

Development of Customized Tools for GIS Modelling to Incorporate Flexible Decision Making

R.M.M. Pradeep and N.T.S. Wijesekera

Abstract: Geographic Information Systems (GIS) and associated modeling enables rational management of spatially distributed resources. GIS models once assembled with a base data set often require many repeated computations either to seek modifications to the methods or assumptions made during computations. The common GIS software needs customization to cater to such changes with minimum requirement of time. These repeated tasks range from computing an area to reclassification of overlaid layers.

This paper presents the development of two simple tools. The tool for computation of area, centroid or length, enables the on-screen operation to save computed values in particular shape file. Tool developed for the classification of spatial information after overlay, demonstrates the tool's capability to easily change the classification criteria to suit a change of values or grouping based on a subsequent decision maker requirement.

Both tools are developed using the Microsoft Visual Basic and linked to the GIS software - ArcGIS 9.2, incorporating the systematic planning, design and evaluation methodology. A special consideration was paid to 100% accuracy of the outputs, usefulness and user friendliness of the tools. Hence the work follows the standard requirement assessment, systematic software evaluation with appropriate user group selection. The tools demonstrate the capability through applications on actual datasets.

Keywords: GIS Tool, Model Building Process, Vector Reclassification, Area Centroid calculation

1. Introduction

Geographic Information Systems are tools that enable rational management of spatially distributed resources. This means that often a GIS enables interaction with either one or many map layers. In the decision making process it is common that the basedata leading to several rational alternatives are prepared [24], at several management positions in an organizational hierarchy. Under these circumstances, a data model developed by a GIS expert following an agreed concept would require repeated computations or repeated decision modifications. Though repeated computations are faster when working with computers, the speed of an activity often depends on number of tools to be used, number of working windows, number of clicks required for a particular operation, menu selection sequence, reaction time of a person etc. Therefore, when working with GIS databases repeated computations with changes to a model concept would consume a considerable time [14]. Significant time consumption for repetitive work tends to affect the quality of outputs since there could be a lethargy to execute repetitive work specially

when such works require significant time and effort.

The strength of GIS is to provide or facilitate alternative options with ease. As such it is necessary to identify repetitive works which normally consume a significant time, and then find ways of reducing time delays by optimising the intermittent operations.

One of the common tasks that require continuous computation is the calculation of either the area, perimeter or centroid of a polygon. In most of the off-the-shelf GIS software such as ArcGIS, the area of a polygon, perimeter or centroid can be calculated by incorporating scripts to run on a built-in computation tool. This operation has to be repeated from map layer to map layer or whenever calculations are required. ArcGIS

R.M.M. Pradeep B.Sc. (Hons) in MIS [NUI], Geographic Information for Strategic Defence Applications, Centre for Research & Development, Ministry of Public Administration, Defence, Law and Order, Army Cantonment, Panagoda, Homagama, Sri Lanka

Eng. (Prof) N.T. Sohan Wijesekera B.Sc.Eng. Hons.(Sri Lanka), PG.Dip (Moratuwa), M.Eng.(Tokyo), D.Eng. (Tokyo), MICE(UK), FIE(SL), Chartered Civil Engineer, Department of Civil Engineering, University of Moratuwa, Katubedda, Moratuwa, Sri Lanka.

software provides a measurement tool with an interface which enables drawing of a polygon to compute area, perimeter etc. This tool eases the computations to a reasonable extent, but does not facilitate quick and accurate information due to the reason that a user has to carryout mouse positioning and clicking along the entire drawn polygon and then save to an attribute table or to an appropriate software. This also does not often provide accurate results because digitizing over the existing nodes would encounter certain accuracy issues. Therefore, it is often found that a tool capable of carrying out simple computations on an already drawn polygon and then saving of the data would be of significant benefit for GIS users who attempt to carryout these calculations many times on various features and involving multiple layers.

In Literature there are several automated options, such as "Easy Calculator 5", "ET Geo Wizard" [6], "Tools for Graphics and Shapes" [19] for such calculations. However, the fundamental issue with these tools is unavailability of on-screen outputs.

Another common need of GIS users is the speedy reclassification of attribute table datasets after an overlay operation. This work consumes a significant time because a reclassification always requires many conditional table operations necessitating calculations on a selected set of features. Time consumptions for reclassification usually become excessive when decision makers tend to make changes to the model concept or parameters. At most instances GIS experts feel the importance of having a flexible lookup table for reclassification. An easy lookup table modification which can be directly fed into a GIS overlay operation for the generation of a reclassified layer would provide a definite advantage to the GIS modeller. Since the objective is to reduce time taken to effect decision changes with respect to reclassification criteria, development of a tool to carryout such tasks would be very effective if on-screen changes could be directly incorporated to a ArcGIS map layer.

A literature search did not reveal the development of a similar tool except for identifying the possibility of using macros. [9,20,21]. The present work describes the needs assessment, tool development, tool testing and

application, with regard to two tools developed for:

- (i) Simple computations such as area, perimeter and centre of gravity; and
- (ii) to carryout easy reclassification of a map overlay output.

2. Objectives

The objective of this work is to develop two GIS tools incorporating user friendliness and saving of user efforts for computations:

- (i). Simple Computational Tool (PolyCal) - A GIS Graphical User Interface to calculate Area, Perimeter and Centroid of a polygon, and the length of a line or a polyline; and
- (ii). Reclassification Tool (Classiφ) - A GIS Graphical User Interface to carryout reclassification of data layers resulting from a map overlay operation.

