Applied Information Technology And Computer Science Vol. 2 No. 2 (2021) 037-052

AITCS

Homepage: http://publisher.uthm.edu.my/periodicals/index.php/aitcs
e-ISSN :2773-5141

A Mobile Forensic Visualization Tool for Android Data Partition

Chow Xiang Quan, Nurul Hidayah Ab Rahman*

Fakulti Sains Komputer dan Teknologi Maklumat,
Universiti Tun Hussein Onn Malaysia, Batu Pahat, 86400, Malaysia

DOI: https://doi.org/10.30880/aitcs.2021.02.02.003
Received 14 June 2021; Accepted 09 September 2021; Available online 30 November 2021

Abstract: In the 21th century, digital crimes would be one of the biggest challenges to government and public. Digital crime cases that involve mobile phones are on the rise, resulting in digital forensic analysis tools are on the demand. However, there are limitations in the current mobile forensic tool, such as lack of automation and visualization process, false positives are too high and performance of the analysis is low. This study therefore aims to design, develop and test a tool - MF Visualizer – to visualize the metadata from databases in the Android data partition. The android data partition is chosen as the scope of the project. MF Visualizer follows the mandatory requirements of the forensic tool and is compatible with suitable modules to accomplish the task. The tool is developed by adopting Object-Oriented Software Development Model and using .Net Windows Presentation Foundation (WPF) framework to develop. The findings show that the tools could extract metadata from android data partitions as well as visualize the data in different visualization forms such as Bar Chart, Word Cloud, Map, Pie Chart and the Timeline method. Functionality and users testing results indicate that MF Visualizer has achieved the project objectives. This further indicates that MF Visualizer is a promising tool to be used in a real world scenario with further improvements.

Keywords: Android Data Partition, Mobile Digital Forensic Analysis, Visualization

1. Introduction

With the increase of digital crimes, the demand of digital forensic analysis is on the rise due to the involvement of digital devices (e.g. personal computer, mobile phone, workstation server) for digital investigations. The digital forensic evidence extracted from the device can be an invaluable evidence source for investigators in civil litigation and criminal prosecution [1]. The analysis on the evidence of the device will be conducted with forensic tools and professionals passed through standard procedure to ensure the integrity and accuracy of the result [2].

With the sophisticated features of smartphones, a lot of stored data could be used as clues for an investigation, for instance location details, phone calls history and text messages [3]. To analyze the data stored in the devices, digital forensic professionals will conduct the analysis with suitable mobile forensic tools. There are some limitations present in the current tool. The first lack of automation and

*Corresponding author: hidayahar@uthm.edu.my

visualization process is the current tools still require manual forensic analysis as the modulo is not fully considered. The second issue is the false positive is high in current forensic tools due to some of the file may be missed. The third issue is the performance of the current tool is too low[4].

This study therefore attempts to propose a tool – MF Visualizer – that is able to extract key metadata from Android data partitions and visualize the metadata to facilitate forensic analysis activities. The objectives of this study are to design, develop and test a proposed tool to solve the current issues present in the current mobile forensic tool and integrate the visualization features. The tool would not involve the acquisition process of the digital device.

The paper is organized into 5 sections. Section 1 introduces this study background while Section 2 discusses relevant reviews includes mobile forensics and visualization method. Section 3 describes the methodology and Section 4 presents the result and discussion. Section 5 concludes this study.

2. Literature Review

2.1 Digital Forensics and Mobile forensics

Digital forensics is the study of the process to find evidence in digital devices [5]. The digital forensics involves four phases which are collection, examination, analysis and reporting [6]. Following the phases is important in digital forensic as this could ensure the data integrity in the digital evidence. The process of digital forensic would also require to follow the digital forensic standards such as ISO/IEC 27043:2015 international standard to ensure the consistency and accuracy of the analysis result [7].

Digital forensic can be divided into different sub-branch based on the type of investigation device and operating system. There are four main branches of digital forensic which are computer forensics, database forensics, network forensics and mobile device forensics [4]. Android mobile forensic is under the mobile device forensics.

Mobile forensics involves the acquisition of the data from the mobile phone, extracting the information for the data, analyzing the data with suitable tools and the presentation of the result from the forensic analysis [8]. As the mobile phone has two platforms which are Apple IOS Operating System and Android Operating System, there are different methods in mobile forensic investigation. The architecture of both operating systems is totally different such as Apple IOS would use UNIX based operating system while Android would use Linux 2.6 kernel based operating system [9].

