DESIGN QUALITY METRICS FOR REMOTE SENSING APPLICATIONS

T. Ahmad, M. F. Hayat, M. Afzal and H.M.S. Asif

Department of Computer Science and Engineering, University of Engineering and Technology, Lahore, Pakistan.
Corressponding Author E-mail: tauqir_ahmad@hotmail.com

ABSTRACT: Remote Sensing is vital for data acquisition and interpretation of information for subsequent use about an object or phenomenon which is not in physical contact. Remote sensing activity caters heterogeneous, temporal/spatial nature and huge amount of data; therefore remote sensing systems must be reliable and bear a good quality. Design metrics can be helpful to measure the quality and efficiency of a system. The design metrics for conventional application areas are unsuitable due to peculiar characteristics of remote sensing applications/systems. To address the quality and efficiency issues we have proposed novel design quality metrics for remote sensing applications in this study.

Key words: remote sensing, design metrics, software quality, software design methodologies, GIS systems.

INTRODUCTION

Remote Sensing (RS) involves in receiving and observing objects or phenomenon information on the surface of Earth from distance by using some real-time wireless sensing device(s) (Longley et al, 2002). RS technologies are precious tools to efficient support early disasters warnings. These technologies authorize acquisition of data and atmospheric conditions monitoring which may cause natural disasters. Due to dynamic, temporal and evolutionary nature of RS data, timely decisions making for crucial issues becomes possible (Zlatanova, and Nayak, 2012).

Design Metrics provide standards for efficient and good quality product (Stephen, 2003). These metrics containing information about an application can be used in several phases of application development. Due to diverse nature of RS applications, conventional techniques (SADT and OO) are not capable to provide some admirable solution to RS applications. To the best our knowledge, no software development technique is reported in literature for analysis, design, implementation and testing for RS applications. Design and measurement (numbers, variables or symbols are consigned to characteristics of entities) directly affect the system's performance. Software measurements (length of program, structure, and correctness) facilitate to understand, control flow, and improve the software applications. A valuable measurement can clearly differentiate the features of one unit from another with the help of analysis and then draw the conclusion. To gauge the attributes of an object, Software Metrics (SM) are used. Generally, quality/significance of a software application/system is sturdily dependent on the quality of design. The main issues of a good design are: (i) Ease of deployment (ii) Real time Efficiency (iii) Huge amount of data (iv) Usability element (v) Heterogeneity (vi) Dynamicity (vii) Scalability (viii) Reliability (ix) Feasibility of installation (x) Auto-Configuration/ Self-organization (xi) Distributed intelligence/ Adaptability (xii) Cost effectiveness (xiii) Ease of use of systems and (xiv) Fault-tolerance (Philippe, 2004).

The design phase is considered to be a time consuming and most expensive phase during the software development because the definitions of its high and low-level structures are defined and resources needed are discussed in this phase. The design and measurement processes enable a software engineer to choose right platform, considering the complexity of a RS application (Keith, 2000).

RS field is now mature enough but less attention has been given to measure design quality of such type of applications. We need to have design quality metrics for RS applications because the existing ones for conventional domains are unsuitable due to peculiar and diverse nature of RS applications (Ayas, 2011). To achieve this goal, an extensive literature survey has been conducted over software development methodologies and design quality metrics.

MATERIALS AND METHODS

RS data differs on the basis of electromagnetic energy distributions. It can be obtained with the help of assortment mechanisms and diplomacy. Types of RS data may be of analogue or digital form. The basic unit of a digital image is called resolution cell/pixel (Mernhardsen, 2002). The methodological proficiencies of RS systems are classified in the form of resolutions i.e. spatial resolution, spectral resolution and temporal resolution.

Geographic Data Storage: Geographical Information System (GIS) facilitates a systematized way to store spatial data. The attributes are stored in the database and displayed on a map. GIS software is basically used to store information regarding spatial components in the form of records in a database table and the attribute components are categorized into columns (Rifaie, 2008). An important storage format is known as Raster format.