3. Methodology

3.1 General

Rapid Application Development (RAD) method of software tool development which is a variation of the prototyping models was selected for the work due to advantages of obtaining early user feedbacks and capability to adopt to rapid changes [3,15]. Overall Approach taken to develop the tools is shown in Figure 1.0.

3.2 Interface Design

Since the most important aspects of User Interface (UI) development are the ease of learning and ease of use [18], guideline directions were followed to ensure an iterative design process, Visual clarity, consistency, compatibility, informative feedback, explicitness, appropriate functionality, flexibility and control, error prevention and correction, and user guidance and support with respect to both tools. [10,16,17,18].

3.3 Requirement Assessment

A requirement assessment through a simple questionnaire survey of 71 GIS users, who belong to various organisations, identified needs of GIS users. The questionnaire assessed;

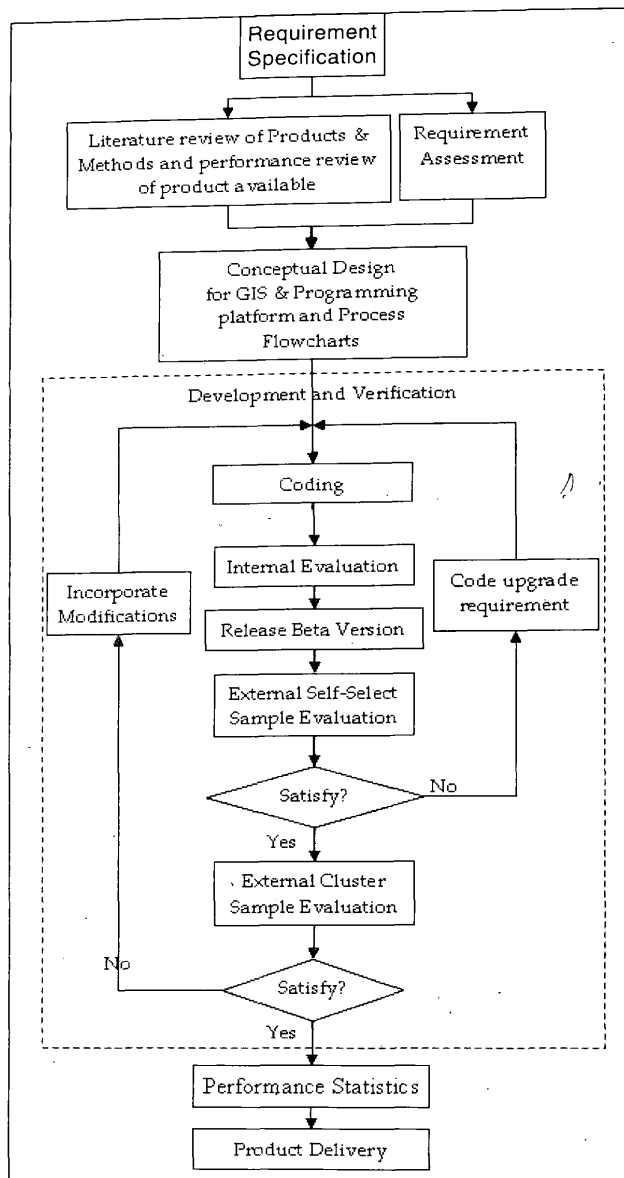


Figure 1 : RAD methodology used for GUI Development

(i) The level of satisfaction with respect to presently available tools and expectations corresponding to a new tool, that would enable simple computations, and (ii) The desired method of operation for the tool for reclassification.

Only 18% found the present methods on ArcGIS as either very easy or easy while 82% found as either satisfactory, difficult or very difficult. 71% indicated that not only the computed values should have the capability of saving data in attribute file, it should also permit the attachment of values to the clipboard. 64% of the sample preferred the Simple Computation Tool to be incorporated as a button on the standard ArcGIS tool bar. With regards to the Reclassification Tool, 72% opted for a similar button enabling an on-screen Graphic User

Interface (GUI) so that adjustments to classification/ reclassification parameters could be made.

In Sri Lanka the most widely used GIS software is ArcGIS [22]. A worldwide survey done in 2000 [1] had indicated that ESRI with 23%, has the greatest market share by a single GIS software vendor. Daratech 2001[2] had indicated that ESRI has increased this share to 35%. Therefore the tool development efforts were carried out for the ArcGIS software because of the greater number of users that would be benefited by the tool. Due to similar reasons and also since ArcGIS facilitates Visual Basic (VB) as the development language, which has also been indicated as user friendly [7]. Interface development was carried out using Visual Basic Language.

GUI was initially developed using Visual Basic for Applications (VBA) and with ArcGIS to use advantages of easy bug fixing. At the final stages of tool development, the tool was migrated from ArcGIS environment to VB. This proved easy for the finalization of tool as a DLL. A DLL was identified as a better option for transferability and flexible installation [25].

3.4 Simple Computational Tool - PolyCal

3.4.1 Conceptual Design and Process Flow

Conceptual design included a simple process (Figure 2.0) indicating the ability to save computed feature information.

Process flowchart for VB coding is shown in Figure 3.0. The tool permits the selection of either a polygon, polyline or a line and based on selected decision criteria incorporates computational procedures. Tool coding was limited to the development of coding components other than the basic VBA procedures in the ArcGIS help file.