2.2 Android Operating System

Android OS is an open source operating system based on kernel 2.6 of Linux Operating System. Google is the company to develop the Android OS while other companies would modify some extra features and release different update patches every year. Therefore, there would be some difference in each of the custom Android OS.

The database would enable the application to store the user data in structural form. The default database used by Android OS is SQLite and it is an open source and relational database [10]. By default, the database of applications would be stored in the database folder with the corresponding folder of package name and it is invisible to other users [11].

2.3 Android Data partition

Android has a file system that is similar to Linux Operating System. In Android, there are different partitions that have different functionalities and roles. There are six directories in the android file system that represent different functionality.

Table 1: The path and functionality of android file system [9].

Directory	Functionality
boot	The partition that allows the phone to boot and all the data for the phone to boot.
cache	The partition that stores frequently accessed data and app components.
data	The partition that contains user data.
recovery	The partition that has recovery file of Android OS
sdcard	The partition that the path of internal SD card
system	The partition that has system application, Android OS and user interface

Table 1 shows the functionality of the each directory. There are different functionalities based on different directory in Android class folder. For example, the boot directory would be the partition that stored all of the components needed for the phone to boot. In this study, Android data partition is the main concern. It is the path where the storage of the user data and it is located in the folder path android/data. It will be located in the folder path: "Android/data/<package name>" where package name refers to the package name for different applications [12].

For security purposes, the Android OS is invisible and hidden from end users. To access the file system that is hidden, the user needs to get the highest permission which is super administration mode or root [8]. The reason is only the super user is able to view and retrieve all the files from the path of android devices.

The root is needed to obtain by the investigator to acquire the target device [13]. Therefore, the digital forensic analysis would need to request permission from the device owner or under investigation purpose before starting to acquire data from the device.

2.4 Visualization in Forensic Analysis

The visualization in forensic analysis is the process to transform the data to the content that could be understandable by the public. For example, judges and juries who do not have formal knowledge in digital forensics. The visualization has the potential to increase the efficiency of the forensic analysis as the visualization can simplify the step of manual forensic analysis [14]. As reviewed in [1], potential visualization method that can be applied for forensic visualization such as bar chart, map, word cloud, map and timeline.

2.4.1 Bar Chart

The bar chart is one of the outcomes of visualization that is suitable to output the result for numerical types of data. The bar chart is suitable for the numerical data since it allows comparison between the numerical data [15].

There are significant differences between the horizontal and vertical charts[16]. The horizontal bar chart is used when there are less elements in the x-axis while vertical bar chart is used if there is more elements that want to present.

2.4.2 Map

The map plots are used to represent the location details in the physical location based on the latitude and longitude coordinates. The coordinates details can be obtained based on the logfile or the EXIF data in the media file. In Android mobile forensics, the database file may store the latitude and longitude coordinates in the columns[14].

The example of maps plotting in map visualizer is shown in Figure 3. The mapping library used is OpenStreetMap. There are a few pins in Figure 3 which shows the history of location of the users of the devices. This feature would be useful to visualize the meaningless longitude and latitude coordinates value to map form. This could also be suitable for documentation to describe the result of the forensic analysis.

2.4.3 Word Cloud

Word Cloud could visualize the data that commonly occurs in the events. Word clouds are a simple and intuitive visualization technique to provide a first impression of text documents [17]. It can integrate in the forensics tool to show the string that has most times of frequency in the communication. It is very suitable for visualizing the evidence such as messages.

In Figure 4, the word "ROCK" would be the highest frequency of the word which represents the biggest font size. It is suitable for the presentation of the result of text processing and can be used to visualize data in the database of forensic analysis.

2.4.4 Timeline Method

The timeline method would be used to show the most occurrence events in a specific timeframe [15]. The investigation may only need the data in a specific moment in time that the events happened. Therefore, the timeline method in digital forensics tools could be useful to find the evidence.

The benefit of the timeline method could enable the investigator to focus more on finding evidence instead of technical issues in the tool. The timeline method could provide the search and sorting modulo that concerns the timeframe.

