

KIRAN RAJ S.R.

MS by Research (CSE) student at IIIT Hyderabad

Contact Information

318, NBH, IIIT Hyderabad
kiran.raj@research.iiit.ac.in
+91-897-831-6426
<http://cstar.iiit.ac.in/~kiranraj>

Areas of Interest

- High Performance Computing
- Machine Learning

Work Experience

IIIT Hyderabad, Research Assistant, July 2014 - Present

- Web application developer (Web2Py)

IIIT Hyderabad, Teaching Assistant, Jan 2014 - May 2014

- Data Structures

Freelancer, Full stack web developer, Dec 2011 - Aug 2013

- Developed & deployed many web apps for early stage startups and clients.

Cognizant Technology Solutions, Programmer Analyst, Oct 2010 – Dec 2011

- Ariba application support.

Education

MS by Research in Computer Science & Engineering

IIIT Hyderabad, July 2013 – Present

Current CGPA: 9.17

B.Tech. in Computer Science & Engineering

St. Joseph's college of Engineering, Chennai, Aug 2006 – June 2010

Percentage: 82 %

Publications / Submissions

- [*Published*] Kiran Raj Ramamoorthy, Dip Sankar Banerjee, Kannan Srinathan, Kishore Kothapalli, "A Novel Heterogeneous Algorithm for Multiplying Scale-Free Sparse Matrices", to **IEEE- IPDPS ASHES 2015** (International workshop on Parallel and Distributed Scientific and Engineering Computing).

Projects

- **Source to Source Translator using Rose Compiler:** A source-to-source compiler translates between programming languages that operate at approximately the same level of abstraction. In this project, we have tried to develop a source-to-source compiler that automatically translates the given C program into another C program with instructions replaced for the target platform (Universal Multifunction Accelerator[UMA], developed by Manjeera Digital Systems) that supports an extended instruction set. In order to accomplish this task we take help of Rose compiler, an open source compiler infrastructure to build source- to-source compilers.
- **Finding Convex Hull using Cache Aware/Software Prefetching based GPU Algorithms:** In this project, we studied different parallel convex hull algorithms for GPU and their memory access patterns. We then identified best parallel convex hull algorithm and introduced cache aware & software prefetching techniques to improve efficiency. Also we conducted experimentations using latest GPU's like NVIDIA GTX 580/680 that has cache system and reported the results for comparison.
- **Face recognition using Eigen Faces:** We implemented a face recognition system using PCA (Principle Component Analysis) and SVM (Support Vector Machines). PCA reduces the high dimensional image to a low dimensional space (referred to as eigen space). Computing the eigen vectors of scatter matrix (of the training samples) and selecting Top k eigen vectors are two key tasks in this process. We have implemented the above techniques on standard datasets maintained by Yale University, CMU University and also on a real time dataset generated during CSE471(Statistical Methods in AI) class at IIIT Hyderabad. We also performed validation, verification experiments on these datasets with the above-mentioned techniques, recorded results and inferences.
- **POS Tagging and CYK Parsing for Indian Languages:** We implemented POS taggers for Hindi, Tamil and Telugu using supervised (Hidden Markov Model) and unsupervised (clustering) approaches. We also used CYK algorithm on datasets of these languages. Python was used for programming.
- **Effective Text Categorization Using Compactness and Position of First Appearance:** Our project automatically sorts a set of documents into categories from a predefined set. In contrast to using the traditional term frequency values solely, including the distributional features requires only a little additional cost, while the categorization performance was significantly improved.

Skill Set

- Parallel Programming:** CUDA, OpenMP, OpenACC, Libraries (Thrust, CuSparse, CUSP)
- Programming:** Python, C
- Web Development:** HTML, CSS, JavaScript, PHP, jQuery
- Web Frameworks:** CodeIgniter, Laravel, Web2Py
- Others:** MySQL, Ariba 9r1, git

Recent Courses

- Statistical methods in AI
- Natural Language Processing
- Artificial Neural Networks
- Machine Learning (Audited)
- Parallel Programming
- Advanced Compilers
- Complexity and Advanced Algorithms