

DIGITS DevBox 深度学习服务器

Dai Jialun

August 29, 2015

1. 硬件配置

实验室所组装的 DIGITS DevBox 深度学习服务器，其具体硬件配置如下。这台服务器的原型为 NVIDIA 官方生产的 [DIGITS DevBox](#)，可参考其配置。

显卡 4 个 ASUS（华硕）GTX 980Ti-6GD5

芯片厂商: NVIDIA
显卡芯片: GeForce GTX 980Ti
显示芯片系列: NVIDIA GTX 900 系列
核心代号: GM200
显存类型: GDDR5
显存容量: 6144MB
显存位宽: 384bit
最大分辨率: 4096×2160
接口类型: PCI Express 3.0 16X
I/O 接口: HDMI 接口/DVI 接口/3 个 DisplayPort 接口
电源接口: 8pin+6pin
产品尺寸: 266.7×111.2×38.1mm
参考报价: 5999*4=23996

CPU 1 个 Intel（英特尔）Core i7-5960X

CPU 主频: 3GHz
最高睿频: 3.5GHz
总线类型: QPI 总线
总线频率: 8GT/s
插槽: LGA 2011-v3

CPU 架构: Haswell

核心: 八核心十六线程

制作工艺: 22 纳米

功耗: 140W

三级缓存: 20MB

最大支持内存: 64G

指令集: SSE4.2, AVX 2.0, AES

内存控制器: 四通道: DDR4 1333/1600/2133

参考报价: 7699 RMB

主板 1 个 ASUS (华硕) X99-E WS

主芯片组: Intel X99

CPU 插槽: LGA 2011-3

支持 CPU 数量: 1 颗

内存类型: DDR4

内存插槽: 8*DDR4 DIMM

最大内存容量: 128GB

内存描述: 支持四通道 DDR4 3000(超频)/3200(O.C.)/2800(超频)/2666(超频)/2400(超频)/2133MHz 内存

显卡插槽: PCI-E 3.0 标准

PCI-E 插槽: 7×PCI-E X16 显卡插槽

SATA 接口: 8×SATA III 接口; 1×SATA Express 接口; 1×M.2 接口 (10Gb/s)

USB 接口: 12×USB3.0 接口 (10 背板 +2 内置); 4×USB2.0 接口 (2 背板 +2 内置)

版型: E-ATX 板型

外形尺寸: 30.5×26.7cm

多显卡技术: 支持 NVIDIA 4-Way SLI 四路交火技术

RAID 功能: 支持 RAID 0, 1, 5, 10

尺寸: 30.5 厘米 x 26.7 厘米

参考报价: 4799 RMB

内存 2 个 CORSAIR (海盗船) VENGERNCE (复仇者)LPX 32GB (4 × 8GB) DDR4 2400MHz
CMK32GX4M4A2400C14R

内存容量：套装（4×8GB）

内存类型：DDR4

内存主频：2400MHz

参考报价：5000*2=10000 RMB

硬盘 3 个 WesternDigital（西部数码）4TB 7200 转

硬盘容量：4000G

缓存：64M

转速：7200rpm

接口类型：SATA3.0

接口速率：6Gb/s

参考报价：1799*3=5397 RMB

固态硬盘 1 个 Samsung（三星）SSD 850pro 512GB

接口类型：SATA3

硬盘尺寸：2.5 英寸

参考报价：2999 RMB

固态硬盘 1 个 Samsung SSD 512GB SM951 cache for RAID

参考报价：3450 RMB

机箱 1 个 CORSAIR（海盗船）900D

机箱样式：台式机箱（全塔）

适用主板：**EATX 板型**，ATX 板型，MATX 板型

电源类型：标准 ATX PS2 电源（选配）

电源设计：下置电源

显卡限长：400mm

5.25 英寸仓位：4 个

3.5 英寸仓位：9 个

2.5 英寸仓位：9 个

扩展插槽：10

前置接口：4*USB 2.0；2*USB 3.0

散热性能：前：3×120mm 风扇（标配），顶：4×120mm 或 3*140mm 风扇（选配），后：1×140mm 风扇（标配），底：8×120mm 或 6*140mm 风扇（选配）

