

# 前列腺癌辅助诊断 GPU 并行算法设计

苏庆华<sup>1</sup> 付景超<sup>1</sup> 谷 焱<sup>2,3</sup> 张姗姗<sup>2,3</sup> 李奕飞<sup>2,3</sup> 江方舟<sup>2,3</sup> 白翰林<sup>1</sup> 赵 地<sup>2</sup>  
(北京物资学院信息学院 北京 101149)<sup>1</sup> (中国科学院计算技术研究所 北京 100190)<sup>2</sup>  
(北京邮电大学 北京 100089)<sup>3</sup>

**摘要** 在癌症高发的当代,前列腺癌作为男性特有的疾病,其发病率逐年升高。卷积神经网络因其在图像识别领域的强大性能而倍受关注,也非常适用于计算机辅助诊断(Computer Aided Design,CAN)领域。由于神经网络模型中通常包含大量参数,因此训练一个卷积神经网络十分耗时。如何加快神经网络的训练成为了深度学习领域中一个十分重要的问题。为了解决这个问题,一般采用多 GPU 并行方案。其中,数据同步在 GPU 性能均衡的情况下表现更佳。因此,文中借鉴已有的基于数据并行算法对前列腺三维卷积网络进行加速。

**关键词** 卷积网络,深度学习,多 GPU 并行,数据并行,神经网络

中图分类号 G312 文献标识码 A

## Parallel Algorithm Design for Assisted Diagnosis of Prostate Cancer

SU Qing-hua<sup>1</sup> FU Jing-chao<sup>1</sup> GU Han<sup>2,3</sup> ZHANG Shan-shan<sup>2,3</sup> LI Yi-fei<sup>2,3</sup>  
JIANG Fang-zhou<sup>2,3</sup> BAI Han-lin<sup>1</sup> ZHAO Di<sup>2</sup>  
(Information Engineering School, Beijing Wuzi University, Beijing 101149, China)<sup>1</sup>  
(Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China)<sup>2</sup>  
(Beijing University of Posts and Telecommunications, Beijing 100089, China)<sup>3</sup>

**Abstract** In the contemporary era of high cancer, prostate cancer is a unique disease for men, and the incidence is increasing year by year. Convolutional neural networks have attracted much attention due to their powerful performance in the field of image recognition, and are also very suitable for computer-aided diagnosis. Training a convolutional neural network is time consuming because neural network models often contain a large number of parameters. How to accelerate the training of neural networks has become a very important issue in the field of deep learning. To solve this problem, a multi-GPU parallel scheme is generally adopted. Among them, data synchronization performs better when the GPU performance is balanced. Therefore, this paper draws on the algorithm based on data parallel to accelerate the three-dimensional convolution network of prostate.

**Keywords** Convolutional network, Deep learning, Multi-GPU parallelism, Data parallelism, Neural network

## 1 引言

前列腺癌是男性泌尿生殖系统较为普遍的恶性肿瘤之一。欧盟癌症死亡率预测表明前列腺癌死亡率位居前三,我国前列腺癌发病率虽低于欧美,但近年来也呈明显上升趋势<sup>[1]</sup>。因为计算机视觉模型可以快速扫描所有具有相同质量的地方,并且不会受到人为因素(如疲劳和情绪)的影响,因此 CAD 适用于前列腺癌诊断。

当前,提高神经网络训练效率的方法主要有两种:1)优化模型算法;2)硬件加速。优化模型算法实际上对训练效率的提升十分有限,且只能根据具体模型的不同来调整其算法,可移植性较差。相比之下,由于图形处理器(Graphic Processing Unit,GPU)在并行处理、浮点计算、带宽等方面的优良特性,GPU 已经取代中央处理器(Central Processing Unit,CPU)成

为支撑神经网络训练的主要硬件。

本文首先对卷积神经网络进行分析,提出了基于多 GPU 的数据并行方案,来解决网络模型训练效率低的问题,实现更高效的多 GPU 并行加速。

## 2 多 GPU 下网络模型的并行算法

### 2.1 多 GPU 并行

在传统的计算机中,CPU 与内存在构造上是相互独立的,CPU 通过内存地址来访问数据等,这就意味着在单机多卡(这里的多卡指的是多 CPU,后文的多卡一般多指多 GPU)情况下不同的 CPU 可以直接访问同一块内存空间,这样的构造可以快速地实现不同 CPU 之间的通信。但与 CPU 不同的是,GPU 拥有独立的板载显存和片上存储空间,GPU 与 GPU

