


PhD Computer Science/Engineering/BioInformatics RAJA MAJID MEHMOOD	Home: Inhudong 2 202 Villa Lucky No. 1571-11, Deokjin- gu, Jeonju-si 561-853, Jeollabuk-do, South Korea	
EMAIL: rmeex07@gmail.com, rmeex07@chonbuk.ac.kr MOBILE: +82 10 6534 3637	Office: LAB 7312, COMPUTER ENGINEERING DEPT. Chonbuk National University, Jeonju-si 561-756 Jeollabuk- do, South Korea	

MOTIVATION

It is my passion to carry on a creative and novel research after completion of my PhD. Therefore, I have a very strong intension to join the academia as a Assistant Professor which will help me to boost my expertise through teaching and other research activities.

EDUCATION

- PhD in Computer Science, Chonbuk National University, Jeonju, South Korea (032014~022017)
- Master in Software Technology, Linnaeus University, Sweden (092007~092009)
- Master in Software Technology, VAXJO University, Sweden (092007~092008)
- Post Graduate Diploma in Software Engineering, LAPS, LONDON, England (092004~092005)
- BS in Computer Science, KICSIT, KRL, Kahuta, Pakistan (032000~062004)

CURRENT RESEARCH FIELD (PhD Research)

Nowadays, I am working on Electroencephalography (EEG) based emotion recognition and analysis of brain signals such as Event Related Potential (ERP). This area is related with Brain Computer Interaction and Signal Processing. The major application of our research may contribute to patients with some mental disorder. More research details are located in section of *Research Publications*.

Graduation: (Expected by: 22 Feb 2017)

Duration: 032014 ~ 022017

Occupation: PhD Research Student

Supervisor: Hyo Jong Lee

E-mail: hlee@chonbuk.ac.kr

Phone: +82 (0) 63-270-2407

Address: Office 7326, COMPUTER ENGINEERING DEPT. Chonbuk National University,
 Jeonju-si 561-756 Jeollabuk-do, South Korea

PREVIOUS WORK

- *Department of Software Engineering, King Saud University, Saudi Arabia.*
<http://fac.ksu.edu.sa/rmehmood>

Duration: 042011~032014

Occupation: Lecturer/Researcher

Supervisor: Abdulla Al Ghamdi

Web: <http://faculty.ksu.edu.sa/abdksu/Pages/abdullah-home.aspx>

E-mail: ghamdi@ccis.ksu.edu.sa

Phone: +966 (01) 4695224 / +966 (01) 4695225 / +966 (01) 4695227

Research Projects

1. C4I Emergency Alert System
2. Cloud-based approach for interoperable C4I System

TEACHING EXPERIENCE

- Lecturer in Software Engineering Dept. King Saud University, Saudi Arabia (042011~032014)
- Computer Science Lecturer in Bahria University, Islamabad, Pakistan (012011~042011)
- Computer Science Lecturer in KIPS, Virtual University, Pakistan (122005~062007).
- Lecturer in KICSIT (Computer Science), Pakistan (122005~082006)

RESEARCH PUBLICATIONS

1. Mehmood, R. M. and H. J. Lee (2016). "A Novel LPP based Feature Extraction Method for Emotion Recognition of EEG Brain Signal Patterns". Computers & Electrical Engineering, International Journal (Accepted).
2. Mehmood, R. M. and H. J. Lee (2016). "Hjorth Parameters Features Used for Recognition of Human Emotions from Brain Signals". Mobile Information Systems, International Journal (Sumitted).
3. Mehmood, R. M. and H. J. Lee (2015). "Exploration of Prominent Frequency Wave in EEG Signals from Brain Sensors Network." International Journal of Distributed Sensor Networks.
4. Mehmood, R. M. and H. J. Lee (2015). "EEG based Emotion Recognition from Human Brain using Hjorth Parameters and SVM." International Journal of Bio-Science and Bio-Technology Vol.7, No.3 (2015), pp.23-32. <http://dx.doi.org/10.14257/ijbsbt.2015.7.3.03>.
5. Du, R., R. M. Mehmood, et al. (2014). "Alpha activity during emotional experience revealed by ERSP." Journal of Internet Technology 15(5): 775-782.
6. Mehmood, R. M. and H. J. Lee (2015). "Towards emotion recognition of EEG brain signals using Hjorth parameters and SVM". Bioscience and Medical Research 2015, vol. 91, pp. 24-27, 2015. URL: http://onlinepresent.org/proceedings/vol91_2015/5.pdf
7. Gao, Y., Mehmood, R. M., H. J. Lee, et al. (2015). Deep learninig of EEG signals for emotion recognition. Multimedia & Expo Workshops (ICMEW), 2015 IEEE International Conference on, IEEE.
8. Mehmood, R. M. and H. J. Lee (2015). Emotion classification of EEG brain signal using SVM and KNN. Multimedia & Expo Workshops (ICMEW), 2015 IEEE International Conference on, IEEE.
9. Mehmood, R. M. and H. J. Lee (2015). ERP analysis of emotional stimuli from brain EEG signal. Int'l Conf. Biomedical Engineering and Science | BIOENG'15.
10. Mehmood, R. M. and G. Iqbal (2010). "Visualization of Statistical Contents." (MS Thesis)
11. Raja Majid Mehmood, Syed Nasir Mehmood Shah, *Research Paper Published on 28th-30th October 2010, in "International Conference on Intelligence and Information Technology (ICIIT-2010)", Lahore, Pakistan.*
12. Majid, R. (2008). "Dynamic and Static Approaches for Glyph-Based Visualization of Software Metrics." (MS Thesis)