尺寸：649.6×252×691.6mm

参考报价：2499 RMB

电源 1 个 CORSAIR（海盗船）AX1500i 1500W

功率：1500W

风扇描述：14cm 风扇

电源尺寸：150x86x225mm

参考报价：3599 RMB

散热器 1 个 CORSAIR（海盗船）H110 水冷 CPU 散热器

参考报价：999 RMB

风扇 6 个 CORSAIR（海盗船）AF120 静音版双包装

参考报价：89*12=1068 RMB

光驱 1 个 AUSU（华硕）DRW-24D1ST

参考报价：120 RMB

配件 1 个 Thermaltake Commander FT 触控式面板风扇控制器，Deepcool FAN HUB（九州风神风扇集线器）

参考报价：299 RMB

显示器

键盘鼠标

2. 名词解释

DVI Digital Visual Interface, 数字视频接口, 如 Figure 1。



Figure 1: DVI 接口

DisplayPort 高清数字显示接口标准, 如 Figure 2。



Figure 2: DisplayPort 接口

PCI-E PCI Express, PCI 型总线接口, 如 Figure 3。

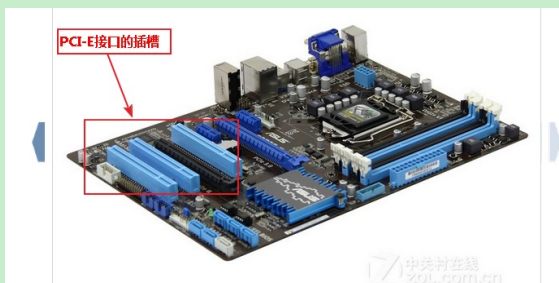


Figure 3: PCI-E 接口

SATA Revision 3.0 Serial Advanced Technology Attachment, 6Gbps, 如 Figure 4。

SATA EXpress SATA 3.0 下一代的 SATA 接口, 10Gbps, 如 Figure 5。

M.2 一种替代 MSATA 新的接口规范, 优势体现在速度和体积。支持 Socket2 和 Socket3 两种接口类型, 如 Figure 6。



Figure 4: SATA 3.0 接口



Figure 5: SATA-E 接口

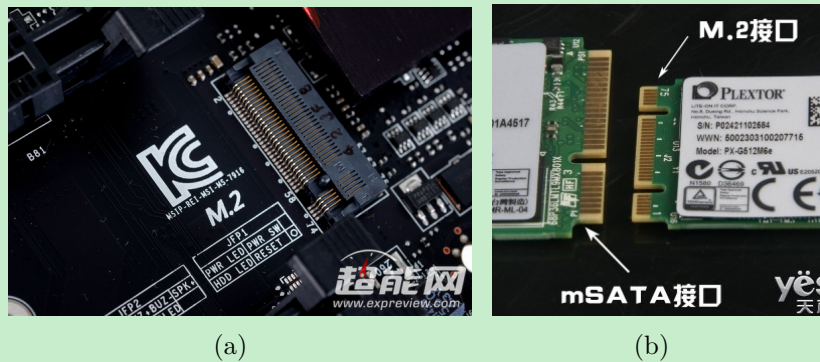


Figure 6: M.2 接口

RAID Redundant Arrays of Independent Disks, 磁盘阵列。磁盘阵列是由很多价格较便宜的磁盘，组合成一个容量巨大的磁盘组，利用个别磁盘提供数据所产生加成效果提升整个磁盘系统效能。利用这项技术，将数据切割成许多区段，分别存放在各个硬盘上。

RAID5 一种存储性能、数据安全和存储成本兼顾的存储解决方案。为系统提供数据安全保障，但保障程度要比 Mirror 低，而磁盘空间利用率要比 Mirror 高。数据以块为单位分布到各个硬盘上。RAID 5 不对数据进行备份，而是把数据和与其相对应的奇偶校验信息存储到组成 RAID5 的各个磁盘上，并且奇偶校验信息和相对应的数据分别存储于不同的磁盘上。当 RAID5 的一个磁盘数据损坏后，利用剩下的数据和相应的奇偶校验信息去恢复被损坏的数据。

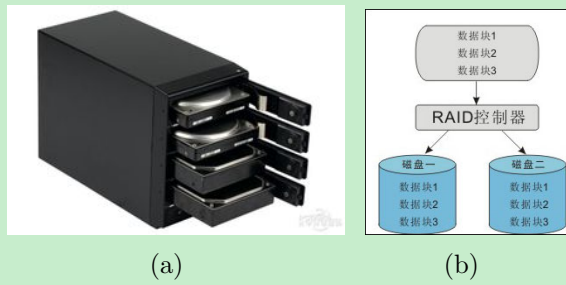


Figure 7: RAID

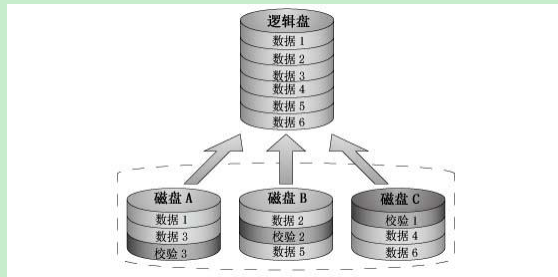


Figure 8: RAID5

SLI Scalable Link Interface, 可灵活伸缩的连接接口（支持多显卡技术）¹。这是一种可把两张或以上的显卡连在一起，作单一输出使用的技术，从而达至绘图处理效能加强的效果，如 Figure 9。

