CHAPTER 3

Digitization
3.1 Introduction

Digitization is the transformation of information from analog format, such as a paper map, to digital format, so that it can be stored and displayed with a computer. Digitization results the vector map where the necessary data for the vector object can be attached. Digitization can be manual, semi-automated (automatically recorded while manually following a line), or fully automated.

Different types of digitizing process are:

- **Manual Digitization:** It is done using digitizing tablet. The operator manually traces all the lines from his hardcopy map and creates identical digital map on the computer. It is very time consuming and level of accuracy is also not good.

- **Heads-up Digitization:** It is similar to manual digitization in the way that lines have to be drawn manually but directly on the computer screen. So in this case the level of accuracy increases and time taken decreases.

- **Interactive tracing method:** It is improvement over Heads-up digitization in terms of speed and accuracy.

- **Automatic Digitization:** It is automated raster to vector conversion using image processing and pattern recognition techniques. In this technique computer traces all the lines, which results in high speed and accuracy along with improved quality of images [151].

There are many issues to consider before the commencement of digitization, including (McGowan, 1998):

- For what purpose will the data be used?
- What coordinate system will be used for the project?
- What is the accuracy of the layers to be associated? If it is significantly different, the layers may not match.
- What is the accuracy of the map being used?
- Each time you digitize, digitize as much as possible. This will make your technique more consistent. For more consistency, only one person should work on a given digitizing project.
- If the source consists of multiple maps, select common reference points that coincide on all connecting sheets. Failure in doing this could result in mismatch of digitized data from different data sheets.
- If possible, include attributes while digitizing, as this will save time.
- Will it be merged with a larger database?

Actually digitization is a process by which all/required objects of a map can be converted as polygon or line or point object. Polygon, line and points are the basic unit components of a vector map. But in a raster map or scanned map all objects are constructed by pixels and there is no boundary attributes of the objects. In case of vector map a database is attached with the unit components of the map. Raster to vector conversion process is called the digitization. To develop GIS based application tool generally the map is digitized using mouse of a PC base system. Two digitizing techniques published in journals are presented in sections 3.2 and 3.3.
Heads-up digitization
using mouse click
3.2.1 Introduction
This is the vectorization of raster maps/images using database management system. This technique calculates the coordinates of the point constructing the object from the raster map and store into the database along with the assigned colors of the objects. The spatial coordinates are calculated by the mouse click method. The boundary and filling colors are assigned to each object at the time of digitization. A mapping technique has been generated to retrieve the attributes from the database to regenerate the digitized objects of the vector map for visual display/interpretation. Provision has also been made to edit digitized objects for correctness or future upgradation. During editing new objects may be inserted, objects may be deleted and color of an object may be reassigned. Along with the tables containing the spatial coordinates and color of the objects this technique stores the name and path of the input raster map so that the user can edit the vector map later.
There is no computer understandable object into a raster map because it is a collection of pixels which have some intensity value only. So it is very difficult to generate any GIS features on this type of maps. The first task in GIS is to digitize the raster map to obtain a vector map. The output of digitization is the vector map which is a collection of polygons, lines and points. Along with the boundary point’s coordinates the polygons may have the boundary color and filling color features. Similarly the line and point objects may have the color feature along with the object constructing point’s coordinates into the database. It is very easy to generate GIS feature on a vector map because the computer can understand the objects as polygons or lines or points [152, 153, 154]. Section 3.2.2 of this chapter deals with the scheme of the implementation techniques. The implemented results and conclusions are given in section 3.2.3 and section 3.3.4 respectively.

3.2.2 The technique
The process of digitization/vector map generation technique is presented in this section. The total scheme may be achieved through four sub tasks which are as follows:
1. Calculation of coordinates of points constructing the objects
2. Color assignment
3. Temporary storage allocation for partial digitization
4. Saving digitized information
5. Editing of the vector map
All of these are discussed in section 3.2.2.1 to section 3.2.2.5 respectively.

