据增强的原始 DenseNet,也不会随着训练次数快速增长(在 150 个循环后,学习率下降 10 倍,但原始 DenseNet 的不确定度却开始快速增大),从而证明了本文的自学习降低不确定度的有效性。

表 5 使用 DenseNet 原始训练参数的验证结果对比

Table 5 Comparison of validation result with original training parameters in DenseNet

	CIFAR-10		CIFAR-100	
	Accuracy	Uncertainty	Accuracy	Uncertainty
DenseNet + data aug.	94.8%	1.5190	75.6%	1.3564
ULNN-aug	95.1%	0.0579	75.9%	0.0753

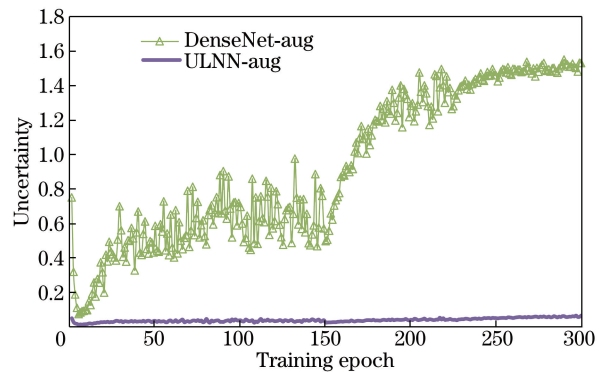


图 5 UL 曲线(CIFAR-10)

Fig. 5 Curves of UL (CIFAR-10)

上述实验结果表明,本文的自学习不确定度的神经网络架构能够以可控的计算代价,在获得不确定度的同时,有效地使用自学习的方法降低神经网络在训练和预测阶段的不确定度,且不会降低原始神经网络的准确率。

3.3 ULNN 框架的通用性

为了证明 ULNN 的通用性,实验基于 Bayesian SegNet^[20-29] 架构在 CamVid 公开数据集上进行了语义分割实验。

Bayesian SegNet^[20] 是基于 SegNet^[29] 改进的,二者网络架构和训练过程相同,且都不能降低不确定度;二者区别在于 Bayesian SegNet 在预测阶段与本文一样使用了文献[14]提出的方法求解不确定度。为了公平比较,使用本文 ULNN-aug 方法,选用与 Bayesian SegNet^[20,29] 一致的训练参数,考虑训练和测试速度,取 $n_u = 3$,训练得到 Bayesian SegNet 和 ULNN SegNet($\beta = 1$)。表 6 是详细结果,其中平均交并比(mIoU,数值越大越好)是语义分割任务中常用的评价指标,结果显示在正确率和 mIoU 有所提高的条件下,ULNN 使得 Bayesian SegNet 不确定度下降了 9 倍以上。

上述实验结果表明:本文提出的 ULNN 作为一种使用自学习的方法降低神经网络不确定度的框架,具有通用性,在多种任务和神经网络架构上都能有效降低模型不确定度,且相比已有的方法能够得到更低的不确定度。

表 6 ULNN SegNet 在 CamVid 上的语义分割结果

Table 6 Semantic segmentation results of

ULNN SegNet on CamVid

	Accuracy	mIoU	Uncertainty
Bayesian SegNet	85.9%	51.4%	452.7
ULNN SegNet	87.1%	52.0%	50.63

4 分析与讨论

本节将从梯度反向传播的角度分析所提不确定度损失项的作用,不确定度损失项是所提 ULNN 的主要贡献和特点,其与现阶段常用的反向传播和 mini-batch SGD 等方法兼容,从而可以直接复用现有的优化方法来实现。

$\forall x \in \mathcal{X}$, \mathcal{X} 为训练集,不确定度损失项 J_{unc} 对 UL 层第 i 个输入张量 $\mathbf{I}_{(i)}$ 的偏导为

$$\frac{\partial J_{\text{unc}}}{\partial \mathbf{I}_{(i)}} = \frac{2}{n_u} (\mathbf{I}_{(i)} - \bar{\mathbf{I}}) - \frac{2}{n_u(n_u - 1)} \sum_{j=1, j \neq i}^{n_u} (\mathbf{I}_{(j)} - \bar{\mathbf{I}}) = \frac{2}{n_u} \mathbf{I}_{(i)} - \frac{2}{n_u(n_u - 1)} \sum_{j=1, j \neq i}^{n_u} \mathbf{I}_{(j)}, \quad (6)$$

