Patrick Stein

(formerly Patrick Fleckenstein)
http://www.csh.rit.edu/~pat/me/resume/

5537 2nd Ave S
Minneapolis, MN 55419-1404
pat@nklein.com
(651) 334-7364

Objective

I am seeking challenging employment which will exercise and extend my skills. Jobs offering a focus in mathematics, image processing, operating systems, or networking are preferred.

Publications

LUT Filters for Quantized Processing of Signals. Ricardo L. de Queiroz and Patrick Fleckenstein. *IEEE Transactions on Signal Processing, Vol. 52, pp. 687–693 (March 2004)*.

Signal Processing Using LUT Filters Based on Hierarchical VQ. Ricardo L. de Queiroz and Patrick Fleckenstein. *Proceedings of IEEE International Conference on Acoustic, Speech, and Signal Processing, (ICASSP), SPTM-P1 (May 2001)*.

Very Fast JPEG Compression Using Hierarchical Vector Quantization. Ricardo L. de Queiroz and Patrick Fleckenstein. *IEEE Signal Processing Letters, Vol. 7, No. 5 (May 2000)*.

Fast JPEG Encoding for Color Fax Using HVQ. Ricardo L. de Queiroz and Patrick Fleckenstein. *Proceedings of IS&T/SPIE Symposium on Electronic Imaging, Color Imaging, San Jose (Jan 2000)*.

Patents

Patent 6,683,994: HVQ-based Filtering Method. Ricardo L. de Queiroz and Patrick Fleckenstein. Issued January 27, 2004.

Patent 6,286,026: Method and Apparatus for Integrating Pull and Push Tasks in Pipeline Data Processing. Dennis L. Venable, Patrick A. Fleckenstein, James E. Bollman. Issued September 4, 2001.

Patent 6,292,168: Period-based Bit Conversion Method and Apparatus for Digital Image Processing. Dennis L. Venable, Patrick A. Fleckenstein, William A. Fuss. Issued September 18, 2001.

Technical Familiarity

Languages: Lisp, C++, Perl, Unix shells, Objective-C, C, sed, make, Java, PHP, JSP, XML, HTML, XSLT, CSS, and SQL. Also: lex, yacc, PostScript, python, Fortran, Modula-2, COBOL, Logo, BASIC, and various assembly languages.

Operating Systems (and hardware): Mac OS X (x86, PPC), Linux (x86, ARM), Solaris and SunOS (various), BeOS (x86), NetBSD (CCI Okee, x86, DECStation, Sparc), Ultrix (MIPS, DECStation, KIMStation, VAX), System V/88 (Motorola 88k), and Windows NT/95/3.1 (x86).

Concepts and Packages: POSIX, STL, CORBA, UML, SDL, Kerberos V, XLib, threads, and sockets. Also: emacs, vi, git, Subversion, CVS, LaTeX, noweb, gnuplot, and gcc/g++.

Mathematics & Computer Science Education

B.S. in Computational Mathematics from the Rochester Institute of Technology in Rochester, NY.

Mathematics courses include: Numerical methods, discrete mathematics, math modeling, probability, applied statistics, set theory, abstract algebra, complex variables, real variables, graph theory, game theory, number theory, differential equations, and topology.

Computer Science courses include: Data communications, data organization and management, assembly language, and digital image processing.

Work Experience

C++/Java Developer

Developed C++ and Java code on Set Top Boxes for a satellite television company. This job has involved the full Set Top Box software stack from the custom device drivers in the Linux kernel for MPEG decoding the whole way up to buttons and messages on the Java/OpenGL GUI.

Nevelex Corporation, Bloomington, MN. 2015–Present.

Lisp Software Developer

I was the second full-time Lisp developer brought in on this communications system. My initial work centered on interfacing with GPS sensors and DSP hardware.

Clozure Associates, Boston, MA. 2013–2015.

Team lead and C++ developer

I am the team lead of and a developer for the Data Central Services team in the ERAM Display Systems group and I am co-lead of the Display Systems failure analysis group. We are developing the next generation of the National Air Traffic Control System. The Data Central Services is responsible for the infrastructure used to synchronize display settings between the active and backup channels of the system.

This work involved extensive development in C++, extensive debugging in a large C++/Ada environment, and team management.

Lockheed Martin, Eagan, MN. 2010–2013.

Information technology professional

For this team, I did development and maintenance for the University of Minnesota's MyU Portal . A large piece of that effort involved collating, several times a day, disparate data sources and learning tools into a cohesive list of courses for each of the 170,000 users. This job also required frequent culling of log outputs and databases to generate reports and to narrow down problems.

I had toyed with Perl before this job. Thanks to this job, I am now an expert Perl programmer.

For this team, I also administered software packages and did user support for a small, research computing cluster owned by the Department of Soil, Water, and Climate in the College of Food, Agriculture, and Natural Resource Sciences.

University of Minnesota's Academic and Distributed Computing Services, Minneapolis, MN. 2007–2008.

System software engineer

On this project, I developed a geoimaging archive and access interface employing Google Maps , Java, and SQL.

Earlier, we integrated the various science instruments on the NASA/USRA SOFIA project and put the framework in place to allow relentless observing. I did a great deal of design work and implementation for this project in C++, C, and Java with heavy use of CORBA, XML, XSLT, JSP, and SQL.

Rochester Institute of Technology's Center for Imaging Science, Rochester, NY. 1999–2007.