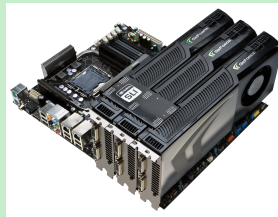


Figure 9: SLI 技术

DDR4 Dual Data Rate SDRAM, 是一种高速 CMOS 动态随即访问的内存。DDR4 支持 2133MHz, 32GB DDR4-2133 达到 48.4GB/s。

GDDR5 Graphics Double Data Rate SDRAM version5, 是一种高性能显卡用内存, 需搭配支持 PCI-E 以上规格的显卡, 高频率达 4GHZ, 低功耗。

UEFI Unified Extensible Firmware Interface, 统一的可扩展固件接口, 是一种详细描述类型接口的标准。这种接口用于操作系统自动从预启动的操作环境, 加载到一种操作系统上。

¹<http://www.geforce.cn/whats-new/guides/introduction-to-sli-technology-guide#1>

BIOS Basic Input/Output System, 基本输入/输出系统。计算机启动时, 最先加载的 ROM 程序, 不可擦写。

固件 Firmware, 固定软件, 写入 EROM 或 EEPROM 中的程序。固件担任着一个系统最基础最底层工作的软件。初期, 这些硬件内所保存的程序是无法被用户直接读出或修改的, 如今这些是可以重复刷写的, 让固件得以修改和升级。

MRB 分区 MRB 分区表是将磁盘的分区信息保存到磁盘的第一个扇区 (MRB 扇区) 的 64 个字节中, 每个分区项 (文件系统、起始柱面号、磁头号等信息) 占有 16 个字节, 因此总共只能记录 4 个主分区。由于在一个分区项中用 4 个字节存储分区的总扇区数 (2^{32}), 每扇区 512 字节 (2^9B), 因此每个分区不能超过 2TB ($2^{32} \times 2^9B = 2^{41}B = 2TB$)。磁盘容量超过 2TB 以后, 分区的起始位置也就无法表示了。

GPT 分区 GPT 分区表是基于可扩展固件借口 (EFI) 使用的磁盘分区架构, 支持每个磁盘可达到 128 个分区, 且最大容量可达 18EB。

cuDNN CUDA Deep Neural Network library², 是专门针对 Deep Learning 框架设计的一套 GPU 计算加速方案, 使工作者投入与设计和训练神经网络模型, 而不用在底层的表现上花费时间。目前支持的 DL 工具包括 caffe, Theano 和 Torch 等。cuDNN 的 Release 版本有 cuDNN v1, cuDNN v2 for CUDA 6.5 and later 和 cuDNN v3 for CUDA 7.0 and later。

CUDA Compute Unified Device Architecture, 是一种由 NVIDIA 推出的通用并行计算架构, 该架构使 GPU 能够解决商业、工业以及科学方面的复杂计算问题。在架构上采用了一种全新的计算体系结构来使用 GPU 提供的硬件资源, 从而给大规模的数据计算应用提供了一种比 CPU 更加强大的计算能力, 即将显卡中的充分使用 GPU 来解决大量数据计算问题。开发人员可使用 C 语言来为 CUDA 架构编写程序, 所编写出的程序可在支持 CUDA 的处理器上以超高性能运行。CUDA3.0 已经开始支持 C++ 和 FORTRAN。

CUDA Toolkit CUDA Toolkit 是进行 CUDA 开发所需要的一个软件。它通过 GPU 加速技术, 给 C 和 C++ 开发者提供了一个综合的开发环境。CUDA Toolkit 包括 NVIDIA GPUs 编译器, 数学库以及 debug 和优化。

HDF Hierarchical Data Format, 可以存储不同类型的图像和数码数据的文件格式, 并且可以在不同类型的机器上传输, 同时还有统一处理这种文件格式的函数库。大多数普通计算机都支持这种

²<http://devblogs.nvidia.com/parallelforall/accelerate-machine-learning-cudnn-deep-neural-network-library/>

文件格式。文件格式是指计算机存储和处理数据的方式。目前常用的图像文件格式很多,如 GIF、JPG 等,其共同的缺点是结构太简单,不能存放除影像信息外其他的有用数据,像遥感影像的坐标值、参数等都无法保存,而且用不同格式存储影像数据使得读取、传输、共享变得复杂。因此,有必要建立一种标准格式以解决上述问题。