3.2.2.1 Calculation of coordinates of object constructing points and storing
The digitization technique calculates the coordinates of the object constructing points and stores the coordinates of each object into the database. In this technique the coordinates are calculated by the help of mouse click. There is no need to calculate the coordinates of all object-constructing points. For polygon and line digitization, only some coordinates of essential points is required to produce the vectorized object. The absolute shape of an object can be constructed by help of these essential points. But more number of points (more closely spaced points) generate better vectorised object. Since there exist three types of objects (polygon, line and point) in a vector map, this process can be divided into three sub-tasks. These are:

1. Calculation of coordinates of polygon constructing points
2. Calculation of coordinates of line constructing points
3. Calculation of coordinates of point objects

All of these are discussed in the following:

Calculation of coordinates of polygon constructing points
Coordinates for the polygon are obtained by the clicking of mouse. No need to calculate the coordinates of all possible boundary points. Number of points may be reduced by using the concept of constructing small line segments. So only coordinates of the two end points of a line segment are essential. If ‘n’ number of line segments are required to construct the polygonal shape of an area then only the coordinates of ‘n’ number of points are to be stored into the database under the corresponding table. This will minimize the table space in the database but more points increase the perfection level of polygon digitization. A polygon name is then assigned; this name and the coordinates are stored in arrays. The coordinates will be stored in the database under table. Each table contains one polygon’ boundary coordinates. The technique of polygon boundary calculation is shown in figure 3.2.1.
Figure 3.2.1: Polygon boundary calculation technique

**Calculation of coordinates of line constructing points**
In this case it is also not necessary to collect the coordinates of all points on the line. Only some points of the line are necessary to maintain the absolute shape of the line. This will minimize the table space in the database. A line name has to be assigned, this name and the coordinates are to be stored in arrays. The coordinates will be stored in the database under the table whose name will be created with the help of the assigned save name of the digitization. So after taking the assigned save name the table will be created. Supposing the line name is ‘llll’ and assigned name of digitization is ‘ssss’ then the name of the storage table for coordinates will be ‘sssslinllll’.

**Calculation of coordinates of point objects**
The coordinate of the point can be calculated by the same method. The assigned name of the point and the coordinate are to be stored in arrays. The coordinate will be stored in the database under table whose name will be created with help of the digitization saved name. So after taking the digitization saved name the table will be created. Supposing the point name is ‘pttt’ and saved name of digitization is ‘ssss’, then the table name will be ‘sssspoipttt’. All the required points will be saved into the database the procedure mentioned above.

The same mechanism is also followed to generate the name of the coordinate storage table for polygon objects and here the digitization saved name is attached with the string ‘pol’ and assigned polygon name by the digitizer.
3.2.2.2 Color assignment

Color will be assigned to each objects of the map. The polygons have two colors boundary color and filling color. Lines and points also have color attribute. To represent different type of objects separately, different colors may used. In this technique, a color chooser button is placed in the digitization page. By this color chooser, colors are assigned to each objects of the map. The attributes of object’s color of a vector map are shown in figure 3.2.2.

![Color attributes of a vector map](image)

Figure 3.2.2: Object’s color attributes of a vector map

3.2.2.3 Temporary storage of partially completed digitization

This digitization tool has an extra facility which is temporary storage of partially completed digitization. If the total number of objects is large, the digitization process may be time consuming. To make it efficient this technique is integrated. The records of completed digitized objects can be stored into the database as tables. The coordinates of all object constructing points of an object must have to store as table in the database. The names of all objects along with the RGB values are stored in the database. Another table which contains the name of the raster map which is used for that digitization is required. By retrieving the data from these database tables the pending task can be finished. In subsequent access, when the pending tasks completed, the temporary tables are deleted.

3.2.2.4 Saving digitized information

At the beginning of software installation one muster table having two fields may be created. The field 1 contains the saved name of the created vector map and field 2 contains the name of the corresponding raster map. The raster map may be used later to edit the obtained vector map.
When digitization of all objects are completed then the saved name is assigned. The saved name must be unique in the database. At the time of assigning saved name a search is performed to check its name unique. After taking the unique saved name, this technique appends the saved name and code for object type (i.e. ‘pol’ for polygon, ‘lin’ for line and ‘poi’ for point) at the beginning of all the object names to get the names of the modified object. This process may create tables with unique name with respect to the database. Under each of these tables, the coordinates of the object constructing points will be stored. Another two tables also will be created at the time of saved name assignment. First table contains the modified object names and object types and second table contains the modified object names and the object colors. The table names are derived with the help of assigned save name in such a way so that all tables under a digitization process can be accessed to produce the corresponding vector map. The algorithm to save the digitized information is given below:

**Algorithm**

*Input:* Digitized coordinates, assigned colors and a unique save name

*Output:* Save all digitized and color information into the database.

*Method:* Check the existing names of the store vector map with the assigning name for the uniqueness of the vector database. Insert the name of the newly created vector map and name of the raster map into the master table. Create database for the vectorized object and its colors.