MASTER DEGREE THESES

- **TITLE: Visualization of Statistical Contents, Master Thesis (30 HE credits)**

Our project presents the research on visualization of statistical contents. Here we will introduce the concepts of visualization, software quality metrics and proposed visualization technique (line chart). Our aim to study the existing visualization techniques for visualization of software metrics and then proposed the visualization approach that is more time efficient and easy to perceive by viewer.

In this project, we focus on the practical aspects of visualization of multiple projects with respect to the versions and metrics. This project also gives an implementation of proposed visualization techniques of software metrics. In this research based work, we have to compare practically the proposed visualization approaches. We will discuss the software development life cycle of our proposed visualization system, and we will also describe the complete software implementation of implemented software.

Areas of expertise: Object-oriented analysis and design, Java programming, 2D & 3D graphics, Logics, Database, Java chart api

- **TITLE: Space-Filling Visualization of Software Metrics, Master Thesis (15 HE credits)**

This project presents a technique that allows viewers to visually analyze, explore, and compare a raw dataset. We present an algorithm that visualizes data along a traditional *2D space filling grid*. We use graphical "glyphs" (simple geometric objects) that vary in color, placement and texture properties to represent the attribute values contained in a data element. When shown together, the glyphs form visual patterns that support exploration, facilitate discovery of data characteristics, and highlight trends and exceptions. Here we identified four important goals for our research:

- To design graphical glyphs that support flexibility in their placement, and in their ability to represent multivariate data objects.
- Build effective visualization techniques that use glyphs to represent each data object on a multivariate dataset.
- To display the attributes and their values using animation.
- We have focused on theoretical and practical comparison of two different approaches to display graphically same multivariate raw dataset.

Our intent is to show not just the intersection of multiple dataset, but also specific details about structure of the data objects. This information can be critical to understanding how the original data objects are themselves related. To our best knowledge, the combination of multivariate display techniques, perception, and animation for direct comparison of different perspectives into a dataset is a useful and novel contribution to the field of software visualization.

Our primary objective in this project is to design a graphical glyph that supports flexibility in its placement, and in its ability to represent multivariate data objects. We focused to accommodate high number of attributes by small graphical representation.

The main part of this project is to develop a software tool that can accept any type of multivariate raw data and represent it graphically with high quality and efficiency. The data exists in any type of form, our software able to read the dataset from any data source; it can be any type of file system.

Areas of expertise: Object-oriented analysis and design, Java programming, 2D & 3D graphics, Logics

PREVIOUS RESEARCH WORK:

- **Hierarchy Graph (Applied Program Analysis)**

Reading/Parsing Process

1. Run ForestWalker
2. Extract required nodes (*Class/Interface/Abstract Class*)
3. Get all super types of current node (*Class/Interface/Abstract Class*)
4. Put into Map<Node-Name, List-Of-Super-Types-Names>

Graph Generating

This algorithm shows the hierarchy of all classes/interfaces involved in given source code.

Display in GML format:

Simply use the existing implementation, pass the graph object (hierarchyGraph()) to MyGML class to create the GML file with following name: *DG-HierarchyGraph.gml*.

- **Call Graph (Applied Program Analysis)**

Parsing Process

1. Run ForestWalker
2. Extract required nodes (*Methods/Constructor*)
3. Get all References (*MethodReference/ConstructorReference*) of current node
4. Put into Map<Node-Name, List-Of-References-Names>

Graph Generating

callGraphGenerator (blockName, List-of-References-Names) gives a call graph

Display in GML format:

Simply use the existing implementation, pass the graph object to GML class to create the GML file with following name: *DG-CallGraph.gml*.