HDF5 HDF5 能处理更多的对象,存储更大的文件,支持并行 I/O,线程和具备现代操作系统与应用程序所要求的其他特性,而且数据模型变得更简单,概括性更强。HDF5 只有两种基本结构:组 (group) 和数据集 (dataset)。

DIGITS NVIDIA Deep Learning GPU Training System,通过对数据实行可视化的实时检测,让开发者快速设计出深度神经网络。DIGITS 最大的特点是,你不需要写任何的代码就可快速上手 DIGITS。DIGITS 是通过网络浏览器来进入的网页应用。

3. RAID5

3.1 RAID 的优点

- 可高效恢复磁盘
- 增强了速度
- 扩容了存储能力

3.2 实现 RAID 方法

硬 RAID Hardware RAID,通过用硬件 (RAID 卡或者磁盘阵列) 来实现 RAID 功能。硬件 RAID 具备了自身的 RAID 控制/处理与 I/O 处理芯片,甚至还有阵列缓冲 (Array Buffer),对 CPU 的占用率以及整体性能都是最优势的,但设备成本也是三种方法中最高的。Hardware RAID 自成一个单元,由自身硬件和软件管理 RAID,与主板和操作系统无关,即 Ubuntu 系统不需要额外的程序来管理。

软 RAID Software RAID,通过用操作系统的软件程序 (Linux 系统下的 mdadm 命令) 来完成 RAID 功能。软件 RAID 的所有功能都是操作系统与 CPU 来完成,没有第三方的控制/处理与 I/O 芯片,与主板 BIOS 程序无关,其效率与稳定性较低。例如在 Ubuntu 系统下的软 RAID,其格式化、挂载、写入与重建全部由 mdadm 负责。

伪 RAID Fake RAID，又称 BIOS RAID。通过主板的集成芯片，内建 RAID 控制器来创建阵列，由操作系统驱动识别（主要表现在 Intel Desktop 的主板上表现的比较明显）。由于缺乏独立的 I/O 处理芯片，所以这方面的工作仍要由 CPU 与驱动程序来完成。另外，Fake RAID 所采用的 RAID 控制/处理芯片的能力一般都比较弱，不能支持高的 RAID 等级。在 Intel 集成芯片的主板，主要使用 Intel Rapid Storage Technology 来管理，该技术主要支持 Window 系统，不支持 Linux 系统。在 Linux 系统下，Intel 主要使用 dmraid 和 mdadm 来管理 RAID，**推荐使用 mdadm**。

3.3 主板集成 RAID 与外插 RAID 卡区别

性能 主板集成的 RAID，它的性能以及速度是通过主板的 CPU 与内存来实现的，它会占有主板一定的带宽，会影响整机的性能；外插 RAID 卡，有自己的 CPU 和内存，所以数据处理大部分都会独立处理，不会影响主板上的 CPU 与内存速度。总体看来，外插的 RAID 卡的 RAID 在功能和速度上，都优于主板集成。

安全性 主板集成的 RAID，其安全性不能够得到保证，因为是通过更改主板的 BIOS 选项做成的，所以一旦主板损坏、主板的 CMOS 电池掉电或无意更改了主板 BIOS 的设置都会带来 RAID 的丢失。通过主板做成的 RAID，一旦丢失，将会不能恢复，后果是非常严重的；而外插的 RAID 卡所做成的 RAID，不会因为主板损坏、主板的 CMOS 电池掉电等现象对数据造成影响，所以外插的 RAID 卡，其安全性远远大于主板集成的。

3.4 实现 RAID 方法比较

在这台 DIGITS DevBox 服务器的 RAID 主要是 Fake RAID 和 Software RAID，分别对应的软件是 dmraid 和 mdadm。Intel 同时支持 dmraid 和 mdadm，**但是更推荐使用 mdadm**。

3.4.1 dmraid

- dmraid 主要是属于 Fake RAID 来创建、管理 RAIDd 的。在启动时候，由主板上的芯片驱动 RAID，当载入 Linux 内核之后，由 Linux 接手管理，消耗 cpu 和内存等资源³。在 Ubuntu 系统中，dmraid 主要是将硬件的 RAID 映射成系统中/dev/mapper/目录下的设备，例如/dev/mapper/isw_dfadcd_Volume1，其中 isw 为 intel 的硬件名字，Volume1 为 RAID 名称⁴。

