CUDA 校园编程竞赛 2012 年指定题目

GPU 加速 CAPTCHA 字符识别

本次 CUDA 校园编程竞赛的指定题目之一是基于 GPU 加速的 CAPTCHA 字符自动识别中的学习和识别过程。解决算法完全开放，参赛选手可以任意选择。

一、问题背景

伟大的数学家和计算机科学的奠基人之一阿兰.图灵在 1950 年的一篇文章提出了“图灵测验”(Turing test)的思想，基本方法是把提问者(人)和一台计算机分隔开，人和计算机之间互相不能看到，但是可以通过某种方式交换信息。提问者提出一系列问题并获得答案，如果提问者不能由答案分辨与之交流的是计算机还是人类，则认为计算机已经具备智能。60 年来，虽然人工智能科学获得了长足的发展，但是目前的计算机尚不具备通过图灵测验的智能。也就是说，还没有一台计算机能够通过图灵测验。事实上，我们可以找到许多对于人工智能非常困难的问题，却可以被人轻松解决。

尽管以上事实对人工智能而言不免有些令人沮丧，但是这一事实却被证明具有巨大的应用潜力！这就是 CAPTCHA(Completely Automated Public Turing test to tell Computers and Humans Apart，完全自动的能够区分计算机和人类的公共图灵测试)技术，由图灵奖获得者、卡内基.梅隆大学的 Manuel Blum 教授和他的学生在 2004 年提出。CAPTCHA 实际上是图灵测验的逆问题，即要求受试者回答一个很难被计算机解决、但对人来说却轻而易举的问题，由此可以确定受试者是计算机还是人类。

那么能够区分计算机和人究竟有什么作用呢？我们今天生活在网络时代，通过国际互连网络与外界进行交流。由于这种交流方式不再是面对面的，因此很多情况下可以使用计算机程序模拟人类而获得利益。一个典型的例子是 1999 年 slashdot.org 进行的一次评选最佳计算机学院的网上投票，人们可以在网页上选择加州大学伯克利分校、麻省理工学院、卡内基.梅隆大学、华盛顿大学、康奈尔大学和斯坦福大学中的一所。麻省理工学院和卡内基.梅隆大学的学生们设

*为节省篇幅，本文引用论文没有严格遵循规范，请读者使用作者名字和文章标题查找原文。

计了自动投票程序，为各自的学校大量投票，结果分别获得了两万多张选票，而其它学校获得的票数均小于一千张。当然这还是开玩笑，更恶劣的例子是 2000 年 9 月，Yahoo 网站的聊天室被自动程序侵入，把聊天客户引向广告网站。因此，Yahoo 求助于卡内基.梅隆大学的 Manuel Blum 教授，从而发展出 CAPTCHA 思想。目前各大主要网站都使用了各种 CAPTCHA 实现方式，对人和计算机进行区分。图 1 是 reCAPTCHA 公司的 CAPTCHA 实现，相信大家都不陌生。现在由于社交网络非常活跃，很多商业组织利用自动程序登录社交网站，获取他人信息谋取利益，因此 CAPTCHA 的作用更加凸显。

图 1. reCAPTCHA 所使用的 CAPTCHA 字符

图 1 是 CAPTCHA 最常见的形式(此外还有例如区分猫狗照片、音频、视频等形式)，即网站在用户登录时显示一张图片，图片中有若干经过扭曲变形并加入噪声信息的字符，用户需要识别并输入字符。如果输入正确，则认为是人类用户，否则必须再次输入。值得注意的是，CAPTCHA 的设计必须非常小心，否则计算机程序可以通过图像处理和改进人工智能的方法学会识别 CAPTCHA。例如图 2 的字符完全可以通过图像处理而自动识别！而图 3 则对计算机和人都比较困难。近来 Google、Yahoo 等各大网站都被指出起 CAPTCHA 设计存在漏洞。英国 Newcastle 大学的 Ahmad S El Ahmad、Jeff Yan 和 MohamadTayara 在一篇名为“The Robustness of Google CAPTCHAs”的论文中给出了 CAPTCHA 设计的基本原则：1)好的 CAPTCHA 应该避免出现重复模式，例如数字“6”下半部分的圆圈可能在各种变形之下都保持拓扑一致，很容易被统计处理程序发现；2) 好的 CAPTCHA 应该很难被分割(segmentation)成单个字符，一旦分割后，单个字符的处理就容易得多，CAPTCHA 的最新变种经常会把不同字符交织在一起，就是处于这个考虑；3) 好的 CAPTCHA 应该尽量减少不同图案的关联性，机器学习算法善于从大量样本中找出有效特征，因此最好是 CAPTCHA 每次出现的图案完全独立。关于 CAPTCHA 强度，Jeff Yan 和 Ahmad Salah El Ahmad