Remote Sensing Applications: Remote sensing is being used in several application areas including:

i) Geological and Mineral exploration
ii) Hazard assessment
iii) Oceanography
iv) Agriculture and forestry
v) Land degradation
vi) Environmental monitoring

Design Metrics for Software: Metrics are to be defined as a measurement unit to control well.
Computer scientists make use of the software metric method in software development (Littlefair, 2001). Software crisis is directly linked to software metrics and can be caused due to the following reasons:

- Disgustingly imprecise plan and cost estimates.
- Low output rate of the product than the demands.
- Deplorable quality software.

In order to have better and resourceful outcome by the computer applications, these problems need to be solved first. Improved use of available metrics and development of enhanced metrics emerge to be an important factor in software crisis solution. The fundamental scope of software metrics is the proper identification, measurement of the necessary parameters which influence the software development (Ebert, 2005).

Need for Software Metrics: Software development is the fundamental component in the completion of computer based systems. The System Development Life Cycle (SDLC) is a model which incorporates methods, tools and processes in the development of computer software. In the process of software development, several frameworks have been proposed, reflecting their strengths and weaknesses. Measurement is cooperative to get better, comprehend and control our software as shown Figure 2.1.

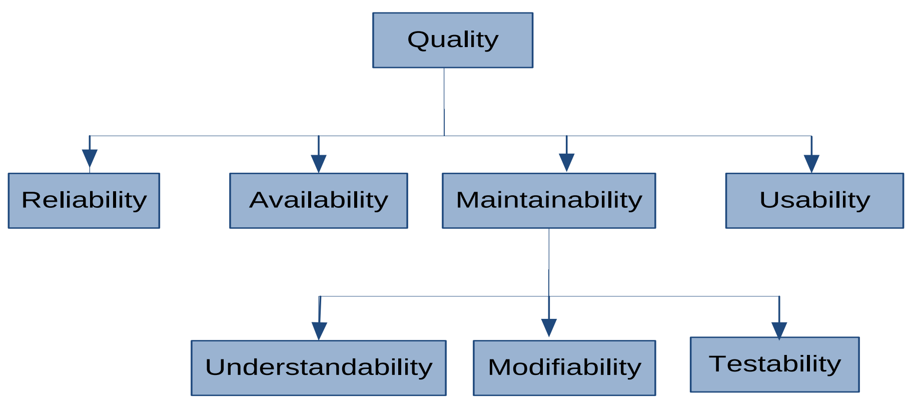

Figure 2.1: Measurement tree for software quality

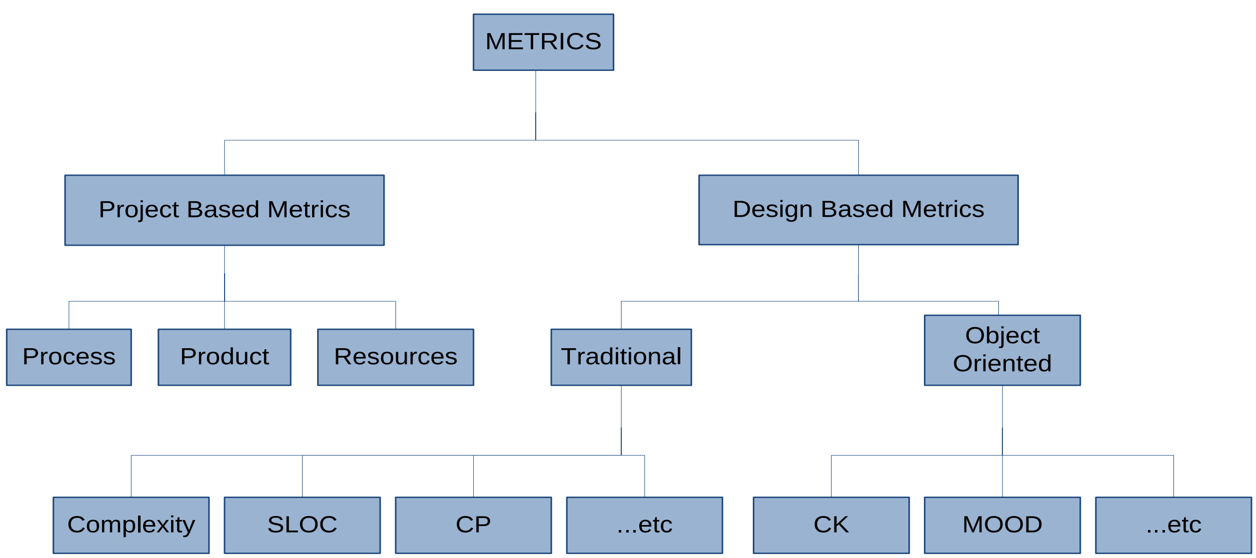

Figure 2.2: Metrics classification tree

It is being used at each phase in software engineering (software development) (Berki et al., 2004). A broad range of metrics are accessible in software engineering literature consisting of small and large software size metrics (Seffah et al., 2006). Software metrics may broadly be classified as shown in Figure 2.2 depicting following basic modules:

i) Project based metrics are used to outfit activities; analysis, design and implementation.

ii) Product metrics are used for test case in designs/codes.

iii) Resource metrics are used in ability/productivity of each tester and designer.

iv) Design_Based metrics

v) Conventional metrics

vi) Object_Oriented metrics

vii) Web_Based metrics

Proposed Design Metrics: The algorithmic complexity is specifically measured in terms of time and space. We have considered a RS application in terms of layers representing an image.

