AES 算法的 CUDA 高效实现方法*

夏春林，周德云，张 堃
（西北工业大学 电子信息学院，西安 710072）

摘 要：针对 AES 算法的 ECB 工作模式安全性低的弱点，提出了一种新的 ECB 工作模式，并在 GPU 最新统一计算设备架构（CUDA）下进行了实现。具体并行实现包括线程组织、数据存储结构以及共享内存的性能优化技术。实验结果表明这种新模式增强了 AES 算法的性能和安全性，与传统 CPU 实现相比，利用 CUDA 能够实现显著的加速性能。

关键词：高级加密标准；电子密码本模式；图形处理器
中图分类号：TP309；TP393.4 **文献标志码**：A **文章编号**：1001-3695（2013）06-1907-03
doi：10.3969/j.issn.1001-3695.2013.06.081

CUDA based high-efficiency implementation of AES algorithm

XIA Chun-lin，ZHOU De-yun，ZHANG Kun
（*School of Electronics & Information*，*Northwestern Polytechnical University*，*Xi'an* 710072，*China*）

Abstract：This paper proposed a new ECB mode for the low security of the AES algorithm in ECB mode. It was implemented under CUDA of GPU. It included the threads organization scheme，data structure and shared memory based performance improvement. Experimental results demonstrate that this new mode enhances the performance and security of AES algorithm and can obtain more efficient performance gains through CUDA than traditional CPU.

Key words：AES（advance encryption standard）；ECB（electronic code book）；GPU

高级加密标准（AES）算法是目前安全性较高的一种分组密码，其密钥最低长度为 128 bit，它可以有效地抵抗差分分析、线性分析和相关密钥分析，保证数据的安全性。对于不同的应用背景，AES 有五种可供选择的工作模式：ECB、CBC、CFB、OFB 和 CTR。其中 ECB、CTR 模式具有并行性。与通用 CPU 相比，GPU 计算密度高和低耗能的优势，不仅适用于图形图像处理，也适合于大规模数据并行的通用计算。最早尝试基于 GPU 的 AES 算法实现的是 Cook 等人[1]，主要利用图形流水线的图像子集来实现查找函数。2007 年 Manavski[2] 第一次证明了 GPU 可以用做快速加速器。Harrison 等人[3] 做了大量工作，提出了基于 GPU 设备的 AES 加密算法实现，并与之前的基于 OpenGL 的 AES 加密进行比较分析。

虽然已有算法实现了一定程度上的并行性，但是普遍存在 CUDA 的流处理器和存储器之间的通信耗时等问题。为减小数据交换的负担，提高数据的吞吐量，充分利用共享存储器，本文提出采用合理分配线程等优化技术以充分利用 GPU 的并行计算能力提高 AES 算法的实现速度。同时提出一种新的 ECB 模式以消除明文图像中存在的相同数据区域。实验测试结果表明该方法在加密计算速度和加密效果上都有了很大提升。

1 AES 算法

1.1 AES 算法描述

AES 是以轮为基础的对称加密算法，是一个密钥迭代型分组密码，它包含了轮变换对状态的重复作用，密钥长度为 128 bit、192 bit、256 bit。对应的加密轮数为 10、12、14，用 N_r 表示轮数[4]。明文经过 AddRoundKey 以及 N_r 轮组合函数变换，生成密文。其中 Round1，Round2 到 Round（N_r-1）都是由字节代替变换、行位移变换、列混合变换、密钥加法变换四种函数组成。RoundN_r 由字节代替变换、行位移变换、密钥加法变换三种函数组成。

1.2 加密变换函数解析

1）字节替换（SubBytes） 它是一种非线性的字节置换变换。将 S-盒 S_{RD} 构造为函数 g 和一个可逆仿射变换 f 的序列。首先对字节求有限域 $GF(2^8)$ 上的乘法逆，用函数表示为 $g(b)$。然后对 $g(b)$ 进行可逆仿射变换 f，用函数表示为 $f[g(b)]$。仿射变换是一个矩阵乘法和加法的混合，表示为

$$b'_i = b_i \oplus b_{(i+4) \bmod 8} \oplus b_{(i+5) \bmod 8} \oplus b_{(i+6) \bmod 8} \oplus b_{(i+7) \bmod 8} \oplus C_i \quad (1)$$