在“The Robustness of CAPTCHAs: A Security Engineering Perspective”一文中有比较详尽的介绍。

图 2. 人和计算机都容易识别的 CAPTCHA

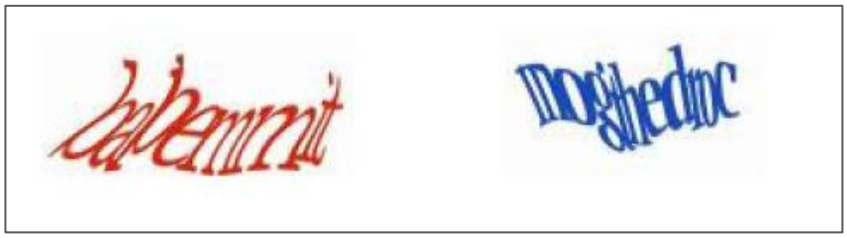

图 3. 人也很难识别的 CAPTCHA

目前自动破解 CAPTCHA 已经引起了广泛关注。俄罗斯和中国都有程序员推出自动破解软件。中国有一位“wangrui”网友的结果令人印象相当深刻，不过不愿透露客户购买其软件的使用目的。印度甚至出现了依靠大量人力进行破解的创投公司。本次竞赛选用这一题目，当然不鼓励参赛选手把破解 CAPTCHA 用于不当用途，而是希望实现这样一些目的：

1. 通过竞赛促进图像识别和人工智能技术的发展，希望大家根据 GPU 计算的特点提出更好的算法；
2. 通过竞赛更加深入地理解容易 CAPTCHA 问题，并且推动 CAPTCHA 技术的发展；
3. 在 IBM“深蓝”计算机击败卡斯帕罗夫之前，国际象棋被认为是一个智能问题，之后则被定位为对策搜索问题。本次竞赛有助于更加深入理解人工智能问题：如果 CAPTCHA 能够被有效解决，那么是由于 CAPTCHA 其实不是人工智能难题呢，还是“智能”问题实际最终可以由高度优化的算法解决？

二、问题定义

本次 CUDA 编程竞赛题目要求大家提交程序能够对 CAPTCHA 字符图像文件进行学习和自动识别。目前生成 CAPTCHA 的方法很多，本题目使用 libcapt

(http://code.google.com/p/libcapt)产生 CAPTCHA。libcapt 软件分为三个部分：catGen, captUtil, libcapt。其中 libcapt 为图像产生模块，catGen 为产生测试图像的控制模块。

在参赛作品设计阶段，选手需要完成的工作包括：

1. 修改 libcapt 软件，自主产生足够多的训练样本图像。其中训练样本中的字符数目可以是 1 个到 4 个字符。
2. 设计并实现一个基于 CUDA 的训练算法，输入上述训练样本图像集合，产生对应的结果文件，并记录训练过程所需要的时间。
3. 训练程序的输入参数应包括：学习样本的目录，其中样本的图像和正确文本信息。输出应至少应包括：训练过程所需要的时间。
4. 设计并实现一个识别算法，能读入测试样本图像文件集合，并输出识别结果。测试样本图像中的字符集合为英文字符和数字，字符长度分别为 1 到 4 个。测试样本共有 100 个图像文件，每种字符长度各有 25 幅图像，存储于同一目录下。
5. 识别程序的输入参数应包括：测试样本图像存储的目录和输出结果的文件路径。输出结果文件为一个文本文件，每行为一幅图片的识别结果，并按照字符 ASCII 码的增序排列。同时，还应给出完成 100 幅图像识别所需要的时间。

提交作品时，参赛选手应该准备以下材料：

1. 训练样本图像集合和测试样本图像集合，如果文件较大(如超过 10M 字节)，请用光盘邮寄；
2. 训练算法和识别算法的源代码，以及对应的程序说明；
3. 400 字以内的摘要，说明训练算法、识别算法、训练样本数、训练时间、识别时间和识别率等主要问题；
4. 详细设计报告和软件使用说明，格式可以参照本次大赛指定的报告格式指南。
5. 如果进入复赛的话，还请准备 10 分钟左右的 PPT。

评委会的成果评价方法如下：

1. 评委会主要考察选手所使用的训练算法和识别算法，特别重视训练算法在GPU 上的加速效果和识别算法的识别率。
2. 评委会将自主重新运行参赛选手提供的训练算法和识别算法（基于选手提供的训练样本集合和测试样本集合），以评估选手报告中数据的准确性。
3. 评委会将另外生成 100 个测试样本，并使用选手提供的识别算法进行识别，以评判程序的适用性。