³<http://www.cnblogs.com/linuxer/archive/2012/03/07/2441224.html>

⁴<http://book.51cto.com/art/200902/110754.htm>

- 在 BIOS 创建的 RAID，在 Ubuntu 系统中，可能会出现大容量硬盘识别不正确的问题。例如，在 BIOS 中创建的 3 个 3.6TB 的硬盘组成的 RAID5，理论上应该为 7.2TB，但是 Ubuntu 系统只能识别为 3.6TB，容量偏小，而 Ubuntu Server 和 Debian 甚至都无法识别，不显示。
- **不推荐使用 dmraid 命令。**首先，dmraid 从 2011 年已经不提供更新了，而 mdadm 仍然不测试和更新；其实，dmraid 对于大容量硬盘的识别容易出错，如今的硬盘都是 1TB 以上的，对于 dmraid 很容易造成错误；最后，dmraid 是将 RAID 映射成 mapper，无法真正实现 RAID 数据恢复等高级功能。

3.4.2 mdadm

- 在 linux 系统中目前以 MD(Multiple Devices) 虚拟块设备的方式实现软件 RAID，利用多个底层的块设备虚拟出一个新的虚拟设备，即使用 mdadm 命令⁵。
- Fake RAID 只提供廉价的控制器，RAID 处理开销仍由 CPU 和内存负责，因此性能与效率基本与 Software RAID 基本一致。对于 Linux 系统，使用 Software RAID 一般比 Fake RAID 更稳定和安全⁶。
- Ubuntu 的软 RAID 相关命令为 mdadm，其配置、测试、删除参考⁷。
- **在没有 Hardware RAID 的条件下，推荐使用 mdadm 实现 RAID。**

3.5 创建 RAID5 步骤 (Ubuntu 下 Software RAID, 推荐!!!)

在 Ubuntu 系统中，通常使用 mdadm，即 Software RAID 方法来创建 RAID5⁸

1. 安装 mdadm，查看实际磁盘情况。

```
1 sudo apt-get install mdadm    # 安装 mdadm
2 sudo fdisk -l                 # 查看磁盘空间以及分区
```

2. 初始化。对各个磁盘删除分区 (fdisk 命令)，且进行格式化 (mkfs 命令)。小容量硬盘 (不到 2TB) 使用 MRB 分区表，大容量硬盘 (2TB 以上) 使用 GPT 分区⁹。如果要对大容量硬盘分

⁵<http://blog.csdn.net/yuesichiu/article/details/8502680>

⁶http://blog.163.com/jiangh_1982/blog/static/12195052014252131760/

⁷<http://blog.itpub.net/27771627/viewspace-1246416/>

⁸<http://blog.itpub.net/27771627/viewspace-1246416/>

⁹<http://wangheng.org/shi-yong-parted-chuang-jian-gpt-fen-qu.html>

区，使用 parted 命令。

```
1 sudo fdisk /dev/sdX          #用fdisk对某块硬盘处理，/dev/sdX中X表示磁盘号，例如
    /dev/sdb
2 sudo mkfs.ext4 /dev/sdX      #用mkfs将/dev/sdX格式化为ext4格式
```

3. **创建 RAID5**。将 3 块磁盘组成 RAID5，-l5 表示 level5,-n3 表示磁盘数目为 3。

```
1 sudo mdadm -C /dev/md0 -l5 -n3 /dev/sdb1 /dev/sdb /dev/sdc /dev/sdd    #sdb, sdc
    和sdd为组成RAID5的磁盘，md0为创建好的RAID盘
```

4. **格式化**。将 RAID 格式化为 Linux 支持的 Ext4 文件系统。

```
1 cat /proc/mdstat            #查看RAID恢复进度
2 sudo mdadm -D /dev/md0      #查看RAID详细情况
3 sudo mkfs.ext4 /dev/md0      #对RAID格式化为Ext4
```

5. **挂载**。将 RAID 挂载到/deep 目录下。

```
1 sudo mkdir /deep            #在/目录下创建/deep
2 sudo mount /dev/md0 /deep    #将md0挂载到/deep下
```

6. **自动挂载**。将/dev/md0 /deep ext4 defaults 1 2，写入/etc/fstab。建议重启多次后再查看 RAID 的磁盘号，可能我们创建的盘号为 md0，但是重启后显示为 md127。如果将之前的/dev/md0 直接写入/etc/fstab，如果出错，可能导致重启出现问题。

```
1 sudo vim /etc/fstab
```

4. Caffe

NVIDIA 的 DIGITS DevBox[默认软件配置](#)，可作为参考。

4.1 Caffe 配置

在本台 DIGITS DevBox 服务器中，Caffe 所安装的依赖或者库文件：

- Ubuntu 14.04.3（默认情况为 Ubuntu 14.04.3，暂时不推荐使用 15.04）
- 显卡驱动 346.82（最新版为 355.06）
- CUDA7.0（目前有 CUDA6.5、CUDA7.0 和 CUDA7.5 三个版本，CUDA7.5 还不稳定，暂不使用）
- cuDNN v2（cuDNN v3 已更新）
- OpenCV3.0.0rc1（推荐用欧新宇老师提供的下载包，之前手动下载的库在安装过程可能出现问题）
- Python 2.7.6（自动安装）
- Matlab2014b
- MKL(作为 BLAS, 使用 parallel_studio_xe_2015_update2.tar.gz, 目前 2016 版已更新。ATLAS、OpenBLAS 试过，没成功，可以再作尝试)
- glog(Google Logging Library, glog-0.3.3)