本文受国家自然科学基金(61803035)资助。

苏庆华(1980-),女,博士,副教授,主要研究方向为医学影像处理,E-mail:qinghuasu@126.com;赵 地(1978-),男,博士,副教授,CCF 会员,主要研究方向为类脑计算、深度学习,E-mail:zhaodi@escience.cn(通信作者)。

之间无法直接共享存储信息,这就导致在多 GPU 并行训练时需要花费额外的时间来进行数据拷贝操作。

多 GPU 间通信通过总线来传输,这就意味着其传输效率受限于总线带宽。以 PCI-E3.0 为例,16 通道时的理论双向带宽是 32GB/s,而 Tesla K80 显存带宽为 480GB/s,相比之下 PCI-E 提供的数据传输能力远远小于 GPU 的计算能力。

总的来说,GPU 并行可以提供十分强大的计算能力,但是多 GPU 间的通信成本较高。因此,在使用多 GPU 并行时,需要尽量为 GPU 分配足够的计算量,以提高 GPU 的利用率。

## 2.2 数据并行

数据并行就是在不同 GPU 上使用相同的网络模型,将数据集划分到不同的 GPU 上,用不同的数据在多卡上训练相同的网络模型,每个 GPU 都能独立完成前后向计算。

一般地,如果在单卡上可以实现一个神经网络的训练,就可通过修改模型获取方式和更新数据方式,将该网络模型以数据并行的方式拓展到多卡上。因此,往往数据并行比模型并行更具有拓展性和可行性。

在不同的 GPU 上采用不同的训练样本,需要对不同 GPU 上的模型进行参数更新。所以,在实现数据并行时最重要的问题就是如何在保证模型收敛的前提下完成其参数更新。

### 2.2.1 归约树结构

归约树结构如图 1 所示,通过特定算法,将多个 GPU 构造成一棵归约树。

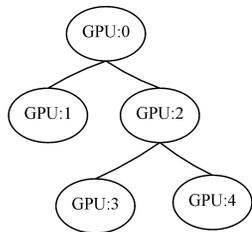


图 1 归约树结构示意图

在一棵完全归约树中,节点的个数为 2 的幂次方。开始时每个节点都是一棵单独的树,且这些树的根节点就是其自身。经过多次循环,每次循环时将当前所有树两两配对,然后将每一对树中的第二棵树作为第一棵树根节点的最右子节点,加入后得到一颗新树。通过这种方式迭代合并,直到所有节点都在一棵树上后,就完成了归约树的构造。由于这种特殊的构造方式,在一个归约树的子节点归约到相应的父节点之后,得到的仍然是一棵归约树。因此数据经过多次重复之后,就可以将所有节点的数据归约到根节点中。

以归约树结构实现数据并行,从根节点开始,每个节点将最新的网络模型发送给子节点,子节点进行相应的参数更新。每完成一次前后向计算,从子节点开始,每个节点将通过相应计算得到的结果累加以得到输出数据,将输出数据发送给父节点,最后再由根节点汇总,用于网络模型参数的更新。这种方案借助归约的思想分散了网络更新操作的计算量。

每次迭代时整体权值进行更新操作的时间降为  $\log_2^{n \cdot t_2}$ ,相应的,叶节点需要更长的时间来与根节点进行同步,通信耗时增至  $\log_2^{n \cdot t_3}$ 。在该算法中,一个节点单次迭代的时间为:

$$t = t_1 + \log_2^{n \cdot t_2} + \log_2^{n \cdot t_3} \quad (1)$$

在归约树中,不同深度的节点,参与归约的次数也不同。例如,叶节点只需要参与一次归约,而根节点需要参与  $\log_2^n$  次归约。对应到神经网络中,叶节点的 GPU 只需要传输一次  $\Delta w$ ,不需要进行更新操作,而根节点的 GPU 需要进行  $\log_2^n$  次的更新。所以,归约树结构本质上只是缓解了更新操作分配不均匀的问题,并没有从根本上解决各 GPU 负载率不均衡的问题。

