GUI for SLEUTH model

Kamakshi Moparthi[1][1], M. Vani[2], P. Rama Chandra Prasad[3]

[1]Department of Computer Science, Mahindra Ecole Centrale College of Engineering, Hyderabad, 500043, India
[2,3]Lab for Spatial Informatics, International Institute of Information Technology, Hyderabad, 500043, India.

Keywords: AWT,GUI, JFrames, SLEUTH, Swings.

Abstract

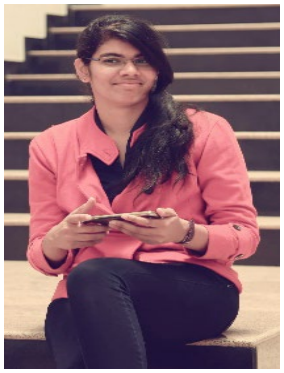
Urban Modelling is a process in which a simulated theory is converted into a mathematical model and develops a computer program such that the input data given to the model is tested, calibrated and validated before prediction. There are many urban models used for prediction. One of the well-known cellular automata based urban model developed by Keith Clarke is the SLEUTH model. This model is a probabilistic cellular automata model with Boolean logic. SLEUTH model consists of two sub-models Urban growth model (UGM) and Deltatron Land Use / Land Cover model (DLM) to simulate the spread of urbanization. SLEUTH (**S**lope,**L**and-use , **E**xclusion layer, **U**rban Extent , **T**ransport , **H**illshade) model as the name indicates has six input parameters. In this model the user has to create six different scenario files for each mode. However, for the non-coders it appears technically difficult to use this model. Existence of a GUI could simplify the work of the user. Towards this the objective of the current research is to create a GUI and publish it as an open tool for users interested in urban growth modelling. The GUI is developed in Java, using Swings , although it is built on AWT (Abstract Window Toolkit) , it provides powerful and flexible GUI components compared to AWT package . In the code to develop GUI , javax.swing package is imported. Generally a swing GUI has two items i.e containers and components, for example when a window is created it has different items like buttons , labels , scroll bar etc. all such items are called components and to hold all such components a container is used. This GUI is created specifically to execute SLEUTH model to obtain the outputs. When the application is opened the first window appears .This is created using JFrame ,JButton , JLabel Classes . This window has a label which displays the text “Interface for executing SLEUTH model ”, on the lower right corner there is a button name “NEXT” , when the user clicks on the button the first window closes and a new window appears . The second window has a layout similar to that of the first window and has a title “Inputs”. All the inputs should be in .gif format for successful compilation of the model. The model requires three different input folders one folder contains the set of images with maximum resolution, the second folder contains all the images which have half the resolution and the third folder has the set of images with one-fourth the maximum resolution ,for example if the maximum resolution is 200x200 , other two folders will have images of resolution 100x100 and 50x50.This window has a label “Input directory”, with a button “Browse”, when the button is clicked the user has to browse for the location of the input folder and has to click on the “NEXT”button. This step will create immediately ‘Output’ folder and ‘Scenario’ folder in the same location where the Input folder is present . In the output folder four folders are created , to store the outputs produced in each mode i.e the outputs produced during testing are stored in the corresponding folder. In the Scenario folder five files are created in the .txt format each file is used to store the data . Immediately after the file creation the details about the input directories and output directories are written to all the files.

The third window appears after the file creation , and has the title “Independent Data” , all the details which are independent of the mode under execution are entered here for example the colour used to

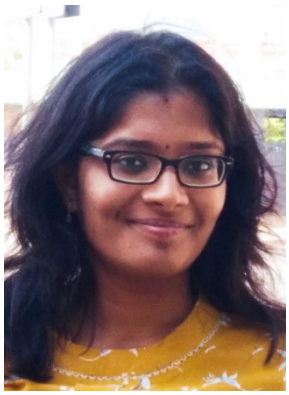
[1] Corresponding Author: Kamakshi Moparthi
ORCID:
Email address: kmoparthi31@gmail.com
DOI:

represent water bodies is independent of mode under execution , There are different labels(seed colour , water colour , landuse details) and their corresponding text fields .When the user enters all the details and clicks on ”NEXT” button this data is written to all the scenario files and the fourth window appears . This window has a title “Data for testing mode”, in this window there are labels and text fields where the user enters the data specific to the testing mode, similarly three other windows are created to receive the data specific to calibration, validation and prediction mode. As soon as the user clicks the “NEXT” button on the seventh window, all the data is written to their corresponding scenario files (in a particular format) and then a new window is opened .The code is then compiled and executed. After compiling certain executable objects are created , each mode is executed separately using the command./<object name><mode><scenario file path> on the command line. So all the commands which are to be entered on the terminal are stored in the array of String and they are executed using getRuntime().exec() ,the command which is to be executed is sent as a parameter to the exec(). After the successful compilation the outputs are created and are stored in their respective Output folders. The eight window is created after the execution of the model and has a title “Output”. It has three button in the window (“testing”,”prediction”, “validation”), when the user clicks on one of the buttons a new window appears which shows all the output images. This helps the user to visualize the outputs by switching between different outputs and they are available in .gif format for exporting into other compatible software.

1. Author/s Biography



Kamakshi Moparthi is a B.Tech. Computer Science Student at Mahindra Ecole Centrale, India. She completed her internship at Intenational Institute of Information Technology, Hyderabad, India during her B.Tech. programme. She is a coding enthusiast and is competent in Java, Python, C, and C++. She also likes to code for real time problems.



Mrs. M. Vani is a budding researcher in the field of Urban Studies and Climate Change. She is a Civil Engineering postgraduate (2014) from Anna University, Chennai and an alumnus of College of Engineering, Guindy, Tamil Nadu, India. With a year teaching experience (2014-2015) as Assistant Professor in the Department of Civil Engineering in Vardhaman College of Engineering, Hyderabad, India, Vani currently pursues her Ph.D. in Civil Engineering at International Institute of Information Technology-Hyderabad, India since 2016. She was also awarded the Senior Research Fellowship (SRF) under SRF-Direct scheme offered jointly by the Council of Scientific and Industrial Research (CSIR) and Ministry of Human Resource Development Group (MHRDG), Government of India (GoI) during 2018. Vani has presented and participated in different International and National conferences, Workshops and Training programmes. During the course of her Ph.D. programme, she was also awarded the Best paper award for her contribution in FOSS4G-Asia 2017 jointly organised by the OSGeo-India and IIIT-Hyderabad, India.

R. C. P. Pillutla works as assistant professor at Lab for Spatial Informatics, IIIT-Hyderabad. He has completed his Ph.D. in the Environmental Sciences under joint affiliation of National Remote Sensing Centre (NRSC-ISRO), and Department of Environmental Sciences, Osmania University, Hyderabad. His research interest includes management of natural resources (primarily forest) using geospatial tools and promoting awareness towards the utility of open source geospatial software.