3.4.2 Development and Verification

PolyCal Tool was developed as a button on ArcGIS toolbar that has an input component of identifying the required features on the topmost ArcGIS layer. Coding for this component was written in VB using sample scripts as initial guidelines [4,5,8,11]. In order to save time, coding efforts opted out the use of an explicit

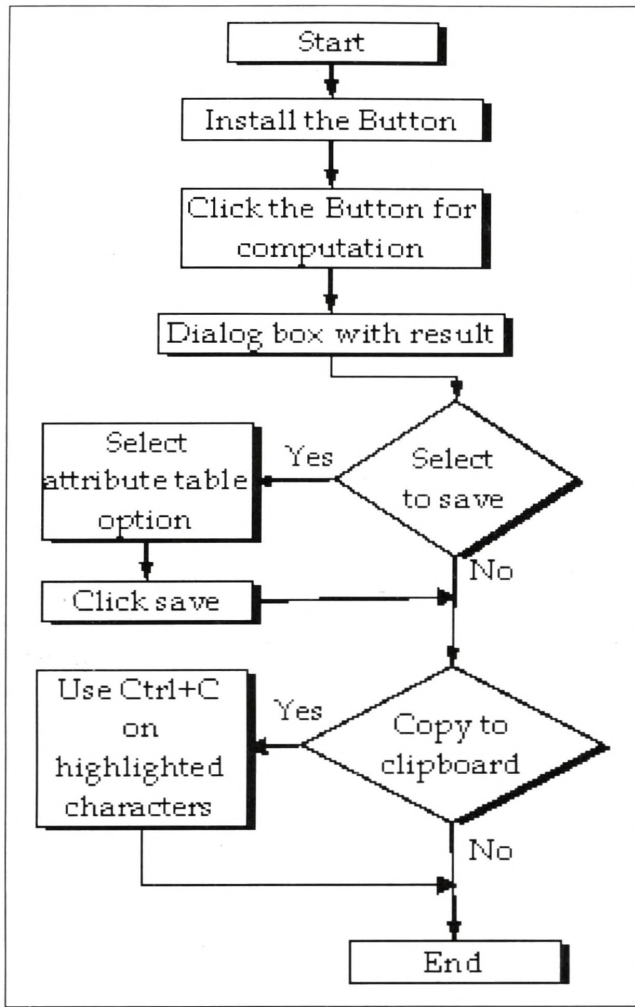


Figure 2 : The Conceptual Design of PolyCal

message action to initiate processing of data. A feedback message was incorporated to indicate satisfactory data capture. Coding for computations utilized the VB coding of individual components presented in the ArcGIS help file and adjustments were suitably incorporated to cater to multiple features. Coding was completed using VBA.

Output form design to suit user requirements was based on a consistent label format, standard font and clarity of wording. Simple texture formats were used for visual clarity. Colour selection coding was restricted to MS Windows standard colours enabling user friendliness. Output form was coded in such a way to avoid the modes and to be positioned as the topmost layer to ensuring better user control, while enabling simultaneous ArcGIS layer operations without disturbance. A logical organization of fields in output form facilitated the extraction of output data, selection of save location, and execution of save to be carried out in a sequence

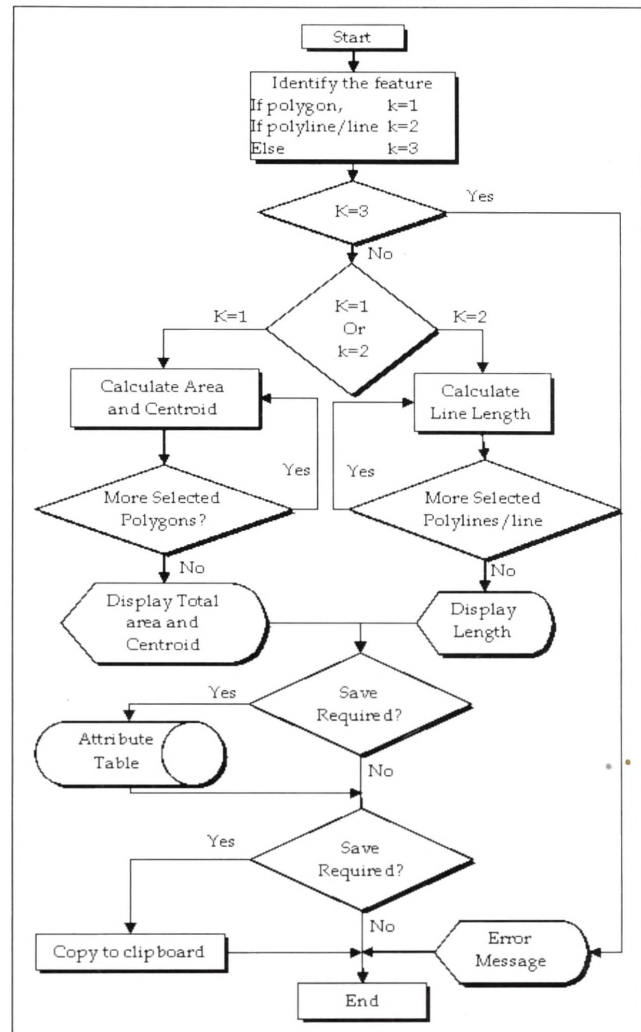


Figure 3 : Process flowchart for Visual Basic coding - PolyCal

from left to right. Computational coding for Output Form included only those for the change of units to user identified units and then to change the display presentation accordingly. A sample of output form is shown in Figure 4.0