Computer graphics programmer

In this position, I did a wide variety of things relating to digital image processing. For the last year, I was lead programmer for our project. I received an appreciation award for work I did with a sibling group to design an API for their toolkit and another award for work I did integrating the image processing modules, communication module, and user-interface for another project. I did a great deal in this position doing image compression research, developing new image processing algorithms, creating and parsing image description languages, creating user-interfaces, integrating large-scale projects, and developing for platform-independence.

Xerox Corporation [DITC/Cadisys/Sandpiper], Rochester, NY. 1991–1995, 1996–1999.

Natural language database retrieval programmer

I hit the ground running when I joined this startup company. The company's lead product is a natural language interface to several databases (focusing on software prior art and financial advising). The system architecture is client-server with the server being composed of many separate daemons running on an arbitrary number of machines. My initial work on this product was implementing various sections of the server.

I quickly became involved in all aspects of the client-server system. I did much of the work on the communications layers in both the client and server, chunks of the user-interface, various logging and debugging mechanisms, and overall system integrity.

This project, even more so than my work at Xerox, highlighted my ability to see and work with the gestalt in a complex system. I am quite proud of this ability.

Manning and Napier Information Services, Rochester, NY. 1995–1996.

Multimedia education software programmer (PanVision)

I designed and implemented (not necessarily linearly) a programming language for the creation of multimedia applications under Windows 3.1. This software has been incorporated in several educational CD-ROMs geared toward German pre-reading-age students and in several promotional CD-ROMs.

PanVision, Essen, Germany. 1994–1996.

Advanced research group member

As a member of the Advanced Research Group at Computer Science House, I have participated in many projects. In addition to the day-to-day system administration of a network of BSD, System V, and Ultrix machines, I have worked on software for our light show, our Coke machine, and the four projects listed below.

Computer Science House, Rochester, NY. 1993–1998.

NetBSD kernel port

Two others and I (working from some sketchy documentation, the NetBSD Amiga release, and the 4.3 BSD port for the Power 6/32) ported NetBSD to the CCI Power 6/40 Okee architecture. This project involved implementing `locore.s` and the Okee specific device drivers. Most of the work I did on this project was in porting the device driver for the VersaBus SMD disk controller. I also assisted in debugging several device drivers and the whole of locore.

X Windows Server port

Computer Science House received a large donation from Kodak of 2-bit grayscale, high-resolution workstations. We ported the X Windows sample server to run on these machines under Ultrix. I made many modifications to the server code to get the server functioning well, including many fixes on the server side to deal with X clients that were less-than-intelligent about dealing with 2-bit displays.

On this project, I learned X Windows from the inside out. I had perused Volume 0, but the rest I learned from reading the include files. I have employed this method of learning on various systems, and it has proved particularly effective for me.

Inferno DIS disassembler

I wrote a disassembler for DIS objects in C. I based it on the documentation provided on the web about the DIS virtual machine. This project was the first step toward implementing a DIS virtual machine, but my interest in this project waned before completion of the VM.

Virtual Reality network

I was a key implementer and designer in the early revisions of the Virtual Reality/Learning Systems Development project at Computer Science House. On this project, I made extensive use of socket communication under Unix and AmigaDOS.

Other Programming Activities

Just Change Law

I designed an implemented this website/CMS using PHP and MySQL.

Just Change Law, St. Paul, MN. 2008–2009.

MN Writes

In less than a month of spare time, I did all of the PHP and MySQL which drove the now defunct Minnesota Writes web site.

Minnesota Writes, Minneapolis, MN. 2005.

nklein software

I started the web site for the distribution of my own software projects. Most of my focus has been on software with a multi-dimensional bent. Several of these projects are highlighted below.

nklein software. 1998–present.

Open-Source Lisp Libraries

I have written a variety of Open-Source Software in Lisp including: mathematical software, networking software, infrastructure software, GUI software, and utility software.

---

iPhone app for kids

I built an iPhone app that helps children learn to spell basic words in English, French, or Spanish. I wrote the application in Objective-C atop the iPhone-cocos2d framework. I did all interface design, artwork, and coding for this application.

---

n-dimensional raytracing

I have developed a raytracer which supports an arbitrary number of spatial dimensions. It supports cylindrical objects (with an arbitrary number of ”round” dimensions), quadratic surfaces, CSG objects, convex hulls, Coxeter polytopes, extrusions, arbitrary rotations, arbitrary scalings, JPEG output, ppm output, PostScript output, multiple light sources, directional light sources, texture-mapping, reflections, and refraction. The original implementation was in C++. There is a partial re-implemenation in Common Lisp.

Clifford algebras library

I created a C++ template library and a Lisp library for Clifford algebras and employed them to do color convolution filtering and fractal generation.

54321

This C++ program is five games in four-, three-, and two-dimensions for one player. This game won an honorable mention in the SDL 1MB Linux Game Programming Contest sponsored by No Starch Press, Linux Journal, and Loki Software.

DVD Tracks

I developed the entire DVD Tracks web site with PHP and MySQL in less than two weeks of my spare time. This site is no longer in use.

DVD Tracks, Minneapolis, MN. 2002–2004.

Operating systems overview seminars

I prepared and gave a two-hour presentation on the fundamental concepts of modern operating systems. This presentation was given as the introductory session of the Computer Science House Operating System Seminar Series. For this seminar, I focused mainly on how system calls differ from library calls and how the operating system arbitrates device contention.

Computer Science House, Rochester, NY. 1996.