提出系数,可将(6)式重写为

$$\frac{n_u}{2} \frac{\partial J_{\text{unc}}}{\partial \mathbf{I}_{(i)}} = \mathbf{I}_{(i)} - \frac{1}{n_u - 1} \sum_{j=1, j \neq i}^{n_u} \mathbf{I}_{(j)}, \quad (7)$$

那么在神经网络梯度下降的每次迭代中,为了使 J_{unc} 减小,就应令

$$\mathbf{I}_{(i)} \leftarrow \mathbf{I}_{(i)} - \Delta \cdot \frac{2}{n_u} \cdot \left[\mathbf{I}_{(i)} - \sum_{j=1, j \neq i}^{n_u} \mathbf{I}_{(j)} / (n_u - 1) \right], \quad (8)$$

式中 Δ 是梯度下降的迭代步长,由(8)式中括号第二项是与 $\mathbf{I}_{(i)}$ 同组的其他特征张量 $\mathbf{I}_{(j)}$ ($j = 1, \dots, i-1, i+1, \dots, n_u$) 的均值形式可以看出:在训练过程中,对 ULNN 输入层的某一个输入特征张量,其在数值上趋向于同组其他特征张量的均值。这样,在神经网络引入随机性的条件下,ULNN 能够使随机性带来的神经网络预测值变化更小,也就增强了神经网络模型的稳健性,使之对输入的微小变化不敏感。

5 结 论

分析和实验表明,ULNN 架构能够有效地使用自学习的方法降低神经网络模型的预测不确定度,且实现容易,与现有神经网络训练方法兼容,适合于众多需要计算不确定度的任务场景。由于能够自学习地降低不确定度,使用 ULNN 框架训练的神经网络比传统神经网络稳健性更强,可以提供置信度更高的推断结果。

该方法适合于大多数存在随机性的神经网络,正如实验部分所演示的,通过合理使用 ULNN 的几个基本组成部分,可用于改造多数已存在的神经网络,使之成为能够自学习地降低不确定度的神经网络。对神经网络模型不确定度的其他度量方法和所提方法的拓展和应用仍然是未来研究的重点。

参 考 文 献

- [1] Matti D, Ekenel H K, Thiran J P. Combining LiDAR space clustering and convolutional neural networks for pedestrian detection[C]. Lecce: IEEE International Conference on Advanced Video and Signal Based Surveillance, 2017: 1-6.
- [2] Liu C H, Li Z, Xu C, *et al.* BRDF model for commonly used materials of space targets based on deep neural network[J]. Acta Optica Sinica, 2017, 37(11): 1129001.
刘程浩, 李智, 徐灿, 等. 基于深度神经网络的空间目标常用材质 BRDF 模型[J]. 光学学报, 2017, 37(11): 1129001.