Since the tool was mostly based on user requirements and it encompasses the end-user

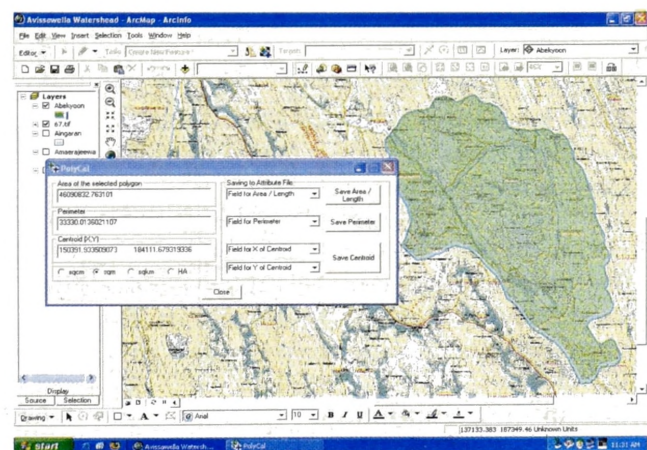


Figure 4 - Output from of Polycal

perspectives, a black-box testing methodology was adopted [3] through a beta-version. Testing of the tool was carried out with three evaluations (Figure 1.0). Internal evaluations included tool evaluation for the process and accuracy of computations, and also for the easy operation capability of GUI. Accuracy evaluation was done for the feature selection accuracy, and relevant computation accuracy ensuring values acceptable for standard user applications. As part of result verification process computational accuracies of the tool were compared with manual computation of regular geometric shapes. Manual computations with five types of geometric shapes varying in size revealed 100% accuracy with tool

Table 1. - Version Testing results of PolyCal

PolyCal Tool			
Beta Version	Errors per user	Catastrophes per user	User acceptance
1.1	2.79	0.57	28%
1.2	1.57	0.14	50%
1.3	0.29	0.00	92%

Table 2. - Version Testing results of Classiφ

Classiφ Tool			
Beta Version	Errors per user	Catastrophes per user	User acceptance
1.1	6.50	0.71	21%
1.2	3.71	0.43	29%
1.3	2.43	0.36	42%
1.4	1.29	0.21	71%
1.5	0.21	0.00	100.00%

Table 3.0 User Acceptance of PolyCal

PolyCal Tool's Functional Usability measure	Aggregate Acceptance score	Mode Acceptance score	Rank
Time reduction	138	5	1
Accuracy Status	136	5	2
Usefulness & Acceptance to work	124	4	3
Easy to learn	123	4	4
Security against user error	121	4	5
Easy to understand of operations	121	4	6
GUI Status	118	4	7
Easy method of operation	116	4	8
Blends well with GIS platform	113	4	9

computations.

At the second level of evaluation, a self selected sample [12] of GIS users was used. Observations were made on the number of errors, catastrophes and user acceptance [16], while each user was applying the tool for his/her own datasets. Evaluation was considered satisfactory once the catastrophe level reached zero. Summary of testing results after three beta versions is shown in Table 1.0.

Final testing was based on a questionnaire survey to identify user response on various aspects such as GUI status, acceptance, security, speed of work, friendliness etc. As it was difficult and costly to evaluate the product using a geographically dispersed random sample [12,23], the Cluster Sample Method was used with 30 ArcGIS users. The evaluation questionnaire development was based on the framework for system evaluation [13], Usability aspects [16], and common threads of Usability testing [2]. The Likert Scale [23] with a 1-5 qualitative classification representing Strongly Disagree, Disagree, No Opinion, Agree and Strongly Agree, was used for assessment. The responses of users were aggregated using rank reciprocal method (Table 3.0).