其中，$\oplus$ 表示异或，$C_i = '01100011'$。

2）行位移（ShiftRows） 它是加密算法中的线性运算。它将状态第 0 行移动 C_0 字节数，第 1 行移动 C_1 字节数，第 2 行移动 C_2 字节数，第 3 行移动 C_3 字节数。

3）列混合（MixColumns） 采用 $GF(2^8)$ 下的多项式与固定多项式 $c(x) = '03'x^3 + '01'x^2 + '01'x + '02'$ 相乘，使得到的结果取模 x^4+1。

4）密钥加法（AddRoundKey）：在这个变换中，状态的调整通过与轮密钥进行逐位异或而得到。轮密钥的长度和分组长度相等。

第 i 轮的上述四种变换可以表示为如下的数学表达式：

收稿日期：2012-09-13；**修回日期**：2012-10-24 **基金项目**：航空科学基金（2011553021）；西北工业大学基础研究基金（JC20110222）

作者简介：夏春林（1962-），男，银川人，博士研究生，主要研究方向为图像处理、模式识别（xiachunlin07@126.com）；周德云（1964-），男，浙江人，教授，博导，主要研究方向为模式识别、智能控制；张堃（1982-），男，陕西西安人，讲师，博士（后），主要研究方向为图像处理、智能控制.

$$\begin{bmatrix} d_{0,j} \\ d_{1,j} \\ d_{2,j} \\ d_{3,j} \end{bmatrix} = \begin{bmatrix} 02 \\ 01 \\ 01 \\ 03 \end{bmatrix} S_{RD}[a_{0,j+C_0}] \oplus \begin{bmatrix} 03 \\ 02 \\ 01 \\ 01 \end{bmatrix} S_{RD}[a_{1,j+C_1}] \oplus \tag{2}$$

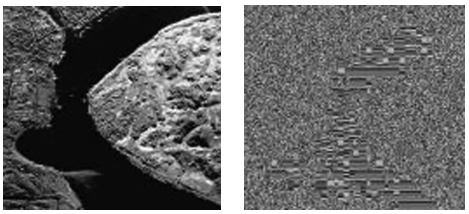
$$\begin{bmatrix} 01 \\ 03 \\ 02 \\ 01 \end{bmatrix} S_{RD}[a_{2,j+C_2}] \oplus \begin{bmatrix} 01 \\ 01 \\ 03 \\ 02 \end{bmatrix} S_{RD}[a_{3,j+C_3}] \oplus K_i$$

$$0 < i \leqslant 10 \quad 0 \leqslant j < 4$$

其中：a 是轮变换开始输入；j 是状态矩阵的列数；k 是第 i 轮的密钥；移位变换偏移量 C_0、C_1、C_2、C_3 分别为 0、1、2、3；⊕表示异或操作。

2 新 ECB 模式

在 ECB 模式加密中，相同明文块被加密为相同的密文块。如果明文块中蕴涵大量的相同连续数据区域，加密算法通过 ECB 模式不能对这些信息隐藏，则密文块中出现相同纹理区域，密文图像的熵值也达不到 8。典型的例子如图 1 所示。



(a)明文图像 (b)密文图像

图 1 ECB 模式加密效果

通过在 ECB 模式加密之前，采用非连续算术序列 $S_j = (IV + \Sigma j) \bmod 2^4$ 与图像明文数据 P_j 进行异或操作，消除明文图像中存在的相同数据区域，利用随机生成的初始化向量(IV)初始化这些序列。解密过程与此类似，先进行解密，然后进行异或操作得到明文输出结果。新 ECB 加密模式如图 2 所示。