- [3] Lu Y S, Li Y X, Liu B, *et al.* Hyperspectral data haze monitoring based on deep residual network [J]. *Acta Optica Sinica*, 2017, 37(11): 1128001.
陆永帅, 李元祥, 刘波, 等. 基于深度残差网络的高光谱遥感数据霾监测 [J]. *光学学报*, 2017, 37(11): 1128001.
- [4] Li S M, Lei G Q, Fan R. Depth map super-resolution reconstruction based on deep convolutional neural networks [J]. *Acta Optica Sinica*, 2017, 37(12): 1210002.
李素梅, 雷国庆, 范如. 基于卷积神经网络的深度图超分辨率重建 [J]. *光学学报*, 2017, 37(12): 1210002.
- [5] Xiao J S, Liu E Y, Zhu L, *et al.* Improved image super-resolution algorithm based on convolutional neural network [J]. *Acta Optica Sinica*, 2017, 37(3): 0318011.
肖进胜, 刘恩雨, 朱力, 等. 改进的基于卷积神经网络的图像超分辨率算法 [J]. *光学学报*, 2017, 37(3): 0318011.
- [6] Pang Y, Sun M, Jiang X, *et al.* Convolution in convolution for network in network [J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2017, 47(1): 1-11.
- [7] Cao J, Pang Y, Li X, *et al.* Randomly translational activation inspired by the input distributions of ReLU [J]. *Neurocomputing*, 2017.
- [8] Hou C C, He Y Q, Jiang X H, *et al.* A deep convolutional neural network based on two-stream convolutional unit [J]. *Laser & Optoelectronics Progress*, 2018, 55(2): 021005.
侯聪聪, 何宇清, 姜晓恒, 等. 基于二分支卷积单元的深度学习神经网络 [J]. *激光与光电子学进展*, 2018, 55(2): 021005.
- [9] Cao J, Pang Y, Li X. Learning multilayer channel features for pedestrian detection [J]. *IEEE Transactions on Image Processing*, 2017, 26(7): 3210-3220.
- [10] Morchhale S, Pauca V P, Plemmons R J, *et al.* Classification of pixel-level fused hyperspectral and lidar data using deep convolutional neural networks [C]. Los Angeles: Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, 2016: 1-5.
- [11] Zhao G, Xiao X, Yuan J, *et al.* Fusion of 3D-LIDAR and camera data for scene parsing [J]. *Journal of Visual Communication and Image Representation*, 2014, 25(1): 165-183.
- [12] Du X, Ang M H, Rus D. Car detection for autonomous vehicle: LIDAR and vision fusion approach through deep learning framework [C]. Vancouver: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2017: 749-754.
- [13] Dieterle T, Particke F, Patino-Studencki L, *et al.* Sensor data fusion of LIDAR with stereo RGB-D camera for object tracking [A]. Glasgow: 2017 IEEE Sensors, 2017: 1-3.
- [14] Gal Y, Ghahramani Z. Dropout as a Bayesian approximation: representing model uncertainty in deep learning [C]. New York: Proceedings of the 33rd International Conference on Machine Learning, 2016, 48: 1050-1059.
- [15] Krzywinski M, Altman N. Points of significance: importance of being uncertain [J]. *Nature Methods*, 2013, 10(10): 809-810.
- [16] Ghahramani Z. Probabilistic machine learning and artificial intelligence [J]. *Nature*, 2015, 521(7553): 452.
- [17] Teerapittayanon S, McDanel B, Kung H T. BranchyNet: fast inference via early exiting from deep neural networks [C]. Cancun: 23rd International Conference on Pattern Recognition, 2016: 2464-2469.
- [18] Srivastava N, Hinton G, Krizhevsky A, *et al.* Dropout: a simple way to prevent neural networks from overfitting [J]. *Journal of Machine Learning Research*, 2014, 15: 1929-1958.
- [19] Huang G, Liu Z, van der Maaten L, *et al.* Densely connected convolutional networks [C]. Honolulu: IEEE Conference on Computer Vision and Pattern Recognition, 2017: 2261-2269.
- [20] Kendall A, Badrinarayanan V, Cipolla R. Bayesian SegNet: model uncertainty in deep convolutional encoder-decoder architectures for scene understanding [J]. *Computer Science*, 2015: 1511.
- [21] He K, Zhang X, Ren S, *et al.* Deep Residual Learning for Image Recognition [C]. Las Vegas: 2016 IEEE Conference on Computer Vision and Pattern Recognition, 2016: 770-778.
- [22] Wan L, Zeiler M, Zhang S, *et al.* Regularization of neural networks using dropconnect [C]. Atlanta: Proceedings of the 30th International Conference on Machine Learning, 2013, 28: 1058-1066.
- [23] Barrow E, Eastwood M, Jayne C. Selective dropout for deep neural networks [C]. Kyoto: Proceedings of the 23rd International Conference on Neural Information Processing, 2016: 519-528.
- [24] Isobe S, Arai S. Deep convolutional encoder-decoder network with model uncertainty for semantic segmentation [C]. IEEE International Conference on Innovations in Intelligent Systems and Applications, 2017: 365-370.

- [25] Goodfellow I, Warde-Farley D, Mirza M, *et al.* Maxout networks [C]. Atlanta: 2013 International Conference on Machine Learning, 2013, 28: 1319-1327.
- [26] Chang J-R, Chen Y-S. Batch-normalized maxout network in network [EB/OL]. [2018-4-24]. <https://arxiv.org/abs/1511.02583>.
- [27] Krizhevsky A, Hinton G E. Learning multiple layers of features from tiny images[R]. CiteSeerX, 2009.
- [28] Nesterov Y E. A method for solving the convex programming problem with convergence rate $O(1/k^2)$ [J]. Soviet Math Dokl, 1983, 269(3): 543-547.
- [29] Badrinarayanan V, Kendall A, Cipolla R. SegNet: a deep convolutional encoder-decoder architecture for scene segmentation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(12): 2481-2495.