3.5 Reclassification Tool - Classiφ

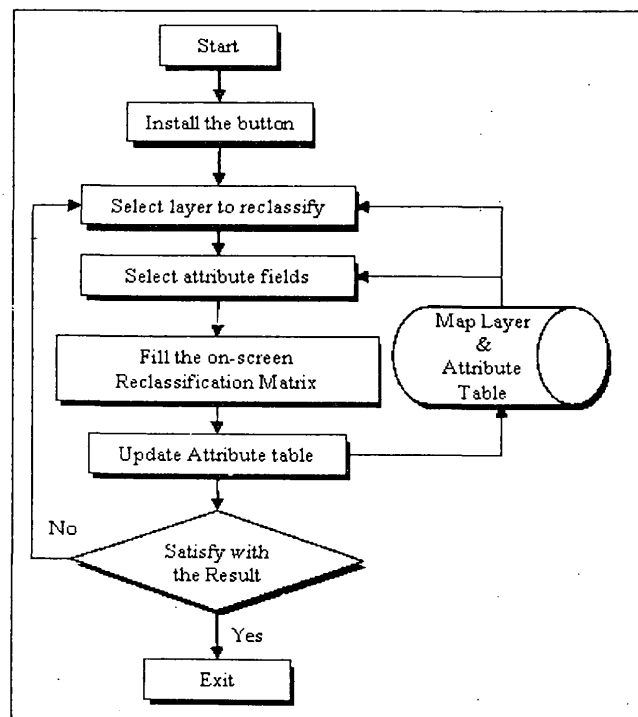


Figure 5 : Conceptual design - Classiφ

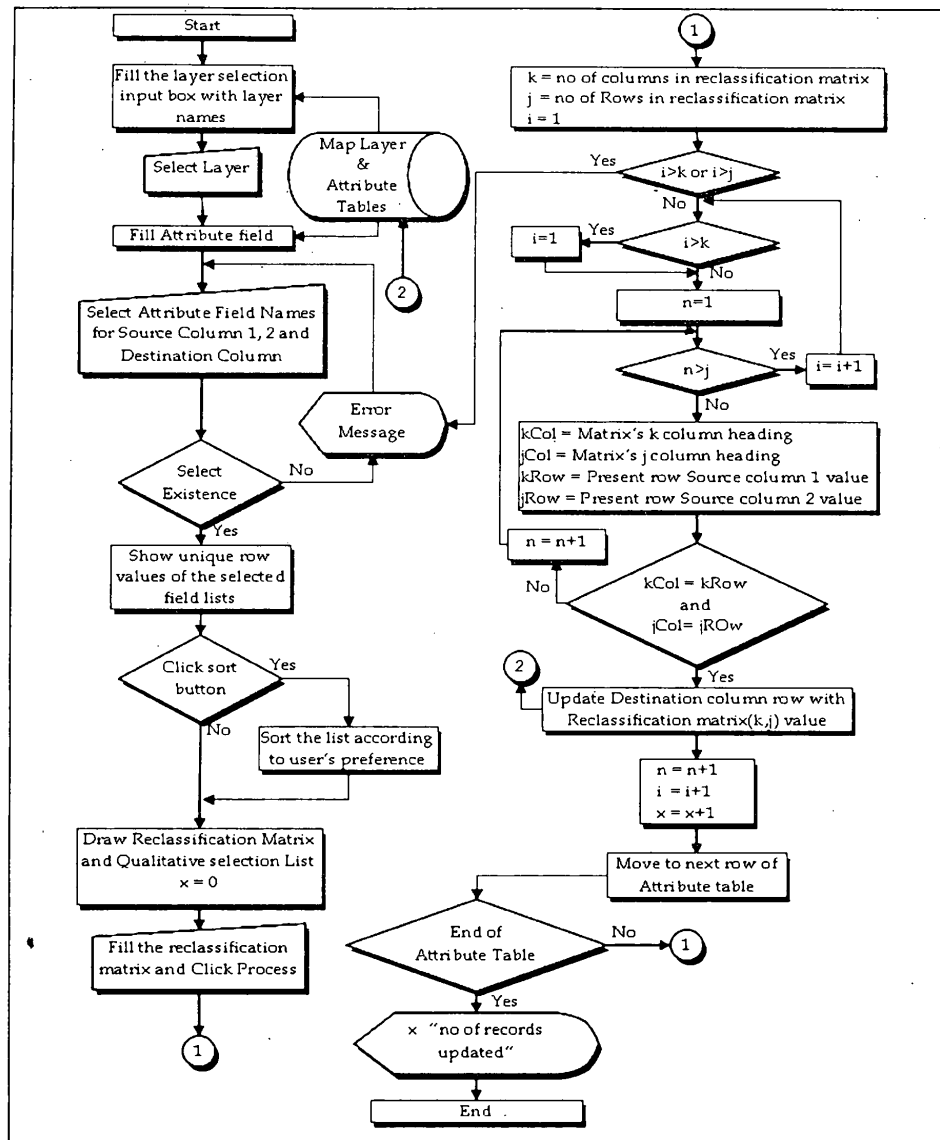


Figure 6.0 Process flowchart for Visual Basic Coding – Classiφ

3.5.1 Conceptual Design and Process Flow

The conceptual design included a process flow as shown in Figure 5.0. Flowchart for VB coding is shown in Figure 6.0. Tool permits the selection of two source columns from a given map layer for the development of reclassification criteria and then the creation or selection of a destination column to save the reclassified value. Further, the tool draws a graphical representation of reclassification matrix utilizing the selected source columns, whilst allowing on-screen filling of particular reclassification values in the matrix. Classiφ ends its task with the updating of attribute table by the values according to decisions indicated on the reclassification matrix. This work involved a significant coding effort other than already available VB/VBA procedures with ArcGIS VB Desktop help files.

3.5.2 Development and Verification

The Reclassification tool Classiφ was also developed as a button on ArcGIS toolbar and the coding procedures were followed similar to the previously described tool. However when coding, special care was taken in the development of GUI, since the tool is intended to facilitate a graphical coordination of a logical process which consists comparing and updating of data fields and field combinations.

Therefore during tool development, the usability principles, desktop integration, Windows, controls, feedbacks, Visual Design and Language were given significant emphasis [10]. A sample of input forms is shown in Figure 7.a and 7.b.

A GUI consisting of two tabs where the Tab 1 prompts a user to select classification layer, output field and layer priority coding. In Tab 2,



these details are graphically arranged to depict a two dimensional matrix operation. The graphical matrix operation facilitates the user