图2 新ECB加密模式工作框图

改进后 ECB 工作模式中加密及解密两种变换如下：

加密变换：$C_j = \mathrm{CIPH}_K(P_j \oplus S_j) \quad j = 1, \cdots, n$

解密变换：$p_j = \mathrm{CIPH}_K^{-1}(C_j \oplus S_j) \quad j = 1, \cdots, n$

证明 令 $M_j = P_j \oplus S_j$，$N_j = \mathrm{CIPH}_K(M_j)$，则 $\mathrm{CIPH}_K^{-1}(N_j) = \mathrm{CIPH}_K^{-1}(\mathrm{CIPH}_K(M_j)) = M_j$，而 $\mathrm{CIPH}_K^{-1}(N_j) \oplus S_j = M_j \oplus S_j = P_j \oplus S_j \oplus S_j = P_j$。其中 CIPH_K 和 CIPH_K^{-1} 互为逆变换。

由表 1 可知，在计数序列中没有重复的数据出现，在实际应用中一般取 $\bmod 2^{128}$，所以增加这些随机序列能够保证扩散明文图像中存在的相同数据块，这样能够很好地消除密文图像中出现的纹理现象。

表 1 计数序列 $IV + \Sigma j$

j	0	1	2	3	4	…	12	13	14	15
Σj	0	1	3	6	10	…	78	91	105	120
S_j	5	6	8	11	15	…	3	0	14	13

3 基于 CUDA 的 AES 高效实现方法

CUDA 编程模型是 CPU + GPU 异构模式，根据 CPU 和 GPU 的特性，它们可以被用于负责不同的任务：在 GPU 上使用 CUDA API 执行可大规模数据并行的运算密集型计算任务；在 CPU 上执行控制密集型计算任务。密码算法是典型的运算密集型算法。CUDA 的优势体现在 GPU 能够执行大量并发线程上，增加并发线程的数目也是提高 CUDA 应用性能的基本手段，这也体现在 AES 算法 CUDA 实现方法中。

基于 CUDA 的 AES 算法并行程序设计时需要依据的原则：

a)提高算法的并行性，细化并行粒度，以增加执行单元的利用率；

b)尽量减少线程间处理数据的相关性，以最小化线程间通信开销；

c)合理组织线程、合理组织数据存储结构以便线程通过线程 ID 定位其所对应的数据。

3.1 算法模式选择

选择合适的工作模式，用于高效地完成 AES 加密算法，其中 ECB (electronic code book，电子密码本)和 CTR (counter，计数器)模式具有很好的并行结构。ECB 是分组密码最基本的加密工作模式，其优点是可并行运算、速度快、易于标准化。CTR 模式因为对每个数据分组使用的计数器不同而且可以同时计算，仍然适用于大规模数据并行计算。本文采用前述的新 ECB 工作模式。

3.2 存储空间分配和线程分布方案

合理设计每个 block 中的线程数以及共享存储器的大小，对提高并行度，充分利用系统资源亦十分重要。线程模块的规模是有限制的(每一个线程模块最多只能分配 512 个线程)，而在线程模块大小一定的情况下，共享存储器的空间分配也会在一定程度上限制硬件上流式处理器的并行执行能力，共享存储器分配过大或者线程分配过少时，都会影响计算能力。

GPU 和内存之间的数据交换是一笔很大的开销，因此如何合理地分配 GPU 存储空间，克服外部通信时间延迟，提高程序执行的并行性是减少这部分开销的关键。

由于同一线程块内的线程需要共享和频繁访问 AES 扩展密钥的信息，为了获得较高的执行效率，将同一线程块内的扩展密钥和部分输入数据一起装载到共享存储器内。为了充分利用共享内存，每一个线程块处理 640 个数据块(10 KB)，加上 4K 的 T 查表盒和轮密钥，保证了共享内存的高效利用。在整个加密过程中，T 查表盒首先装载到常量内存，然后加载到共享内存，而明文数据块和轮密钥首先放在全局内存中，然后导入共享内存中，如图 3 所示。