温馨提示：

1. libcapt 软件中，Question（定义于 captGenerator.h）数据结构中的 ANSWER_LENGTH 可以选择产生图片中字符的数目（缺省为 4）。
2. libcapt 产生的图像为 tga 格式，可以使用 ADCSee 等软件转换为 BMP 或其他格式。
3. 建议参赛选手提供完整的程序编译、安装和使用文档，以便于评委会的测试。
4. 建议参赛选手使用 Linux 和 Windows 操作系统，CUDA SDK 版本号，编译器 版本号。请 NVIDIA 同事确定 SDK 和编译器版本。
5. 训练或预处理时间不得超过 12 小时，且一定要考察在 CUDA 上的优化和加速性能。
6. 参赛选手可以分析 libcapt 代码并且进行逆向工程来提高识别准确率，但这样的算法不具备普遍意义，因此评委有可能使用其它常见 CAPTCHA 方法产生的一组图像进行识别测试。
7. 比赛中使用的字符集至少为英文字母和数字，但不局限于此。如果能有效识别中文等复杂字体，将得到额外加分。

三、现有算法介绍

目前自动破解 CAPTCHA 的算法已经很多，这些算法可以分为两大类：图像处理和人工智能算法。

1. 基于图像处理的方法

这类方法一般需要预处理过程，即对现有样本进行分析，找出字符变换过程中的不变量，即特征。在进行破解时，程序通过寻找定位特征进行识别。

Jeff Yan 和 Ahmad Salah El Ahmad 的论文“Breaking Visual CAPTCHAs with Naïve Pattern Recognition Algorithms”是通过简单图像处理实现模式识别的典型工作。该论文针对由 Captchaservice.org 产生的 CAPTCHA 字符。在预处理阶段，对 CAPTCHA 样本进行分析，该论文找出的特征是每个字母的像素个数。破解时，首先以图片左上角像素颜色作为背景色，此后与之不同颜色的像素均作为前景字符的一部分；然后该算法对 CAPTCHA 图片进行分割，基本方法是寻找完全由背景色形成的一条直线；分割后对单个字符依次计算像素个数，即可实现识别。显然，该算法对字符之间具有重叠的 CAPTCHA 就很难奏效了。因此，该论文也引入了一种“字典”攻击方式，把一些常见字母组合存储在字典中。遇到难以分割的字符组合时，查字典确定若干可能结果，然后通过计算像素点之和获得破解结果。以上方法虽然简单，但是对 Captchaservice.org 的样本集上居然可以实现 90％以上的识别率！注意查表这种近乎“暴力破解”的办法常常是有效的，一些商业破解软件会搜集大量样本建立巨大的表格，但是本次竞赛不提倡这样的做法。

2. 基于人工智能的方法

基于各种认知网络的监督式学习(supervised learning)也是字符识别的常见方法。这种方法首先建立人工神经网及其各种变种的认知网络，然后输入已知字符集合对神经网进行训练，训练后的神经网络即可对字符进行快速识别。注意这类方法在计算时间上的瓶颈一般在训练过程，而训练结束之后的检测过程速度很快。

这一方向上，Greg Mori 和 Jitendra Malik 在 2003 年发表的论文“Recognizing Objects in Adversarial Clutter: Breaking a Visual CAPTCHA”是较早的工作。针对早期 CAPTCHA 技术中只具备有限字符集合的特点，该工作存储已知 CAPTCHA 图片，然后使用简单的机器学习技术比较待破解图片和数据库中的已知图片。该技术对简单 CAPTCHA 的成功率很高，但是已经不能对付今天的 CAPTCHA 技术。

当前的 CAPTCHA 需要学习能力更强的认知网络。University of Oklanoma 的一项工作“Machine Learning Approaches to CAPTCHA Recognition Requiring Minimal Image Processing ”(文章草稿的链接为

http://www.cs.ou.edu/~amy/courses/cs5033_fall2008/Tidwell_Shadoan.pdf) 提出了用人工神经网进行识别并辅之以图像处理技术的解决方法。卡内基-梅隆大学的 Yensy James Hall 和 Ryan E. Poplin 在名为“Using Numenta's hierarchical temporal memoryto recognize CAPTCHAs”的论文中，尝试使用一种新的认知网络“Hierarchical Temporal Memory (HTM)”来破解 CAPTCHA，并与支持向量机(Support Vector Machine, SVM)认知算法进行了比较。在中等难度的 CAPTCHA 字符集上，SVM 能够实现 9%的识别率，而 HTM 能够做到 20%。