- **Cluster Analysis (Applied Program Analysis)**

Implements a cluster analysis based on the call graph constructed before. Therefore.

1. Aggregate the call graph to a member usage graph.
2. Implement preparative analyses, i.e., remove utility classes (String, Hashset, etc.), aggregate inner classes, and aggregate single entry sub-trees of your member usage graph.
3. Construct a dominance tree based on the aggregated usage graph – mark direct and strong dominance edges in the tree.
4. Suggest a hierarchical clustering based on the dominance tree computed: interpret each sub-tree of the dominance tree as a component.

- **Java Tiny Compiler (Compiler Construction)**

1. Design and construction context free grammar
2. By using javacc, construct the AST nodes (parse tree) of given CFG
3. Construct *Symbol Table* that contains all information of each identifier
4. Doing *Type Checking*
5. Machine independent *Intermediate Code Generation* and
6. Build a *Stack Machine* that can execute the input code.

- **Parser and static semantics (Compiler Construction)**

Implement the analysis phase for Tiny Java

1. Scanner definitions
2. Parser
3. Abstract Syntax Tree (AST) construction
4. Static semantic analysis (AST attribution, some attributes stored in a Definition Table)
5. Test the correctness of analysis phase

- **Task graph generation and interpretation (Compiler Construction)**

1. Implements task graph generation using AGs
2. Write an interpreter for the generated task graphs
3. Test the implementation with a set of test programs. Are the outputs correct? Compare with results that the "real" Java yields for your test programs!

- **SSA construction (Static Single Assignment)**

Write a mapping from task graphs to SSA-graphs

1. Construct a basic block graph first:
 - A new basic block begins for each node that has more than one predecessor
 - A basic block ends with a control-flow operation ((un-)conditional jump, return)
2. Then, perform SSA construction
3. Use a graph-based approach where nodes represent operations and edges represent local variables and control flow
4. Omit memory-accessing operations (Load, Store, Call, Alloc)
 - If a task graph contains memory accessing operations, retain the original task graph for later use in your program

Write a mapping back to task graphs

1. Remove Φ functions
2. Perform copy-propagation

- **λ – calculus (Programming Language Paradigm)**

Implemented the following features in this compiler

1. lambda.parser
2. lambda.runtime
3. lambda.scanner
4. lambda.semantics

- **Graph Implementation (Software Technology Project)**

To develop the algorithms for most common problems in Directed graph. And choose best algorithm for final implementation. The problems include Graph traversal, connected components, transitive closure, Dominance tree, strongly connected components.

Areas of expertise: Java programming, Algorithm analysis and design

MAJOR SOFTWARE PROJECTS

EEG based Computer Aided Educational System	
Description	A computer aided education system (CAES) which is capable of detecting the current mood of subject during his/her class time. The proposed system provide a facility to Instructor in special schools where student having a learning difficulties due to their physiological disorder. It helps the Instructor to manage the student during the learning process.
Functions	<ul style="list-style-type: none"> • User (Student) Information Module <ul style="list-style-type: none"> ○ It contains the existing user of system ○ It allows to create new User or Subject ○ It allows the User to connect with brain device • Device Control Module (Headset Setup) <ul style="list-style-type: none"> ○ It shows the current status of all device electrodes ○ It shows the signal activity in brain through each electrode in 2d Chart ○ It shows the battery status of device ○ Events Module (This module records all activities of system such as, device connection, exceptions, training, user login, logoff, etc). • Expressive Module <ul style="list-style-type: none"> ○ It shows the current mood of subject ○ It records current mood of subject ○ It allows the Instructor to add or update new treatment of current subject ○ It guides the instructor about the treatment plan for current subject
Development	Java 1.8, Java2D, Java2DCharts, Matlab2013a, EEGLAB

Visualization tool for Micro-CT large datasets	
Description	This visualization tool is developed in Java, and it is based on ImageJ open source api. This tool is already deployed in GFBR LAB, and use for normal statistical analysis of scanned images. The main components of this tool are,
Functions	<ul style="list-style-type: none"> • User Interface; attractive user friendly interface • Color Editor; it is main motivation of this project, where user is enable to change the color on demand. Also automatic function can apply default colors on large images dataset according to different density levels of image. • Cropping; it is very useful function that enables to subtract the selected Region of Interest by its image Background. • Scaling; it is simple function that just provide a facility to resize the image with its background. Also it provides to scale only selected ROI. • Statistical Analysis; user easily calculate few parameters of scanned images, like; surface density, volume, width, height, area of selected dataset. • 3D Rendering; Easily render 3D model of selected dataset. • Movie Generator; Can export the AVI format movie file.
Development	Java 1.6, ImageJ, Java2D, Java3D, Java Threads