Remote Sensing System (RSS) can be defined in Equation (1), consisting images, taken at diverse time instances.

$RSS = [(I_{mi}, t_i) \mid 1 \le i \le N]$ …………… (a)

Where, $I_{mi} = \{ L_{i,j} \mid 1 \le j \le n_j \}$ ……….. (b)

Similarly, an image $I_{mi,}$ can be categorized further in the form of different layers.

$L_k = \{ L_{j,k} \mid 1 \le j \le N,\ 1 \le k \le M \}$………..(c)

We have already proposed three types (data change, structural change and combinational change) of operators for tracing change in RSS.

Let I_{mi} represents the i^{th} image in the remote sensing system and L_k represents k^{th} layer in corresponding to image I_{mi}.

We considered here two images denoted as I_{mi} and I_{mj} in RSS. These images have been taken at two diverse time_intervals [t_i and t_j] respectively. The subscript '*k'* varies *(1≤ k ≤ N). 'k'* represents the type (*forest*, *urban area*, *water body* and *soil)* of layer.

Metric for Nodes: An image contains different types of layers which are divided into number of pixels/nodes.

We can define a metric in Equation (d) to compute the number of pixels/nodes.

$NN_i = \sum_{i=1}^{K} n_i$ ---------------------------- (d)

$NN = \sum_{i=1}^{M} \sum_{j=1}^{N} n_{j,i}$ ----------------------- (e)

NN_i represents total number of nodes, in Equation (d) associated with image *i*,

Whereas, NN computes total number of pixels/nodes in an image and is given in Equation (e).

Complexity Metric for change in Nodes

$CNC = \sum_{k=1}^{n} CNC_k$ ------------------------(f)

$$CNC_k = \begin{cases} (NNt_i - NNt_j)_k, & (NNt_i > NNt_j) \text{ nodes increased in Ist image i.e. } N_1 \\ (NNt_j - NNt_i)_k, & (NNt_i < NNt_j) \text{ nodes increased in 2nd image i.e. } N_2 \\ (NNt_j = NNt_i)_k, & (NNt_i = NNt_j) \text{ No change, both images are same} \end{cases}$$

Where, NNt_i, represents number of times a loop executes to compute pixels/nodes of an image *'i'*.

NNt_j, represents number of times a loop executes to compute pixels/nodes of an image 'j*'*.

The time Complexity for algorithms is **O(p*(n+m)),**

Change in Data Complexity Metric: Data change complexity metrics is given in Equation (g).

CDC = O(N*M) ------------------------(g)

Where,

M = represents No. of times internal "loop" executes for each pixel/node of a certain layer.

N = No. of times the external "loop" executes for the maximum layers presented in two images. CDC computes the algorithmic time complexity for finding a data change in an image. The minimum value of CDC will be 1.

Complexity of Change in Data at the algorithm level quantifies the total time required to execute the algorithm.

Complexity Metric for Structural Change: Structural change complexity metric is shown in Equation (h).