### 2.2.2 C/S 结构

C/S 结构(Client/Server 结构)即客户端/服务器结构, GPU 为 Client, CPU 为 Server。

C/S 结构有两种实现方法:1)数据同步并行,如图 2 所示;2)数据异步并行,如图 3 所示。

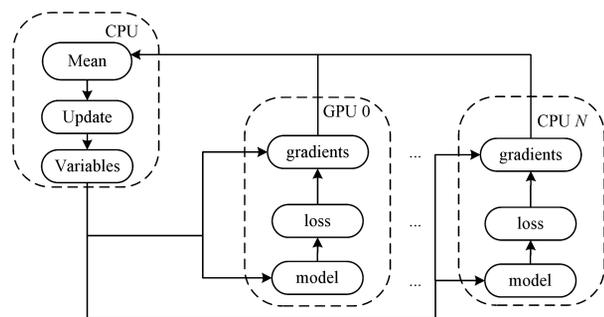


图 2 数据同步并行示意图

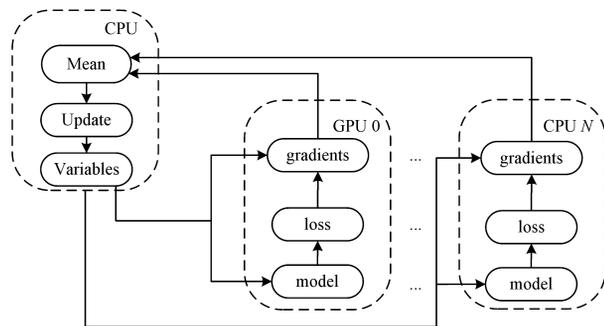


图 3 数据异步并行示意图

数据同步并行是指在一次迭代开始后,每个 GPU(从节点)都从主节点(一般为 CPU)获取最新的网络模型,使用不同的数据进行训练,然后将得到的  $\Delta w$  传回主节点。当主节点将所有节点传回的  $\Delta w$  都更新到模型之后,系统的一次迭代完成,则模型的一次更新完成。每个 GPU 会从主节点获取更新后的网络模型。假设系统中包含  $m$  个 GPU,使用的 mini-batch 大小为  $N$ ,则等价于利用单个 GPU 对大小为  $m \cdot N$  的 mini-batch 进行训练。

数据异步并行是指在第一次迭代开始后,每个 GPU 从主节点获取网络模型,利用不同的数据进行训练,将各自得到的  $\Delta w$  传回主节点后,立即利用当前主节点中的  $\Delta w$  对该 GPU 中的当前网络进行更新。由于每个 GPU 只与主节点进行数据同步,故只有在 GPU 性能差异较大的情况下,数据异步并行才会有更好的鲁棒性。

由于本次实验所用的 GPU 型号相同,故在无故障发生的情况下可默认性能相同。在训练数据批次大小、GPU 性能都相同时,选择多 GPU 数据同步并行更为合适,故选择数据同步并行实现 C/S 结构,可以更好地提升训练效率。

### 3 实验与分析

#### 3.1 实验平台与实验设计

实验服务器所使用系统为 Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-96-generic x86\_64), 连接服务器的 ssh 软件为 putty。硬件配置为 40 块 Intel Xeon E5-2640 v4 CPU, 以及 NVIDIA Tesla K80 GPU。

实验思路: 在实验中, 基于多 GPU 数据同步并行的算法使用不同数量的 GPU 来训练卷积神经网络。记录在 GPU 数量不同时整个训练过程的耗时情况以及最终准确率。从多 GPU 加速比和最终模型准确率两个方面, 分析基于多 GPU 数据同步并行算法的拓展性。

#### 3.2 ProstateX 数据集上的实验

##### 3.2.1 ProstateX 数据集

SPIE-AAPM-NCI Prostate MR Gleason Grade Group Challenge 是由美国医学物理学家协会 (AAPM) 与 SPIE (The International Society for Optics and Photonics) 和美国国家癌症研究所 (NCI) 联合发起的一个医学影像识别比赛。

该数据集总共包含了 162 个 MRI 病例, 每个病例都来自于一个不同的患者。每个病例由 4 组 MRI 扫描数据组成: 两组 T2 加权图像 (横轴和矢状图, DICOM 格式)、Ktrans 图像 (由动态对比增强 (DCE) 图像计算, mhd 格式) 和视扩散系数 (ADC) 图像 (由扩散加权 (DWI) 成像计算, DICOM 格式)。这些病例总共包含 182 个病灶; 训练集包含 112 个病灶, 而测试集包含 70 个病灶。每个病灶都提供了位置和参考缩略图的图像, 且都有已知的病理定义的 Gleason 等级。

DICOM 格式是医学影像常用的格式, 文中查看 DICOM 文件的软件为 RediAnt DICOM Viewer。

由于数据集过小的问题, 训练而来的神经网络可能会产生过拟合, 因此我们利用镜像翻转、椒盐加噪、高斯加噪等方式将数据量扩增到了 16000+, 虽然数据量仍然稍显不足, 但利用 dropout 等正则化方法可以有效防止模型的过拟合问题。