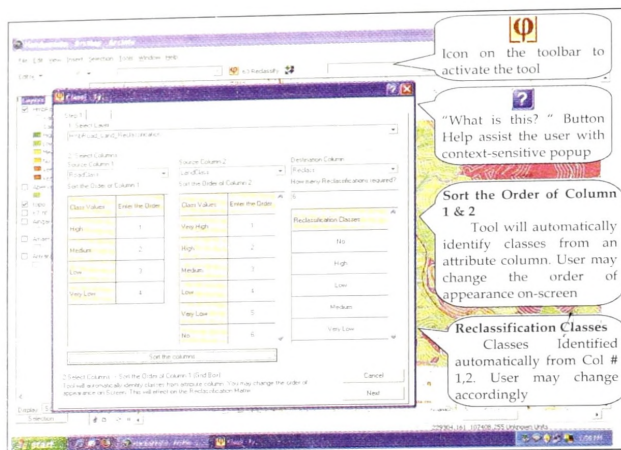


Figure 7.a - Input Form 1:
Attribute Column Selection - Step 1

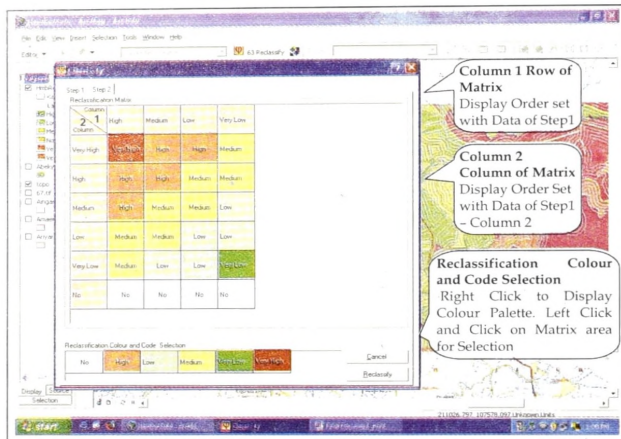


Figure 7.b Input Form 2: Selection of criteria for
Reclassification Matrix - Step 2

with a visual picture to rationalize the selection of reclassification and then to store output data in a designated column through an explicit action button to initiate reclassification.

Testing of the tool was carried out similar to the previous tool and with same user groups. Computational accuracy of the tool compared with a manual operation using ArcGIS menu functions, revealed 100% accuracy of tool computations.

In case of Beta-Version evaluations with self selected sample achieved a zero catastrophe level at the fifth beta version (Table 2.0).

Final testing was based on a questionnaire survey to identify user evaluation of various aspects such as GUI Status, acceptance, security, speed of work, user friendliness etc. Results pertaining to an assessment on a 1-5 class Likert

Scale are in Table 4.0.

4. Performance

4.1 Time and Clicks

Table 4.0 - User Acceptance of Classi_

Classi_ Tool's Functional Usability measure	Aggregate Acceptance score	Mode \ Acceptance score	Rank
Time reduction	156	5	1
Accuracy Status	125	5	2
Blends well with GIS platform	120	4	3
Easy method of operation	119	4	4
GUI Status	118	4	5
Easy to understand of operations	116	4	6
Security against user error	115	4	7
Easy to learn	111	4	8
Usefulness & Acceptance to work	110	4	9

Evaluated products were further assessed for performance with the comparison of time and a click count with common methods. The simple computational tool was named PolyCal due to its capability to calculate parameters pertaining to both polygons and Polyline. The reclassification tool was named Classi, considering the brevity of name and also its function. PolyCal was compared with computations done using Field Calculator tool and Calculate Geometry tool of ArcGIS. Classi was compared with; (i) Straight forward attribute selection and update by field calculator method (Method 1) and (ii) The method of computing with the use of an ArcGIS Macro (Method 2).

Both tools showed very satisfactory advantages with respect to time and effort for task execution (Table 5.0, 6.0, 7.0)

4.2 Application

PolyCal and Classiφ were both tested on two applications using actual geographic data. Classiφ was used to classify two data layers of Hambantota District to arrive at a layer of combined preference considering performances corresponding to Landuse and Proximity to

Table 5 - PolyCal Tool time comparison

Tool	Time Taken (s)
Filed Calculator	55
Calculate Geometry	45
PolyCal Tool	2

Table 6.0 PolyCal Tool's click comparison

Action		No of Clicks
Field calculator	Select Feature	2
	Open Attribute table	2
	Open Field Calculator	2
	Calculate Area	5
	Show Area	1
	Total	12
Geometry Calculator	Select Feature	2
	Open Attribute table	2
	Open Geometry Calculator	2
	Calculate Area	3 to 5
	Show Area	1
	Total	10
PolyCal	Select Feature	2
	Show Area	1
	Total	3

Table 7.0 Performance Comparison of Classiφ

Used Method	No of users	Average Time (min)
Method 1	8	35
Method 2	3	15
Classiφ	11	5
Time Saving than Method 2		66%

roads (Figure 8.0). PolyCal was used to compare the watersheds drawn by students with a master data set. Student drawn watersheds were in separate shape files the results taken to a single attribute table as shown in Figure 9.0.

5. Results

Self selected group evaluation of PolyCal and Classiφ are shown in Table 1.0 and 2.0 respectively. Cluster group evaluations of the tools are shown in Tables 3.0 and 4.0. Results of the performance check of PolyCal with time consumption for tool execution and clicks required are shown in the Table 5.0 and 6.0. Table 7.0 shows the performance comparison of Classiφ.