4.2 Caffe 安装

目前，Caffe 主要的安装步骤是参考了欧新宇老师博客中的[Caffe + Ubuntu 15.04 + CUDA 7.0 新手安装配置指南](#)。安装 Caffe 时请认真阅读。

4.3 问题解决

- 在安装过程中，必须根据 CUDA 版本选择 NVIDIA 驱动版本，例如 CUDA7.0 对应的版本应该是 NVIDIA346 版。如果按照 Caffe 安装步骤给出的方法，基本上不会出现问题。但是如果是手动安装 NVIDIA 驱动，且安装了最新版本驱动，例如安装了最新的 355 版本后，可能显示 **no CUDA-capable device is detected**。则卸载最新版本保证回到对应版本驱动。

```
1 sudo ./NVIDIA-Linux-x86_64-355.06.run --uninstall
```

- 目前(2015.08.26)的从 caffe 的 Github 上下载的 caffe-master 是有问题的。在编译 make 阶段, 运行没问题, 但是在跑 Cifar10 数据集时, 会报错 `H5FOpencv()` 等。因为部分文件格式改为了 *.h5, 这是 HDF5 格式。没有依赖文件的话, Ubuntu 暂时不支持。如果碰到这个问题, 将 *.h5 格式全去掉.h5, 改为二进制格式则正常运行。例如在 `caffer-master/examples/cifar10/train_quick.sh`, `caffer-master/examples/cifar10/train_quick.sh` 与 `caffer-master/examples/imagenet/train_caffenet.sh`。目前该问题只能通过这个方法解决, 但是在将来可以查看、安装 [HDF5](#) 来尝试解决问题。
- 在 Github 的 Caffe 库里的 release 版本存在很多错误, 例如 `caffe-rc2`, 在编译或者安装上会出现问题, 总之尽量使用最新的或者之前备份的 Caffe 版本。

5. DIGITS

[DIGITS](#) 是一款检测深度学习的软件工具。目前使用的版本为 DIGITS v2.0.0-rc for Ubuntu 14.04

5.1 软件需求

DIGITS 具体软件参考 [Github 官网](#)。

- NVIDIA 驱动版本必须是 346 或者更高版本
- CUDA toolkit (≥ 6.5)
- cuDNN
- Caffe 或 Torch
- graphviz

```
1 sudo apt-get install graphviz
```

5.2 安装 DIGITS

安装 DIGITS 可参考 [Github 官网](#)。

1. 下载安装包 [digits-2.0.0-rc.tar.gz](#)
2. 解压安装包

```
1 tar xvf digits-2.0.0-rc.tar.gz
```

3. 进入解压目录 digits-2.0, 运行安装脚本

```
1 cd digits-2.0
2 ./install.sh
```

4. 运行 runme.sh, 启动 DIGITS server。在浏览器打开 <http://localhost:5000/> 来查看 DIGITS 工具。

```
1 ./runme.sh
```

5.3 DIGITS 使用

DIGITS 主要是通过浏览器的网页应用。具体操作可参考[Github 官网](#)。

1. Datasets
2. Model

6. 其他工作

6.1 SLI

SLI 是一种多 GPU(图形处理器) 扩展引擎, 从本质上来讲这是一种利用一块以上的显卡来提升游戏性能的方法, 每额外增加一颗 GPU, 性能最多增加 100%。通过让多颗图形处理器能够独立地渲染单一 3D 环境, 理论上我们能够让帧速率达到单颗 GPU 时的帧速率与 GPU 数量相乘的水平。因此, 其主要作用是提升游戏场景的渲染和加速, 在 DeepLearning 中, 不需要设置 SLI。

6.2 安装 NVIDIA 显卡驱动

此方法只适合安装 NVIDIA 最新的显卡驱动, 较为麻烦, 不推荐使用, 只供参考。在 Caffe 安装过程中, 提供了更简便的方法来安装 NVIDIA 显卡驱动, 请使用其方法。

1. 以下是驱动来源, 更新受限制驱动列表 (源)。

- 开源驱动 nouveau (livecd 安装时用的驱动)
- 源 (受限制驱动列表)
- PPA 源 (一般是私人建的, 方便群众用)
- 自己下载编译的驱动 (我们使用的方法)

```
1 sudo apt-get install nvidia-current nvidia-settings
```

