## David Dai

3436 SouthWick Street

Mississauga, Ontario, L5M 7L4

Cell:(647)918-0834

E-Mail: djx.zhenghua@gmail.com

## Profile

An Excellent Software Engineer with an MS degree in computer science. I have a full understanding of parallel programming, algorithm design and optimization with C, C++ and assembly language. I have in-depth knowledge of programming on cluster, programming on CPU with CUDA and OpenGL, parallel programming language compiler architecture, GUI development with MFC and FLTK, and programming with script language. I have strong experience in software design, documentation and test.

## Work Experience

### Intel China Center of Parallel Computing(ICCPC) 2009- 2010

### Software Engineer

ICCPC is Intel’s first cooperative parallel computing center around the world.  It started operation in January 2010 under the management of Beijing SoftTech Technologies Co., Ltd.

* Design the Ct training material including: Ct Programming basics, Ct Functions and Ct Tools.
* Optimize application with Ct, SCC(Single Chip Cloud Computer), the applications include mpeg2 and H.264 decoder, Image process algorithms, Haartraining in openCV.
* Design a parallel language ctmpi based on Ct and MPI , which makes programming on cluster easier.
* Set up working environment and developing environment including setting up SVN, bugzilla and Develop the ERP system using VBA and javascript.

### Intel Lab(China), 2007.3~ 2009.12

### Software Engineer contractor of Ct team Employee of Softtech Inc.

* Develop the Ct Front-End with C++, the work mainly includes opcode and opcode dispatching system, dumper system, Dump the data in Ct container into text file, Exception interface, the unit test cases of Ct Front-End.
* Optimize the workloads SSE, VIP(Vector Intermediate Language ) , Ct and larrySim , CUDA. Such as FFT, LU and so on
* Design the display system of projects which are showed on IDF(Intel Developer Forum) including Golden Pig Demo, Drive-aid system.

### Seastar, China. 2006.9~2007.2

### Software Engineer

* Design the graphics work station.
* Optimize dynamic programming algorithm on GPU and GPU cluster.
* Optimize application on Cell.

## Education

* MSc. Computer architecture, June. 2006

The Institute of Computing Technology of Chinese Academy of Sciences

Master thesis: Optimization of String Matching Algorithm Based on Computer Architecture

* BSc. Computer Science, June, 2003

Shandong Normal University, Jinan, China

## Technical Skills

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Skills | Years | Level |  | Skills | Years | Level |
| C | 6 | Expert |  | OPENMP | 5 | Expert |
| C++ | 4 | Expert |  | Ct(ABB) | 4 | Expert |
| MPI | 5 | Expert |  | SSE | 6 | Expert |
| CUDA | 3 | Expert |  | OPENGL | 5 | Intermediate |
| Perl | 2 | Intermediate |  | Bash | 4 | Intermediate |

## Interesting Projects

Golden Pig Demo, which was showed on IDF(Intel Developer Forum) 2007 spring(Beijing).

* I designed and implemented the display system with OpenGL. The demo simulates the motion of 20,000 atoms in limited space.

* [http://video.google.com/videoplay?docid=-369279912802383146&hl=en#](http://video.google.com/videoplay?docid=-369279912802383146&hl=en)

Drive-aid system, which was Showed on IDF 2008 spring(Beijing).

* I designed and implemented the display system with MFC. The Drive-aid system detects the objects nearby through analyzing the video captured by camera, and displays the objects to the driver.

* [http://video.google.com/videoplay?docid=-1814219667765323229&hl=en#](http://video.google.com/videoplay?docid=-1814219667765323229&hl=en)

## Open Source

* Library of **String matching** algorithms, includes:
  + Single pattern sting matching and Multiple pattern string matching algorithms.
  + The very fast strstr function(optimized with SSE2), **4~10x speedup** to strstr of glibc.
  + Accessing the source code: svn co <http://strstrsse.googlecode.com/svn/trunk/>
* Cilk-like **Thread Interface**(implemented with C macro), with which one can
  + Create thread just like calling a function: eg. slaunch3(foo)(arg1, arg2, arg3);
  + Dispatch a task to a suitable thread : eg. dlaunch4(foo)(arg1, arg2, arg3,arg4);
  + Accessing the source code: svn co <http://codelibrary.googlecode.com/svn/trunk/>
* A web browser kernel in which the rendering engine is implemented with OpenGL.
  + Display the html page as 3-D scene.
  + You can view the source code at: <http://code.google.com/p/matrixseer/>
* A tool of processing and analyzing video and image, including
  + Collection of video and image processing algorithms.
  + A compiler that compiles high level video-processing-language.
  + <http://code.google.com/p/videoanalyze/>