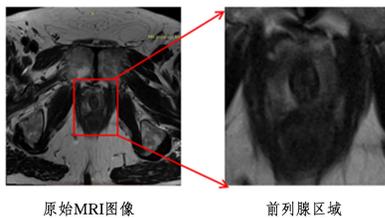


图4 前列腺区域图像

##### 3.2.2 训练参数

AlexNet 是一个具有代表性的卷积神经网络, 其模型的设计思想至今仍被不断模仿。本节实验模型借鉴了 AlexNet。

在训练过程中, 输入层数据规模为  $4 * 1 * 25 * 224 * 224$  ( $\text{mini\_batch\_size} * \text{channel} * \text{deep} * \text{height} * \text{width}$ )。第 1 层为卷积层, 卷积核大小为  $3 * 11 * 11$ , 卷积核移动步长为  $2 * 4 * 4$ , 对边界进行补充, 输出 32 个特征图, 并使用 ReLU 函数作为激活函数。第 2 层为池化层, 采样窗口的大小为  $2 * 3 * 3$ , 窗口滑动步长为  $2 * 2 * 2$ , 池化方式为最大池化。第 3 层为卷积层, 卷积核大小为  $3 * 5 * 5$ , 卷积核移动步长为  $1 * 1 * 1$ , 对边界进行补充, 输出 64 个特征图, 并使用 ReLU 函数作为激活函数。第 4 层为池化层, 与第一个池化层参数设置一

样。第 5 层为卷积层, 卷积核大小为  $2 * 3 * 3$ , 卷积核移动步长为  $1 * 1 * 1$ , 对边界进行补充, 输出 128 个特征图, 并使用 ReLU 函数作为激活函数。第 6 层为卷积层, 卷积核大小为  $3 * 3 * 3$ , 卷积核移动步长为  $1 * 1 * 1$ , 对边界进行补充, 输出 128 个特征图, 并使用 ReLU 函数作为激活函数。第 7 层为卷积层, 卷积核大小为  $3 * 3 * 3$ , 卷积核移动步长为  $1 * 1 * 1$ , 对边界进行补充, 输出 64 个特征图, 并使用 ReLU 函数作为激活函数。第 8 层为池化层, 与第一个池化层参数设置一样。第 9 层为全连接层, 输出节点为 2048 个, 使用 ReLU 函数作为激活函数, 且使用 dropout 方法进行正则化。第 10 层为全连接层, 输出节点 1024 个, 使用 ReLU 函数作为激活函数, 且使用 dropout 方法进行正则化。最后一层为全连接层, 输出节点为 5 个, 使用 softmax 函数实现分类。

其中, ReLU 函数如式 (2) 所示。该函数收输入值  $x$  后, 当输入值  $x$  大于零, 则输出等于输入  $x$ ; 当输入值  $x$  小于等于 0, 则输出为 0。

$$\text{ReLU}(x) = \begin{cases} x, & x > 0 \\ 0, & x \leq 0 \end{cases} \quad (2)$$

ReLU 函数图像如图 5 所示。

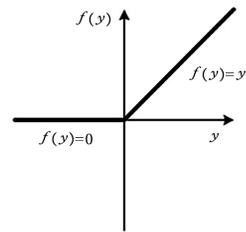


图5 ReLU 函数图像

其中, softmax 函数见式 (3)。softmax 函数用于多分类过程中接收多个输入值后, 将接收的多个输入值映射到 (0, 1) 区间中, 从而实现分类。

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad (3)$$

整个训练过程迭代 30000 次, 基础学习率为 0.01, 并在训练过程中随着迭代次数增加而减小, 下限为  $1 \times 10^{-3}$ 。

#### 3.3 实验分析

下列 3 张图展示了在多 GPU 数据同步并行方案下, 使用不同 GPU 训练卷积神经网络时的性能。

图 6 清楚的表明, 由于采用了平均梯度, 实质上减少了模型的更新次数, 对最终的准确率产生了影响。但是基于随机梯度下降的训练具有较好的鲁棒性, 其准确率始终在 72.5% 上下浮动, 即使是在使用 12 个 GPU 进行训练时, 最终模型的准确率几乎不变。