这里我们简单介绍一下 HTM 认知算法(http://www.numenta.com/htm-overview/education/HTM_CorticalLearningAlgorithms.pdf)。该算法由 Jeff Hawkins (Palm 手机的发明人，并以此入选美国工程院院士)提出，属于人工神经网络范畴，但是没有采取仿生的思路。换言之，HTM 模仿人脑认知的模式，但没有从生物结构上直接模仿。HTM 由层次式组织的节点构成，如图 4 所示。最底层单元数量最多，类似于人类的感觉细胞。每层节点进行一定分析处理后将结果传向上层节点，因此越往上走节点数量越小，类似于各级神经中枢。以上结构对应于空间认知，同时每个节点上也需要时域信息。HTM 的一个重要贡献是利用时域相邻性确定目前认知的是否是同一对象。比如一只猫在向人走来时，猫在人视网膜中的像越来越大，简单依靠图像处理的方法很难判别这些像属于同一只猫；然而，从时间的连续性上我们很容易就知道这些像都对应同一只猫。为了完成图像识别，HTM 必须首先进行训练，这是需要向 HTM 输入连续变化的样本。接受样本过程中，HTM 以逐层方式处理，一层上每个节点都要进行时间和空间上的训练，结果稳定后向上层节点输出。上一层节点重复以上过程，直至到达顶层节点。训练过程可能非常耗时，上述 HTM 破解 CAPTCHA 工作中，训练可能需要一周时间。识别过程与训练过程类似，但是由于不需要输入大量样本，因此速度可以很快。

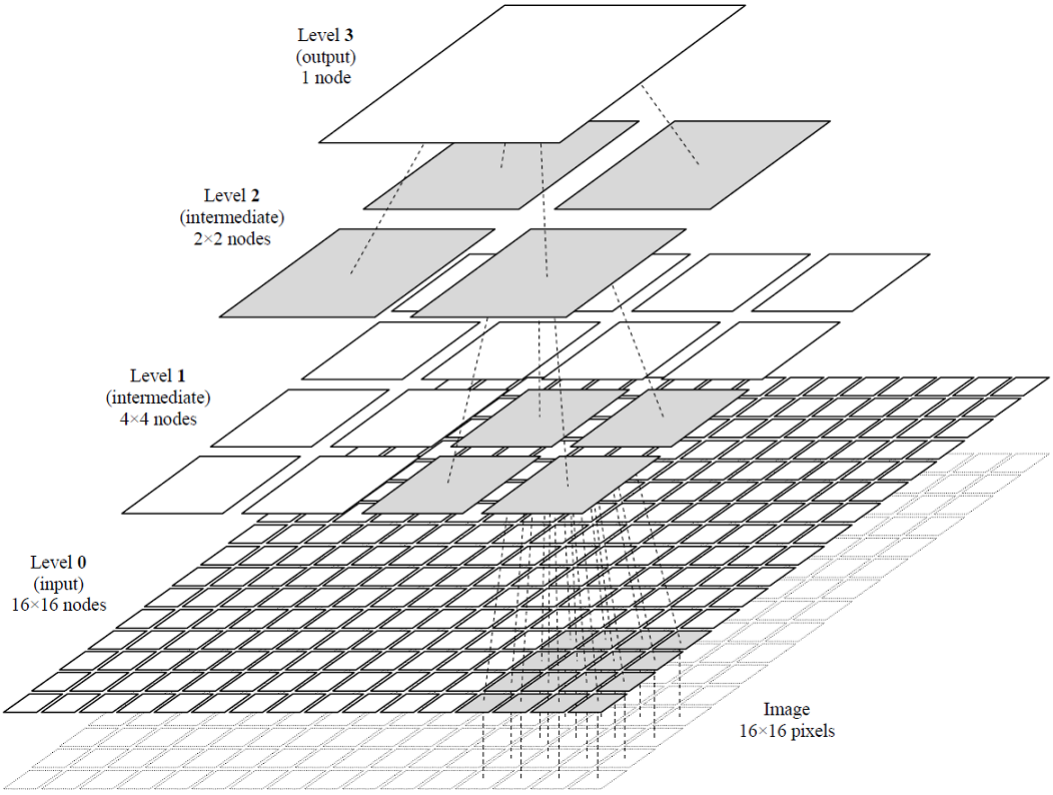

图 4. HTM 网络结构(摘自 DavideMaltoni，“Pattern Recognition by Hierarchical Temporal Memory”)

除了 HTM 外，人工神经网还有许多变种，例如 Convolutional Neural Networks、Deep Belief Networks、Restricted Boltzmann Machines 等等，都有潜力用于本问题。有兴趣的读者可以在 http://deeplearning.net/tutorial/contents.html 找到入门级的介绍。