在 AES 加密 GPU 实现过程中，每一个线程块由 16 个线程组组成，每个线程组由 16 个线程组成，其具体分配过程如图 4(a)所示；每一个线程组对 M 个数据块进行处理，如图 4(b)所示；然后复制到共享内存中，协助块用来存放前一次生成的轮密钥，如图 4(c)所示。每个分组线程在 AES 的每次循环中对数据块进行处理。加密完成后，输出数据的结果将再次被写到全局设备存储器中，随后，CPU 程序将输出数据结果从 GPU 的全局设备存储器中取回，这样完成了整个加密过程。

4 性能测试

测试环境：GPU 为 NVIDIA GeForce 8800GTS，CPU 为 Intel Core DUO 2.70 GHz，内存为 2 GB。

编程环境：CUDA 4.0、Visual Studio 2010。

实验步骤：首先确定线程组织方式，以最优化计算单元的

利用率,确定对存储器的合理分配使用方案,这样做的目的是提高程序执行的并行性、促进数据间的相关性,从而提高程序的执行效率。

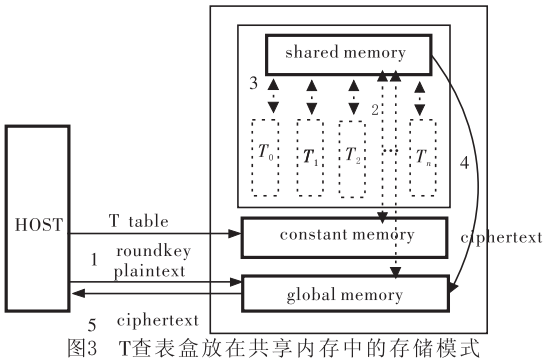


图3　T查表盒放在共享内存中的存储模式

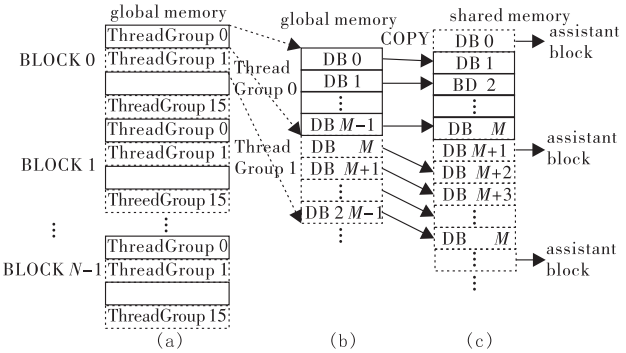
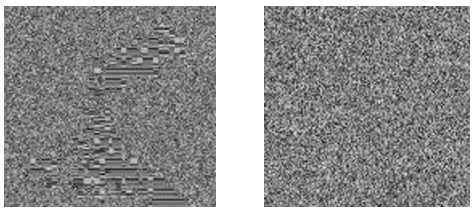


图4　明文数据通过多线程的分布

测试结果:采用图 1(a)作为明文测试图,新 ECB 模式加密效果如图5(b)所示。通过图5(b)可知,新 ECB 模式加密图像中没有出现特殊纹理属性,相同的数据信息已经完全被扩散,消失在密文图像中。



(a)明文图像的加密图　　(b)新 ECB 模式加密图

图5　新 ECB 模式加密效果

从图6可以看出,CUDA-AES 实现与 CPU-AES 实现方式相比较,CUDA 实现的时间耗费比 CPU 明显少。从图7可以看出,在明文数据规模较小的时候,加速不是很明显,这是因为 GPU 和 CPU 之间的通信效率比较低。在计算量小、数据传输时间占运行时间的比重较大。根据 Amdahl 定律可知,加速效果并不明显。随着明文数据增大,数据在 CPU 和 CPU 之间的通信效率比较低。在计算量小、数据传输时间占运行时间的比重较大。根据 Amdahl 定律可知,加速效果并不明显。随着明文数据增大,数据在 CPU 和 GPU 之间的传输时间所占用的时间比例越来越小,加速效果越来越明显。