2.5 Mobile Forensic Analysis

There is some useful information that can be obtained in the android data partition. In the android data partition, there are different folder stores in there. In an unrooted android device, the folders in the path will be invisible and hidden for the user. The folder will only be visible to the corresponding application and the super user. To gain privileges, the forensic investigator needs to gain the super administration mode to gain access to the folder for analysis. There are different folders that will be stored in an application.

Table 2: The folder in corresponding folder package name [18].

Folder Name	Data inside the folder
webview	webview data (cache and cookies)
databases	SQLite database and its temporary file
files	any type of file
cache	app cache data
lib	external software library that import by the application
shared_prefs	Preference file in android XML based format

Table 2 shows the folders that may exist in the Android class folder. This is the folders that stored different source of data. For example, there is webview files that stored the web browser data, such as caches and cookies.

The database file will be found in the corresponding folder package name. The database file is used for each application for the storage purpose. In the Android operating system, the database file will be

stored with all of the privacy and personal information. The automation process is required for the forensic tool to present data that was retrieved from the database [19].

In the database file, it will have several tables. The default of the database of Android is SQLite, the forensic tool would require integrating the SQLite browser to open the table in the database [3]. After the SQLite browser extracts the table from the database, the normalization of the database is needed for the data visualization to output the data in more readable format or visual data [19]. There are many evidence that is stored in the tables in databases once the application is done with the operations [20].

From the previous research paper, there are facebook forensic analysis tools and Whatsapp forensic visualizer tools that have been done [21] [10]. The studies were focusing on one of the applications for analyzing the evidence that may occur on the database. This is because the user data would be the potential evidence of a crime and useful for forensic analysis [22]. The database in the Android data partition would be the potential evidence sources for the investigator. Therefore, the investigator would always look for the user data and the database would be the file that stored the user data created by user and application synchronization [23].

3. Methodology

In this study, the Object Oriented Software Development model was adopted as a development model [24]. The reason of choosing this model is because the programming language chosen is C# and it is an object-oriented programming language. It consists of five phases and is described in the next section.

3.1 Object-Oriented Requirement Elicitation

This is the first phase that focuses on describing the purpose of the new system. In this phase, the requirements will be analysis which include functional requirement, non-functional requirement, software requirement and hardware. Elicitation activities identified that seven modules are needed by MF Visualizer namely: (1) import the android image, (2) extract image data t o data form, (3) analysis the data, (4) data visualization, (5) data sorting, (6) keyword searching and (7) generate report.

The seven modules are the phases that used in the digital forensic framework. The phases of the digital forensic would defined as acquisition, extraction, analysis and reporting. The acquisition would ignore because it is not in the project scope. The extraction would involve the process of extracting the Android Class Folder, each of the files in folders will be scanned and classified into database file and no database file which represent whether there are potential evidence. The extraction will process the files to databases, databases to tables and tables to rows and columns.

The analysis phase would involve the analysis the data, data visualization, data sorting and keyword searching. The proposed tool is allowed the user to analyze the data, select the data and perform data visualization. Besides, there are data sorting and keyword searching are included in the proposed system.

Lastly, the reporting phase would be generate report based on the data visualization. Then, the report can be exported to PDF document.

3.2 Object-Oriented Analysis

This phase focuses on the production model of the system which is complete, correct, consistent and verifiable [5]. The Unified Modelling Language is used to define the requirements of the tools. There are functional models, object models and dynamic models in UML to describe the requirements. For a functional model, a User case diagram is used to describe the functionality of the tool from the user's point of view. The class diagram is known as an object model to describe the overview structure of the tool in attributes, associative and method while the dynamic model will present in sequence

diagram and activity diagram (refer to Appendix A) to show the flow of the tool and the interaction between the objects. In the context of forensic analysis, this phase involves identifying potential visualization methods to represent the extracted metadata.

3.3 Object-Oriented Design

Design phase includes visualizing the result from the previous phase. Interface and algorithm design are the two main activities in this phase (refer to Appendix B). The interface design of the tool will be present to visualize user' interaction with the tool. The algorithm design involves pseudo code design to the identified seven modules.

3.4 Object-Oriented Implementation

This phase is to implement all the features and develop the MF Visualizer as according to the previous phases. The main task of this phase is to translate the ideas to application to follow the requirements. The proposed tool is developed by using C# programming language with .NET Windows Presentation Framework (WPF) under visual studio.