Visualization tool for statistical analysis of software systems	
Description	This project presents a technique that allows viewers to visually analyze, explore, and compare a statistical data of different projects. We present an algorithm that visualizes data along 2D-Charts. We use simple graphical objects that vary in color and placement properties to represent the attribute values contained in a data element. When shown together, the glyphs form visual patterns that support exploration, facilitate discovery of data characteristics, and highlight trends and exceptions.
Functions	<ul style="list-style-type: none"> • Designing of Database (in MySQL) • Develop the Web-Based visualization tool on client-side <ul style="list-style-type: none"> • 2D Charts • Editing Features (Colors, Texture, Styles) • Downloading and Uploading Utilities • Web-Service that provides the interface between

	<ul style="list-style-type: none"> • <i>online Visualization-Tool and Database</i> • Visualization Technique that is more <ul style="list-style-type: none"> • <i>user friendly and time efficient for a large dataset</i> • Apply statistical computation on given input/data • Different Operations on <ul style="list-style-type: none"> • <i>Database and local File System</i>
Development	Java 1.5, Chart2D, Java Threads

Visualization tool for analysis of software metrics	
Description	The main part of this software project is to develop a software tool that can accept any type of multivariate raw data and represent it graphically with high quality and efficiency. Our software able to read the dataset from data source i.e. MS Excel.
Functions	<ul style="list-style-type: none"> • Static Space-Filling Grid • Dynamic Space-Filling Grid • Main Controller Window • Attribute Definition Window • Attribute Comparison Slider • Data Object Slider • Dataset Slider • SF-Grid Display Settings • Color Picker
Development	Java 1.5, Java2D, SWT, Java Threads

- **Dynamic website for GFBR, www.gfbr-ksu.com**
- **Bi-Lingual chat system (Multi Threaded System)**
- **Network Administration on Mobile (Multi Threaded System)**
- **Media Player for Window98+**

TECHNICAL COMPUTING SKILLS

Operating Sys:-

Windows 9x/NT/2000/XP, LINUX, DOS

Database:-

Oracle 8i, MS Access 2002, Transact-SQL, SQL Server 7.0/2xxx

Tools:-

Eclipse, Mx Flash 6.0, Fireworks, Dream Weaver, Adobe Photoshop, Oracle Developer 6i, Crystal Report Writer 8.5, Microsoft Front Page 2002

Languages:-

- Matlab, EEGLAB
- Object Oriented Programming Using JAVA, C/C++
- Mobile Programming Using J2ME (Java 2 Micro Edition)
- Network Programming Using JAVA (Socket Programming)
- Java, Spring, Hibernate, JSP, Servlets, JDBC, EJB, JNI, RMI, JMF
- Web Development (HTML, XML, ASP, JSP, Vb Script, Java Script)
- Visual Basic 6.0, VB.NET, PHP, SQL, Crystal Reports 8.5, PL/SQL

Software Design and Development:-

- Software development life cycle (SDLC)
- Structured System Analysis and Design Method (SSADM)
- Unified Modelling Language UML.
- Object Oriented Analysis (OOA), Object Oriented Design (OOD)

Major Interested Courses:-

- Applied Program Analysis
- Compiler Construction
- Graph Drawing with Theory and Practical
- Network Programming using Java J2SE
- Data Mining and Visualization
- Web Services
- Software from Components
- Data Structure using Java, C/C++
- Algorithm Analysis & Design
- Object Oriented Programming using JAVA, C++
- Advanced Software Engineering (SDLC)
- Object Oriented Modeling & Design Using UML
- Wireless Computing & Mobile Programming (WAP) using J2ME

MS Research work:-

- Graph Implementation in JAVA
- Algorithm analysis and design
- Dijkstra Algorithm for Shortest path finding in directed graph
- Implementation of different graph algorithms Transitive closure, strongly connected components, dominance tree, connected components in Directed graph.
- Compiler Construction,
- Design and Development of Java mini compiler.
- Parsing, Symbol table generation and type checking.
- Research on algorithm and programming languages

TRAVEL HISTORY

- USA (2016)
- Schengen (Italy) 2015
- Saudi Arabia 2011
- Denmark 2010
- Sweden 2007
- Schengen (France/Spain) 2007
- United Kingdom 2004