2. 下载驱动

- (a) [Nvidia 中文官网](#) 下载驱动。
- (b) 将下载的目前最新的驱动文件 NVIDIA-Linux-x86_64-355.06.run, 放到 [/home/用户名/](#) 目录下面。
- (c) 编译依赖文件,

```
1 sudo apt-get install build-essential pkg-config xserver-xorg-dev linux-headers-`uname -r`
```

3. 屏蔽开源驱动 nouveau, 选一种方法即可

- blacklist (推荐)
 - (a) 打开终端, 输入 `sudo vim /etc/modprobe.d/blacklist.conf`
 - (b) 添加 blacklist nouveau
- grub2
 - (a) 打开终端, 输入 `sudo vim /etc/modprobe.d/blacklist.conf`
 - (b) 修改 `GRUB_CMDLINE_LINUX=""` 为 `GRUB_CMDLINE_LINUX="nomodeset"`
 - (c) 输入 `sudo update-grub`

4. 安装装备

- (a) 清除之前与 nvidia 相关的驱动程序, `sudo apt-get -purge remove nvidia-*`

- (b) <ctl+alt+F1> 切换到虚拟终端 tty1 (如果不屏蔽 nouveau, 可能会出现黑屏现象); 黑屏则 <ctl+alt+F7> 切换回图形界面, 然后重启后, 按下 Esc 或者选择 low-quality, 进入 tty1, 进行驱动的安装
- (c) 关闭图形环境 `sudo stop lightdm` (Ubuntu15.04 下, 运行 `sudo systemctl stop lightdm`)

5. 安装过程

- (a) 在驱动文件目录下, `sudo ./NVIDIA-Linux-x86_64-355.06.run`
- (b) 安装中基本选择 <Yes>
- (c) 重启后, 即完成 NVIDIA 显卡驱动安装

6.3 创建 RAID5 步骤 (Fake RAID, 在 BIOS 界面)

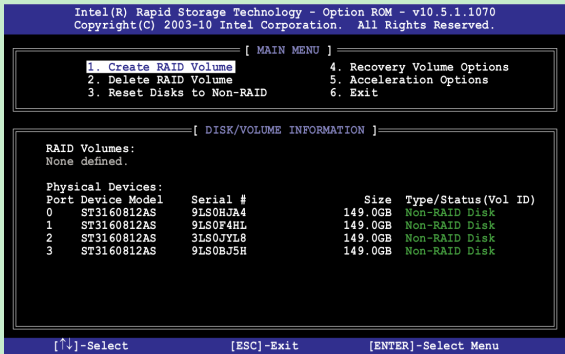
此方法是使用 BIOS 界面的 Intel Rapid Storage Technology 程序来完成 RAID 的创建, 使用的 dmraid 方法, 在 Ubuntu 系统下映射在 /dev/mapper/ 下, 不推荐使用, 只供参考。在 Ubuntu 系统下, 请使用 mdadm 命令完成 RAID 相关任务。

1. 确认主板芯片组是否支持 RAID 功能。
2. 初始化。对各个磁盘删除分区 (fdisk 命令), 且进行格式化 (mkfs 命令)。小容量硬盘 (不到 2TB) 使用 MRB 分区表, 大容量硬盘 (2TB 以上) 使用 GPT 分区¹⁰。

```
1 sudo fdisk -l           #查看磁盘空间以及分区
2 sudo fdisk /dev/sdX     #用fdisk对某块硬盘处理, /dev/sdX中X表示磁盘号, 例如/dev/sdb
3 sudo mkfs.ext4 /dev/sdX #用mkfs将/dev/sdX格式化为ext4格式
4 sudo parted /dev/sdX    #用parted工具对大容量硬盘分区, 为GPT分区
```

3. 在 BIOS 程序中设置 RAID。在 Advanced Mode 下, 在状态栏中点击 Advanced, 选择 PCH Storage Configuration, 将 SATA Controller 1 Mode Seletion 设置为 RAID。(只有 Controller 1 支持 RAID 模式)
4. 进入 Intel Rapid Storage Technology (Intel RST)。如果系统运行开机自检 (POST) 时, 按下 <Ctrl + I> 进入程序界面进行管理。否则, 在 BIOS 界面中, 选择 Intel Rapid Storage Technology 为 On 后, 重启再进入 BIOS, 在 Advanced Mode 下, 在状态栏中点击 Advanced, 在底部可看到 Intel Rapid Storage Technology 的选项, 点击进入设置。