CSC = O(N) ---------------------------(h)

Where, N = number of times the "loop" executes. 1 ≤ N ≤ k

CSC returns maximum value when *k* is minimum. *Complexity of Change in Structure* at the algorithm level quantifies the total time required to execute the algorithm.

Data Change Complexity Metrics in Each Layer: DCL Metric

$:DCL_k = \left(NN_{k,t_j} - NN_{k,t_i} \right)$ where , $t_j > t_i$ *and* 1 ≤ k ≤ N-----------------------(i)

$DCL = \sum_{k=1}^{n} (NN_{k,t_j} - NN_{k,t_i})$ where , $t_j > t_i$ -------------------------------(j)

Where,

NN_{k,t_j} = represents a particular layer's node, related to an image taken at time instance '$t_{j'}$

NN_{k,t_i} = represents a particular layer's node of an image taken at t_i, time instance.

The Data complexity (DC) associated to a specified layer can be computed by subtracting number of nodes in a layer '*k*' of the image at time t_j from the number of nodes of the same layer at time instance t_i. If the difference is a non_zero value then it means there occurs, a data change in the images.

RESULTS AND DISCUSSION

We have proposed a design methodology for RS applications (Ahmad and Shah, 2012). The work in this study is about the metrics for remote sensing applications to evaluate the quality of design. The mathematical formulations given in this study can be used to enhance the performance of methodology phases and four types of changes are suggested in RSA. Structural change is difficult to capture and at the same time it is hard to write its metrics because it may contain a combination of two types of changes. The proposed metrics can help to review design quality of RSA and can review in general, the performance of RSA going through implementation phase. After the design metrics assessment, we can take any important decision at the design time, before going to implementation phase. Hence, it facilitates to develop a RS application cost effective. Another important factor is a decision making by comparing two methodologies for selecting the best one to fulfill user requirements. For further improvement we need to work actively for the formalism, implementation and standardization of our proposed metrics for remote sensing applications.

REFERENCES

Ahmad, T. and A. Shah. A Methodology for Developing Remote Sensing Applications, Ph.D Thesis, University of Engineering and Technology, Lahore (2012).

Ayas, E. Engineering Quality Feelings: Applications for Products, Service, Environments and work systems, Ph.D. Thesis, Royal Institute of Technology, Sweden (2011).

Berki, E., E. Georgiadou and M. Holcombe. Requirements Engineering and Process Modeling in Software Quality Management-Towards a Generic Process Metamodel, Software Quality J. 12: 265- 283 (2004).

Ebert, C. (2005). Best Practices in Software Measurement, Springer, pages 295. Keith, H., Software maintenance and evaluation: A roadmap, ICSE '00 Proceedings of the Conference on the Future of Software Engineering, ACM, New York:73-87 (2000).

Littlefair, T., An Investigation into the use of Software Code Metrics in the Industrial Software Development Environment, Ph.D. Thesis, Faculty of Communications, Health and Science, Edith Cowan University, Mount Lawley Campus (2001).

Longley, P., F. Michael, J.M. David and W.R. David. Geographical Information Systems and Science. 2nd edition, Published by John Wiley and Sons, USA (2002).

Mernhardsen, T. Geographic Information Systems: An Introduction, 3rd Ed. John Wiley & Sons, New York, Pp, 448 (2002).

Philippe, K. The Rational Unified Process: An Introduction, 3rd Ed. Addison-Wesley Professional, Pp, 310 (2004).

Rifaie, M. Data Warehouse architecture of GIS applications, iiWAS '08 proceedings of 10th International Conference on Information Integration and Web-based applications & services, ACM, New York: Pp, 178-185 (2008).

Seffah, A., M. Donyaee, R.B. Kline and H.K. Padda. Usability measurement and metrics: A consolidated model, Software Quality J. 14: 159–178 (2006).

Stephen, H. K., Metrics And Models in Software Quality Engineering, 2nd Ed. Pearson Education India, Pp, 556 (2003).

Zlatanova, S. and S. Nayak. Remote Sensing and GIS Technologies for Monitoring and prediction of Disasters, Int. J of Digital Earth, 5(1): 256-270 (2012).