3.5 Object-Oriented Testing

This phase is the last phase and used to perform functional testing and user acceptance testing. There are three testing plans used in this phase which are user testing, unit testing and system testing that will be carried out.

4. Results and Discussion

This section discusses the implementation of the proposed tool. Firstly, the MF Visualizer will be going through forensic analysis to obtain evidence. The evidence will be used to determine the suitable visualization model. Lastly, the implementation and testing of the tool will be discussed.

4.1 Evidences in Android Mobile forensic analysis

To obtain the potential evidence in Android, the image source of Android 10 from digitalcorpora.org was used. There is numerous evidence that can be obtained from the Android Data Partition. The analysis of the data partition would give the overview of the evidence that can be extracted and the evidence could be visualized.

Table 3: The data stored and data partition extracted from sample image of Android 10.

Data partition location	Data stored
/Android/data/com.android.providers.telephony/databases/mmssms.db	Messages
/Android/data/com.android.providers.maps/databases/gmm_sync.db	Geolocation and Timestamp
/Android/data/com.android.providers.photos/databases/gphotos-1.db	Geolocation of photo and Timestamp
/Android/data/com.android.providers.contacts/databases/calllog.db	Call log of phone calling
/Android/data/com.android.chrome/app_chrome/Default/History	Google chrome history

Table 3 presents data partition location and its corresponding data are stored in the database. A database file comprises several tables. The default database of Android is SQLite, therefore the forensic tool would be required to integrate the SQLite browser to open the table in the database file.

```
public void FetchDatabase()
{
        try{
        DataTable alltable = new DataTable();
        selectedtable = new DataTable();
        alltable.files[_selectedfile].databases[_selecteddatabase].GetAllTables(files[_selectedfile].datab
ases[_selecteddatabase].GetFilePath());
        int count = 0;
        if (files[_selectedfile].databases[_selecteddatabase].Getcounttable() == 0)
        {
                selectedtable.Columns.Add("Database Name", typeof(string));
                for (int j = 0; j < files[_selectedfile].databases.Count; j++)
                {
                        selectedtable.Rows.Add(files[_selectedfile].databases[j].GetDatabaseName());
                }
                foreach (DataRow row in alltable.Rows)
                {
                        files[_selectedfile].databases[_selecteddatabase].SetTables(row.Field<string>(2
                ));
                        count = count + 1;
                }
        }
        int i = 0;
        foreach (DataColumn col in selectedtable.Columns)
        {
                selectedtable.Columns[i].ReadOnly = true;
                i++;
        }
    }
    }
    catch
    {
        System.Windows.Forms.MessageBox.Show("Set databases error", "Message");
    }
}
```

Figure 1: Code segments of method FetchDatabase().

Figure 1 presents the method of processing the tables in the database based on user selection to combo box. There may be a few tables in the selected database, therefore there are needed for extra control to handle the tables in each of the databases. The code segments in Figure 1 started with the try and catch exception to make sure when there is error, the process will stop and alert the user with an error message. This is to ensure the integrity of the proposed tool, as consistent with Tassone [14].

The code will start with declaring two data table variables. One of the data table variables will be used to obtain all of the tables from that database and another variable will be used to fill the value in the combo box. It would start with an if statement to check whether the tables have been assigned. If there are no tables in the database, the process of assigning tables to the variable will start. It would start with for loop and for each loop. For loop is the process to assign the tables into combo box control that is visible to the user while for each loop is the process to assign the value to the list of the variables that work in the background.

Then, each of the columns will be assigned to read only to prevent the user from altering the data. This would prevent unintentionally alteration and preserve the integrity in the proposed tool to minimize the false positive and false negative that may happen in the tool..

After the SQLite browser extracts the table from the database, the normalization of the database is needed for data visualization to output the data in more readable format or visual data. The table names of databases have their own functionalities, there are some of the table use for configuration and some of the table use for the configuration and some of the table use for the data.

4.2 Selection of Visualization Model

From the metadata in Table 3, there is a need to choose suitable visualization models for each of the data that can be extracted from the database.

Table 4: Visualization Method and its example of metadata.