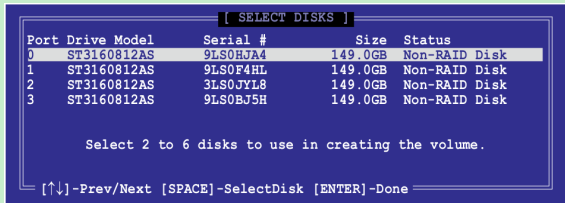
¹⁰<http://wangheng.org/shi-yong-parted-chuang-jian-gpt-fen-qu.html>



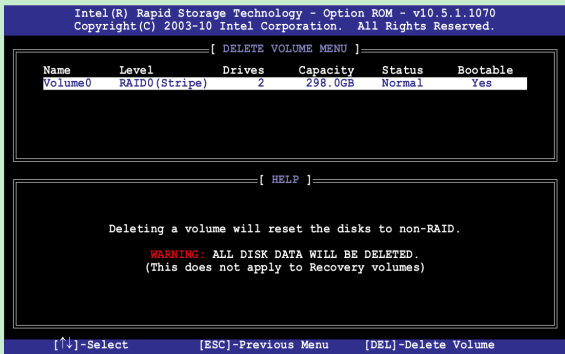
5. 创建 RAID，选择 Create RAID Volume。



6. 设置 RAID，选中上图的选项中的 Disks，显示下图。



7. 创建成功的 RAID，如图。RAID 的 status 比较重要，应为 Normal，如果出现 Rebuild、Degrade、Failed 等，请重新创建。



8. 格式化, 进入 Ubuntu 系统后, 在 `/dev/mapper/` 下可看到 RAID5 被映射成为 `isw_dfafd_Volume1`。将其格式化为 Ext4 文件系统。(对于大容量的硬盘的识别会出现问题, 会显得比理论容量小; 在 Window 系统下, 不会出现这中情况。)

```
1 sudo mkfs.ext4 /dev/mapper/isw_dfafd_Volume1
```

9. 挂载, 将格式化好的映射硬盘, 挂载到 `/deep` 目录下。

```
1 sudo mkdir /deep
2 sudo /dev/mapper/isw_dfafd_Volume1 /deep
```

10. 自动挂载, 为了重启后, 直接使用映射硬盘, 让其自动挂载。按照格式进入 `/etc/fstab`

```
1 sudo vim /etc/fstab
```

6.4 RAID5 实验情况

DIGITS DevBox 的 RAID5 在各种软件配置和系统下的测试。目前 X99 主板集成的 RAID 功能, 为 Intel Rapid Storage Technology, 在 Linux 下主要使用的是 DM RAID 和 MD RAID, 即 `dmraid` 和 `mdadm` 命令。DM RAID (`dmraid`) 默认情况下, Ubuntu 可以识别, 但是已经几年没更新了; 而 `mdadm` 经过几年的测试, 在工业界更受欢迎。`mdadm` 在 Window 下有 UI 界面, 在 Linux 下只有命令行, 其产生的中间数据支持两个系统下, 可用在双系统环境下。在单 Linux 系统下, 使用 `mdadm` 比较合适。

本此实验是使用 3 块 3.6TB 的硬盘, 在 BIOS 中和 Ubuntu 系统下分别创建 RAID5 进行实验, 理论上 RAID5 应该为 7.2TB。

- 默认情况下, BIOS 中已创建 RAID5
 - Ubuntu 识别 `/dev/mapper/isw_dafadfadsf_Volume1`, 只有 3.6TB
 - Ubuntu server 无法用 `dmraid` 激活 `mapper`, 无法显示
 - Debian 不识别
- 安装 `mdadm` 的情况下, BIOS 中已创建 RAID5, 但是 ubuntu 系统下 `mdadm` 不创建 RAID5

- Ubuntu 识别 /dev/mapper/isw_dafadfadsf_Volume1, 只有 3.6TB
 - Ubuntu server 无法用 dmraid 激活 mapper, 所以无法显示
 - Debian 不识别
- 安装 mdadm 和 dmraid 的情况下, BIOS 中不创建 RAID5, Ubuntu 系统下 mdadm 创建 RAID5
 - Ubuntu 不识别 RAID5
 - Ubuntu server 识别 RAID5 为 7.2TB
 - Debian 识别 RAID5 为 7.2TB

由以上实验, 可以看出 Ubuntu 只识别 dmraid 所创建的 RAID5 的映像, 而 Ubuntu server 和 Debian 都不能识别; 对于 mdadm 所创建的 RAID5, Ubuntu server 和 Debian 在安装系统时, 都能够识别, 而 Ubuntu 不能够识别。

因此, 考虑到目前的系统为 Ubuntu14.04, 而且需要使用硬盘的 RAID5 功能, 且 mdadm 比 dmraid 更稳定与安全, 所以在此决定使用 mdadm 创建软 RAID。