**Step 1:** Compare the assigned saved name with the names of the existing tables of the database. If the name is unique with respect the database then accept it. Otherwise repeat the name assigning process.

**Step 2:** For the created vector map insert the assigned save name and the name of the raster map into the master table where these information stored as a record.

**Step 3:** Create table for each object which contains the x, y coordinates of the object constructing point/points. Generate the names of tables of this step by appending three substrings where substring 1 is the assigned saved name, substring 2 is the three first character of the object type and the last substring is the assigned object name.
**Step 4:** Create a table which contains the names of the tables created in step 3 and the corresponding object types.

**Step 5:** Create a table which contains the names of the tables created in step 3 and the assigned color/colors for the objects.

### 3.2.2.5 Editing of the vector map

On completion of digitization operation, the vector map can be generated with edit option (‘EDIT’ button). If the user wants to edit the vector map, then the vector map layer will be generated on the original raster map. The hierarchical organization of edit menu items is shown in figure 3.2.3.

![Figure 3.2.3: Menu items in the editing page](image)

If the user needs to delete an object, then he has to select that object. To select a polygon object the mouse needs to be clicked inside the polygon object. For line selection, mouse is to be clicked at the end or any bending position of the object. The polygon and line selection scheme is shown in figure 3.2.4.
In case of polygon selection the program search the polygon which surrounded the clicking point. The polygon which enclosing the point will be selected. For line all the bending and end points of a line are fetched from the database and small rectangles are formed around every fetched point. If any rectangle contains that clicked point then the corresponding line will be selected. For point object selection the mouse is to be clicked on that point or inside the small rectangle created around that object. For line and point objects, the small rectangles will not be drawn on the vector map but program will generate internally. The object blinking fetcher is also integrated in this tool. This fetcher makes the tool more user friendly. If a selected object is drawn with two separate colors with a fixed time interval, then the object will be blinked. After selecting an object the color can be changed and the object can also be deleted from the database.

The facility of object insertion is integrated in this tool. Here the calculation of boundary points coordinate and storing and the color assignment techniques are to be reinitialized. For object insertion operation, the tables created in saving step will be indifferent state with more records and tables containing the boundary data are to be created.

### 3.2.3 Implementation and results

The implemented resultant pages and its operation are presented and discussed in this section. The implementation has been done by using java swing and oracle. At the beginning of the original operation, the input image will copy into a specific folder mentioned in the original programs. The organization of modules is shown in figure 3.2.5.
Figure 3.2.5: Structure of the modules of the technique

Figure 3.2.6 shows the initiation page of this implementation. There are five panels of this page. By selecting the name of the vector map from the first panel, the user can see and edit the vector map. The second panel provides the opportunity to observe the selected vector map. If there is any terminated vectorization operation, a message informs the user to complete the pending task. The user can continue the pending task by pressing the ‘GO’ button of the 3rd panel. To create a new vector map, the user has to go through the 4th panel. The 5th panel helps the user to delete the selected vector map and all the corresponding records from the database.

To create new vector map, the user has to press the ‘GO’ button of the 4th panel. The file chooser program helps the user to select the raster map. Then the main digitization attributed form, shown in figure 3.2.7, will be opened. The user has to select the object type and has to assign the object name in this page. For polygon object the following operations will be performed sequentially.

1. Assignment of polygon name
2. Selection of boundary color
3. Selection of boundary points
4. Selection of fill color
5. Pressing ‘GO’ button for database operations

If the selected object type is ‘Line’ then the following operations has to perform
1. Assignment of line name
2. Selection of line color
3. Selection of line constructing points
4. Pressing ‘GO’ button for database operations

For point type object the following operations have to perform sequentially
1. Assignment of point name
2. Selection of point color
3. Selection of the point by mouse click
4. Pressing ‘GO’ button for database operations
At the time of object-boundary point’s coordinates selection if any wrong point is selected then that point will be eliminated by pressing the ‘UNDO’ button of the page of figure 3.2.7. By pressing the ‘COLOR’ button the user can select color for the object. The color chooser is shown in figure 3.2.8.