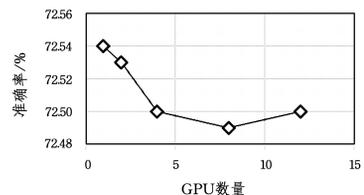


图6 GPU 数量与准确率的关系

图 7 展示了呈线性增长的加速比。对比于优秀的加速比, 损失的精确度是值得的。

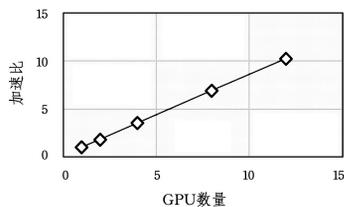


图7 GPU数量与加速比的关系

我们的结论如下:采用多 GPU 数据并行方案,可以在精度损失极小的情况下,有效地加快卷积神经网络的训练过程,有效地提高了卷积神经网络的训练效率;且其优秀的拓展性使其并不局限于固有的模型,可更换模型进行训练,加快了卷积神经网络的训练过程,更好地提高了训练效率。

图 8 中的实验结果表明,GPU 数量的提升并没有显著降低单 GPU 的使用率。

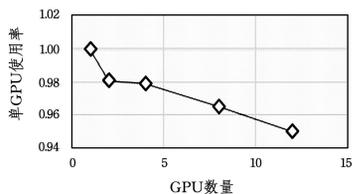


图8 GPU数量与单GPU使用率的关系

**结束语** 本文针对多 GPU 上卷积神经网络的实现,分析了模型并行和数据并行这两种主要的并行思路,并详细分析了数据并行中 C/S 结构的同步并行方案。针对网络训练效率低的问题,提出了多 GPU 数据同步并行方案来实现网络模型的训练。实验结果表明,相对于单 GPU,该算法可以取得很好的并行加速比。但该算法仍然具有一定的局限性。

GPU 传输总线的传输速率远远小于其计算能力,在 GPU 负载率低的情况下,用于 GPU 间通信的时间所占比例会进一步提升,这就意味着训练效率的降低,我们所需要做的是尽可能使 GPU 负载量满足要求,充分利用 GPU 的计算能力。同时,也可以考虑参考利用其他结构进行实验,通过对比训练也许可以得到更好的结果。

## 参考文献

[1] 曹德宏,柳良仁,魏强,等.前列腺癌的治疗研究进展[J]. 华西医学,2017(2):277-281.

[2] SHIN H C,ORTON M R,COLLINS D J,et al. Stacked Autoencoders for Unsupervised Feature Learning and Multiple Organ Detection in a Pilot Study Using 4D Patient Data[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence,2013,35(8):1930-1943.

[3] ESTEVA A,KUPREL B,NOVOA R A,et al. Corrigendum: Dermatologist-level classification of skin cancer with deep neural networks[J]. Nature,2017,542(7639):115-118.

[4] GULSHAN V,PENG L,CORAM M,et al. Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs[J]. Jama,2016,316(22):2402.

[5] LITJENS G,KOOI T,BEJNORDI B E,et al. A survey on deep learning in medical image analysis[J]. Medical Image Analysis,2017,42(9):60-88.

[6] 周飞燕,金林鹏,董军. 卷积神经网络研究综述[J]. 计算机学报,2017,40(6):1229-1251.

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

[8] KRIZHEVSKY A,SUTSKEVER I,HINTON G E. ImageNet classification with deep convolutional neural networks[C]// International Conference on Neural Information Processing Systems. Curran Associates Inc.,2012:1097-1105.

[9] HSU K L,GUPTA H V,SOROOSHIAN S. Artificial Neural Network Modeling of the Rainfall-Runoff Process[J]. Water Resources Research,1995,31(31):2517-2530.

[10] RUMELHART D E,HINTON G E,WILLIAMS R J. Learning representations by back-propagating errors[M]// Neurocomputing: foundations of research. MIT Press,1988:533-536.

[11] BOUVRIE J. Notes on Convolutional Neural Networks[J]. Neural Nets,2006.

[12] 钟联波. GPU 与 CPU 的比较分析[J]. 技术与市场,2009,16(9):13-14.

[13] RAMPASEK L,GOLDENBERG A. TensorFlow: Biology's Gateway to Deep Learning? [J]. Cell Systems,2016,2(1):12.

[14] ROSSUM G V,DRAKE F L. Python 3 Reference Manual[J]. Department of Computer Science,1995,111(254):1-52.

[15] 刘琦. 卷积检测模型的 GPU 加速研究[D]. 上海:上海交通大学,2014.

[16] 张任其,李建华,范磊. 分布式环境下卷积神经网络并行策略研究[J]. 计算机工程与应用,2017,53(8):1-7.

[17] LIAO F,LIANG M,LI Z,et al. Evaluate the Malignancy of Pulmonary Nodules Using the 3D Deep Leaky Noisy-or Network [J]. arXiv:1711.08324,2017.

[18] ABADI M,AGARWAL A,BARHAM P,et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems[J]. arXiv:1603.04467,2016.