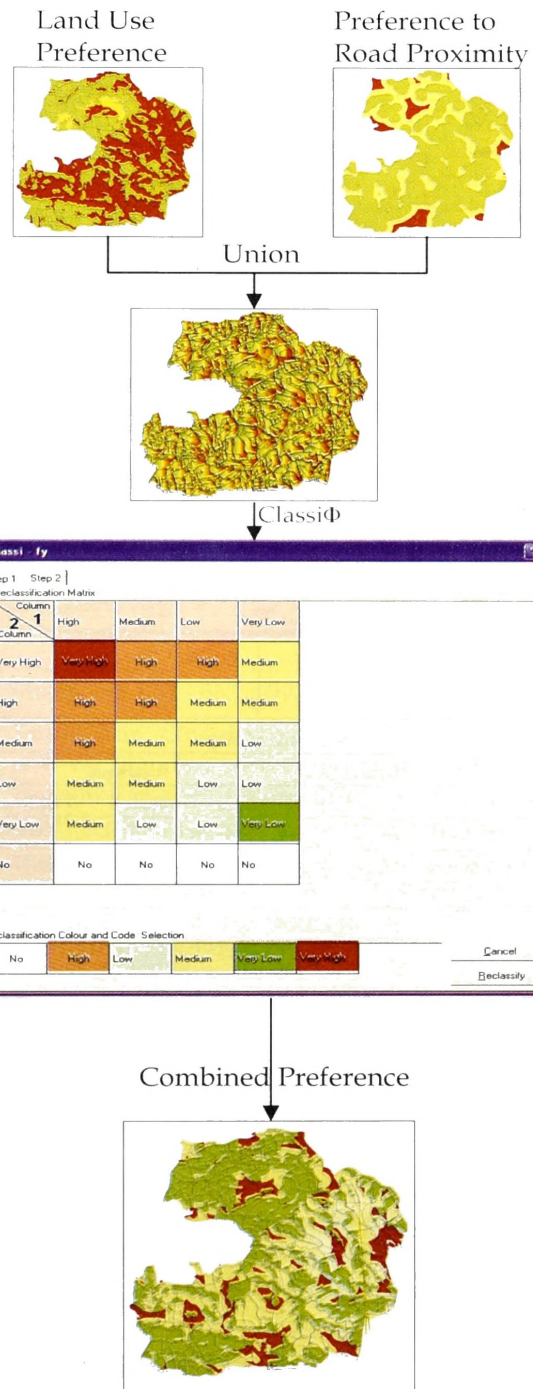


Figure 8.0 - Reclassification of Hambantota Layers - Combined Preference

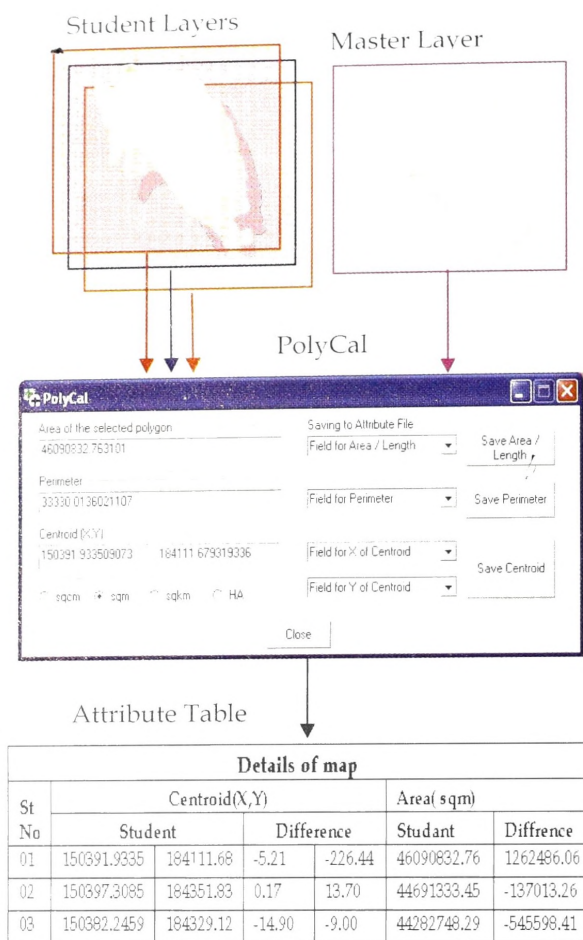


Figure 9.0 - Student Watershed calculations

6. Discussion

1. Literature and user requirement assessment surveys indicated that though there are many tools that are for the ArcGIS platform, general tools for vector layer operations are limited and hence requires the development of tools that can be customised. User survey clearly indicated a general liking to work within the ArcGIS platform and to possess decision making capability through GUI. User survey indicated the need for tools that would save time and effort without compromising accuracy.
2. The tools, systematically developed through a rigorous methodology, ensured that the major concerns such as effective use of standard operations, system functions pertaining to colour selection, copying or mouse clicks have been incorporated to achieve user consistency. Functionalities such as, enabling an ArcGIS user to easily move from computing window of the tool to project layers, were specially targeting user friendliness. GUI of Classiφ enabled a user

to identify feature names and attributes automatically extracted from the table thereby eliminating computational errors due to erroneous fields or attribute naming.

3. Tool development evaluations indicated that the use of simple tools could reduce the processing speed significantly while ensuring the user friendliness. Time reduction and click reduction of the PolyCal was approximately 96% and 70% respectively. The Classiφ showed that a time saving of about 66% would be achieved during a single reclassification operation when compared with the use of a macro. This indicated a clear advantage of having tools developed for general uses such as those carried out by PolyCal and Classiφ.
4. The developed tools indicated a part of their potential through the use of actual case study data thereby enhancing the user confidence. The evaluation of software using standard methods, appropriate user samples and acceptable evaluation indicators have enabled the development of the tool ensuring user requirements. The PolyCal and Classiφ both were finalised at zero catastrophe level. However, Classiφ had a 100% user acceptance while PolyCal had 92%. This reflects that the user sample fully accepts Classiφ for usefulness but does not accept usefulness of PolyCal for their work. Since this can occur due to a bias in the sample, and also that the catastrophe level was 0, the tool development was concluded.