If the user wants to terminate the program temporarily by saving the partially completed digitization, then the user will press the ‘BREAK’ button of the page of figure 3.2.7. After the completion of total digitization, the ‘COMPLETE’ button of the digitization page has to be pressed.
If the digitization is completed and the ‘COMPLETE’ button is pressed, then the form of figure 3.2.9 will be opened. In this page the name of the vector map under which all records will be stored in the database will be assigned. If there exists any table in the database of the same assigned name then a dialog box inform the user that assigned name has to be changed. But if the assigned saved name is unique in the database, then the conformation dialog box shown in figure 3.2.10 informs the user that the digitization saved in the database successfully.

After the save operation, the output vector map will be visualize in the ‘vector map visualization page’ which is shown in figure 3.2.11. The user will decide whether he want to accept this digitization or not. If the user wants to change the color of an object or wants to add or delete any object then he has to press the ‘EDIT’ button. If ‘EDIT’ button is pressed then the ‘vector map editing page’ shown in figure 3.2.12 will be opened.
New object can be inserted or existing objects can be deleted from this page. There are two lines in the middle of figure 3.2.12 which are deleted to obtain the vector map shown in figure 3.2.13. If the user wants to insert a new object, then he has to select required menu item from the menus of the menu bar. After the selection, if ‘ACTION’ button is pressed, one page similar to the ‘Digitization Page will be opened where the required object can be digitized along with the color assignment. After the object-insertion if ‘BACK’ button is pressed then the edit page containing the newly inserted objects will be opened again. If the user wants to delete any object from the
vector map then he has to select the object by the method discussed in section 3.2.2.5. The selected object will be blinked with two different colors (In this implementation red and green). The blinking fetcher is developed using thread. If the object blinks then the ‘ACTION’ button has to press to perform the required operation.

![Edited vector map containing one boundary line and six polygons](image)

Figure 3.2.13: Edited vector map containing one boundary line and six polygons

When editing operation is completed, the ‘EXIT’ button has to be pressed. The output digitized map can be opened by selecting the digitization-saved name from the Initial page. A completely digitized map is shown in figure 3.2.13.

### 3.2.4 Conclusion

By this head-up digitization technique a user can digitize maps for small projects. The databases to be generated in the digitization operation can be used for the further reference of the data attachment and thematic representation. The temporary storage and vector map editing is also integrated in this technique to facilitate the users to complete time consuming operations efficiently and make corrections on the previously performed mistakes. Analysis, comparison and more conclusions are also discussed in chapter 8 and 11.
Automated digitization of Polygon Segments
3.3.1 Introduction

There are many techniques to digitize maps in various GIS based systems but most of them use the head up digitization techniques such as mouse movement, mouse clicked or dragging it along the borders of the various sections of the maps [153, 155]. The raster-vector conversion of a binary digital image is a very important task for updating roads and other linear cartographic objects in cartographic processes. However, many troubles appear in the automatic solving of this problem. On the one hand binary image is frequently obtained as a result of any automatic segmentation method, reason why noise and other undesirable distortions have to be considered. On the other hand the variability in the geometrical structure and properties of the road networks is so high that a manual process in order to improve automatic results is often needed [156]. For open sections, it is still a reality that it has to be selected using mouse movement or dragging along the borders of the section but for closed sections it is possible to digitize the whole section using a single mouse click inside any point of the section.

A GIS tool has been designed and implemented where the whole border of a closed section of a black and white map is digitized using a single click inside the section (instead of moving the mouse pointer along the border of the section). The boundary points of a segment contain similar RGB values. When one inner point of a closed segment is selected by the mouse, the program calculates the RGB value (which is not black) and stored in an array say inner array. Fetching elements from the inner array, the program calculates the RGB values of the eight connected pixels. If the pixels are black then stored into another array say boundary array. Other points are stored into the inner array. The elements which are fetched must be removed from the inner array. The fetching and comparing RGB values of the eight connected pixels are performed until the inner array becomes vacant. When the inner array becomes empty the boundary array contains the boundary pixels. Thus boundary pixels are identified. The array elements of boundary array are arranged in a cyclic order to generate the proper boundary of the polygon object.

Section 3.3.2 of the chapter describes the proposed technique. The results of the implementation of the technique are presented in section 3.3.3. The conclusion is drawn in section 3.3.4.
3.3.2 The technique
This technique digitizes the polygon objects which are completely closed and contains no black pixel inside the segment. If the required segments are not closed by the boundary or if there exists any black pixels inside the segment then manual image correction is needed. Once the closed boundary of the segment is obtained the rest of the process of the scheme can be preceded. The technique may be splitted into four subtasks.

i. Image correction
ii. Spatial data generation
iii. Creation of database
iv. Output generation

These four subtasks are described in section 3.3.2.1 to 3.3.2.4. The Flow diagram of the scheme is shown in figure 3.3.1.