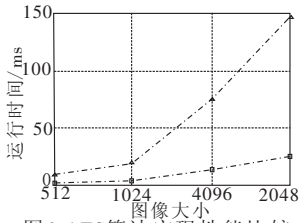


图6 AES算法实现性能比较

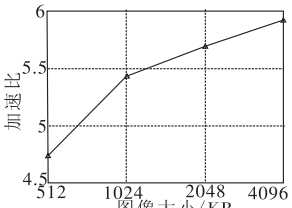


图7 AES算法加速比

在进行 CUDA 实现 AES 加密与 CPU 实现的运算性能比较时,除 GPU 执行运算外,也已经将运算过程中必需的数据传输时间计算在内,包括从主存中将数据复制到显存中,以及从显存中读出运算结果的时延。

5　结束语

本文基于 NVIDIA 的硬件平台和 CUDA 的软件编程环境,对 AES 加密算法进行了并行化优化实现,介绍了相应的线程组织、数据存储结构以及基于共享内存的性能优化。在此基础上,给出了 AES 算法的 CUDA 实现的运算性能的测试结果,并与 CPU 实现进行了比较。实现结果表明,AES 新 ECB 工作模式能够正确、可靠地保证图像信息的安全。GPU 在支持大规模的并发 AES 加密方面可获得显著的加速性能。对于大规模数据输入,数据加载和写回时间占用了总时间的大部分,从数据加载速度方面进行优化将是提升大规模数据加密输入情形的关键。

参考文献:

[1] COOK D, LOANNIDIS J, KEROMYTIS A, *et al*. Cryptographics: secret key cryptography using graphics cards[C] //Proc of RSA Conference, Cryptographer's Track (CT-RSA). 2005.

[2] MANAVSKI S A. CUDA compatible GPU as an efficient hard-ware accelerator for cryptography[C]//Proc of IEEE International Conference on Signal Processing and Communication. 2007:65-66.

[3] HARRISON O, WALDRON J. AES encryption implementation and analysis on commodity graphics processing units[C] //Proc of the 9th International Workshop on Cryptographic Hardwave and Embedded Systems. Berlin: Springer-Verlag, 2007:209-226.

[4] DAEMEN J, RIJMEN J. The design of Rijndael: AES-the advanced encryption standard [M]. New York: Springer-Verlag, 2002.

[5] HUANG C W, YEN C L, CHIANG C H, *et al*. The five modes AES applications in sounds and images [C] //Proc of the 6th International Conference on Information Assurance and Security. 2010:28-31.

[6] BANU R, VLADIMIROVA T. Fault tolerant encryption for space application [J]. IEEE Trans on Aerospace and Electronic Systems, 2009, 45(1):266-279.

[7] 田绪红,江敏杰. GPU 加速的神经网络 BP 算法[J]. 计算机应用研究,2009,26(5):1679-1691.

[8] 王磊,张春燕. 基于图像处理的通用计算模式[J]. 计算机应用研究,2009,26(6):2356-2358.

(上接第 1906 页)

[4] 代科学,李国辉. 一种基于码本的监控视频运动目标检测算法[J]. 计算机工程,2007,33(14):27-29.

[5] 齐美彬,杨爱丽,蒋建国,等. 一种基于改进码本的车辆检测与跟踪方法[J]. 中国图象图形学报,2011,16(3):406-412.

[6] 谭文明,李斌,张文聪. 基于中心对称局部二值模式的背景建模方法研究[J]. 中国科学技术大学学报,2010,40(11):1112-1116,1152.

[7] JABID T, KABIR M H, CHAE O S. Gender classification using local directional pattern (LDP)[C]//Proc of International Conference on Pattern Recognition. 2010:2162-2165.

[8] 朱晓临,邓祥龙,胡德敏. 多阈值选取与边缘连接的边缘检测算法[J]. 图学学报,2012,33(2):72-76.

[9] 李峰,周荷琴. 融合码本和纹理的双层视频背景建模方法[J]. 中国科学技术大学学报,2012,42(2):99-105.