7. Conclusions

1. The development of two ArcGIS tools was achieved by incorporating systematic planning, design and evaluation methodology.
2. Both tools PolyCal and Classiφ showed excellent results in relation to accuracy, usefulness and user friendliness.
3. In the development of both tools, the requirement assessment, systematic evaluation with appropriate user group selection, followed by a demonstration with actual spatial data, proved as very important factors for successful achievements.
4. PolyCal and Classiφ tools both showed a time saving potential of greater than 66%.

Acknowledgement

The authors wish to extend sincere appreciation of the encouragement given by the management and the staff of Ministry of Defence, and the Centre for Research and Development to publish this work. Support of the ICGAT of University of Moratuwa and its staff is gratefully acknowledged.

References

1. Cahan, B., *Financing the NSDI: National Spatial Data Infrastructure*, Urban Logic, 2000, p. 31.
2. Daratech 2001 GIS Software Revenue Estimates Released, http://www.gismonitor.com/news/pr/110702_Daratech.php, Visited, 1st March 2008.
3. Désikan, S., Ramesh, G., *Software Testing-Principles and Practices*, 4th Imp., Dorling Kindersley, India, 2007, pp 34,73,275.
4. Documentation for developers using ArcGIS 9.2, <http://edndoc.esri.com/arcobjects/9.2/welcome.htm>, visited, 20th February 2008.
5. ESRI, Getting started with VBA- ArcGIS desktop Help- ArcGIS 9.2, 2007.
6. ET SpatialTechniques - ArcView and ArcGIS extensions, <http://www.ian-ko.com>, visited, 10th May 2008.
7. Gemitzi, A., Tolikas, D., "HYDRA model: Simulation of Salt intrusion in coastal aquifers using Visual Basic and GIS", *Journal of Environmental Modelling & Software* 22 (2007), Elsevier, pp. 924-936, 2006.
8. Getting Started with EDN- For Developers by Developers, <http://edn.esri.com>, Visited, 2nd April 2008.
9. GIS Applications in Community Telecommunications, http://top.bev.net/archive/tamp/7-Common_Appendices/Main_Project_Papers/GIS_Applications_in_Community_Telecommunications.pdf, visited, 8th May 2008.
10. GNOME Human Interface Guidelines 2.0, <http://developer.gnome.org/projects/gup/hig/2.0/hig-2.0.pdf>, Visited, 23rd January 2008.
11. Grasteit, C., Customizing ArcGIS or This is not as difficult as you think, <http://justspatial.com/sfbagisusers/Customizing Arc GISVBA and Samples.pdf>, Visited, 3rd May 2008.
12. Keller G, Warrack B, *Statistics for Management and Economics*, Thomson Learning, 6th Ed, 2003, pp 143-148.
13. Marakas, M.G., *Decision Support Systems*, Prentice-Hall Press, New Jersey. 1999, pp. 420-423.
14. Michener W.K., Brunt J.W., Stafford G.S., *Environmental Information Management and Analysis: ecosystem to Global Scales*, Taylor & Francis Ltd, 1994, pp 113-125.
15. Ngan, S., Dennerlein, J., RAD and ESRI desktop solution: a new horizon on GIS Application Development, <http://gis.esri.com/library/userconf/proc97/proc97/to650/pap634/p634.htm>, Visited, 5th May 2008.
16. Nielsen, J., "Iterative User Interface Design", *IEEE Computer*, Vol 26, No. 11, pp. 32-41, 1993.
17. Rinner, C., Raubal, M., Spigel, B., User Interface Design for Location-Based Decision Services, http://ifgiweb.uni-uenster.de/~raubal/Publications/RefConferences/paper-geoinfo2005_final.pdf, Visited, 4th June 2008
18. Sullivan, K., "The Windows 95 User Interface: Iterative Design and Problem Tracking in Action", *International Conference on Software Engineering - Proceedings of the 19th international conference on Software engineering*, Boston, Massachusetts, United States, pp. 562 - 563, 1997.
19. Tools for Graphics and Shapes, http://www.jennessent.com/download/Graphics_Shapes_Online.pdf, Visited, 5th May 2008.
20. View and edit attributes, http://folk.uio.no/maalfrih/Lab3_extra.pdf, visited, 8th May 2008.
21. Web-based Help Overview, <http://webhelp.esri.com>, visited, 5th May 2008.
22. Wijesekara, N.T.S., "Spatial Resource Planning and Management Applications in RS/GIS and present day needs, Proceedings of the Workshop on Remote Sensing/GIS for Environmental Monitoring and Spatial Resource Management", Arthur C Clerk Center for Space Technology, Katubedda, Moratuwa, October 2005
23. Willie, T., *Practical Research Methods*, 2nd Ed., Prince hall, 2004, pp 87-96, 140
24. Wu, W., Shibasaki, R., Yang, P., Tan, Guoxin., Matsumura, K., Sugimoto, K., "Global-scale modelling of future changes in sown areas of major crops", *Journal of Ecological Modelling* 208, Elsevier, pp. 378-390, 2007.
25. Zeng, H., Talkkari, A., Peltola, H., Kellomaki, S., "A GIS-based decision support system for risk assessment of wind damage in forest management", *Journal of Environmental Modelling & Software* 22 (2007), Elsevier, pp. 1240 - 1249, 2006