![Flow diagram of the technique](image)

3.3.2.1 Image correction
First step of the scheme is to make correction of the raster/ Scanned map, if necessary. The calculation of boundary/ spatial data for polygon object are performed by single click in a closed boundary. So the map is to be tested to identify whether there exists any discontinuity in the broader line of a segment. If it exists, the boundary is to be made continuous. The correction is to be made in such a way that black pixels within the segments are also removed/ converted for better results.

3.3.2.2 Spatial data generation
This technique calculates the RGB value for selected point (the clicked point). Since the selected point being inside the closed boundary of the segment, this RGB value must be unequal with the
boundary points’ RGB value. The RGB values of the eight connected pixels of the selected points (figure 3.3.2) are then calculated and compared with the RGB value of black pixel.

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>X</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
</tbody>
</table>

Figure 3.3.2: Eight connected pixels of the selected pixel ‘X’

The points/pixels with different RGB values from the boundary are stored into an array (say array 1). RGB value -16777216(RGB value of black pixel) indicates the boundary points of the segment. Boundary points are stored in another array (array 2). After the comparison the selected points are removed from the array 1. Fetching pixels from the array 1 the RGB values of its eight connected pixels are examined. If the RGB value is not equal to -16777216, then it stored in array 1 else stored in array 2. All the boundary points are calculated using this mechanism. The pixels which are fetched from the array 1 are removed. At the end of the process, array 1 will be vacant and array 2 contains the pixels of the boundary of the segment. The boundary pixel calculation process is shown in figure 3.3.3.

![Block diagram of boundary pixel calculation process](image-url)

Figure 3.3.3: Block diagram of boundary pixel calculation process
The co-ordinate points remain in array 2 are not in order so, the elements are then arranged in an order to generate the polygon properly. The algorithm of segment’s boundary calculation is given below:

**Algorithm of segment’s boundary calculation**

*Input: Raster map, Clicked point inside the segment.*

*Output: Boundary points of the segment.*

*Method:* Gradually move from the inner selected pixel to the boundary side pixels and compare the elements with the boundary. Store the obtained boundary pixels in an array in orderly manner.

*Step 1.* Store the clicked point in to an array (array 1).

*Step 2.* Fetch point from array 1 and compare the RGB values of the eight connected pixels with the RGB value of the black pixel. If the RGB value is unequal then store the pixel in to array 1 else store in array 2, which contains the boundary pixels of the segment. Remove the fetching point from array 1.

*Step 3.* Repeat step 2 until array 1 becomes vacant.

*Step 4.* Arrange the elements of array 2 in orderly manner.

*Step 5.* Stop.

### 3.3.2.3 Creation of database

In GIS, the database is essential for all objects of a vector map. Since this tool produce polygon objects of a raster map, the database management system is essential to store the obtained spatial data. The spatial data i.e. the coordinates of the boundary points of a segment are stored into the database. For each segment, one database table is created, which contains the coordinates of the boundary points of the segment. The table names for each segment are assigned in accordance with segment name. A table is also created which contains the name of the tables which contains the spatial data of the segments.

### 3.3.2.4 Output generation

The output will be the vector map containing the generated polygon objects. The coordinates of the segments are fetched from the database table and the polygon objects are constructed using
the coordinates. The output vector map only contains the polygon objects because it is the closed segment boundary calculation tool.

3.3.3 Results
The technique has been implemented through Java swing and Oracle 9i. The DBMS is used to store the boundary coordinates in the database. Figure 3.3.4 is the first input map and the output is shown in figure 3.3.5. The first output only contains 17 interested polygon objects which are obtained from the segments of input raster map.

Figure 3.3.4: Input raster map 1

Figure 3.3.5: Output vector map contains only 17 segments

Figure 3.3.6 shows the second raster map and the output vector map is shown in figure 3.3.7. This output contains all the segments of the input raster map. These output polygons are obtained by the single mouse click inside the closed segments.
3.3.4 Conclusion

This technique digitizes the closed segments of a black and white map as polygon objects. One single mouse click calculates all the boundary coordinates of a polygonal segment. Those coordinates can be stored in a database, to be used later for data attachment and thematic representation. This new automatic digitization technique produces the digitized objects in accurate shape and the time required for digitization is very low.