Visualization	Description	Examples of metadata
Bar Chart	Shows the frequencies of one event	Messages, phone call
Map	Shows the geolocation based on latitude and longitude	Geolocation of Google Map, photo
Pie Chart	Shows the percentages of one event occurs	Phone number, messages who had the highest frequency
Word Cloud	Shows the frequencies of words in one column of table	Frequencies of word in messages and text
Timeline Method	Shows the timestamp of one event	Timestamp of messages, map, photo

Bar Chart is the chart that shows the frequencies of one event. It is suitable to visualize the numerical data. Map is can visualize the geolocation and pin in the map. The map can provide the user with the coordination of latitude and longitude. There are geolocations that are stored in the database. Pie Chart is the chart that could help to visualize the data and extract in pie form that shows the percentages. It is very suitable to use if the investigator want to know the percentages of one event. Word Cloud is the chart that can show the frequencies of words in one column of table. With visualizing all rows in a table column, the investigator could know the most frequent word in that column. It is very useful for the investigator to know overall contents in all messages. Timeline method will be allowed to add as the additional feature for the other chart. It can be corporate with bar chart and map to visualize the map and bar chart data and based on the timestamp. This is very useful for the crime investigation that concerns about the timestamp.

4.3 Implementation

The analysis interface is the interface that enables the user to analyze the data from the image data. The analysis would start from the Android class folder, database, table, data in columns and rows. The user is allowed to choose 4 types of the visualization method to visualize the data in different visualization forms.

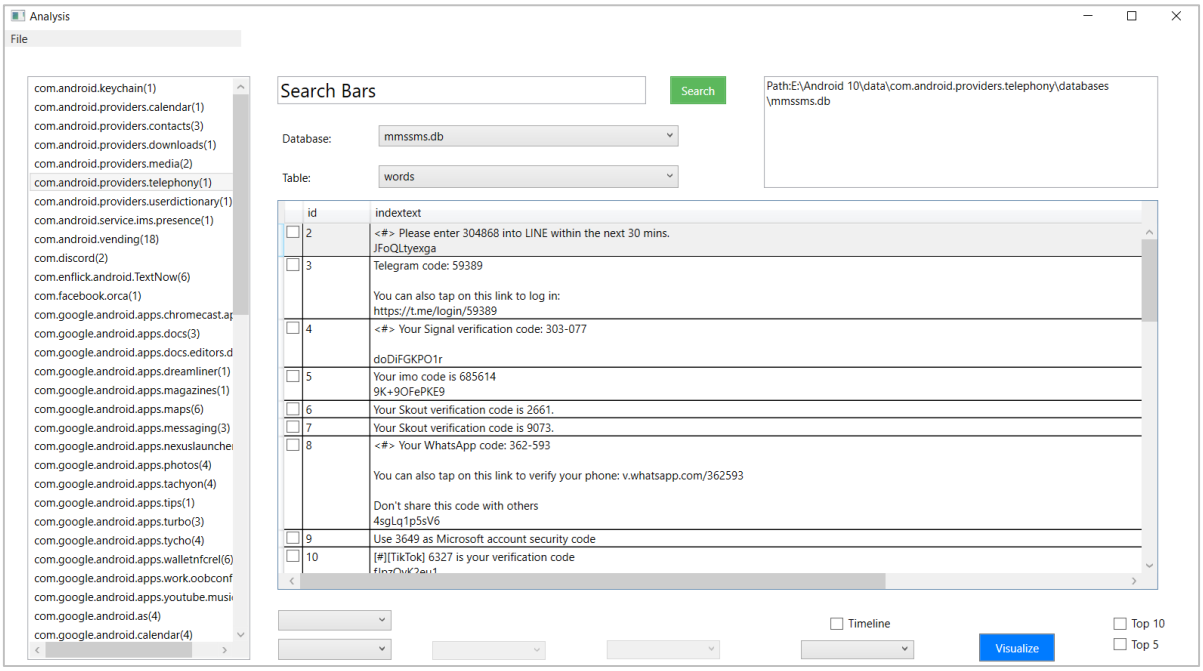


Figure 2: The analysis page of the tool.

Figure 2 shows the analysis panel that chose the "mmssms" database from the com.android.providers.telephony that can show the default message application in Android 10. In the table panel, the table words are chosen and it shows the messages. The user is allowed to choose the data from the column and passed to the visualization interface.

The visualization interface that enables the user to visualize the data from the analysis panel. This interface has different results based on the selection of users. The users are allowed to select the data and tabulate the data in visualize form such as bar chart, map, pie chart and word cloud.

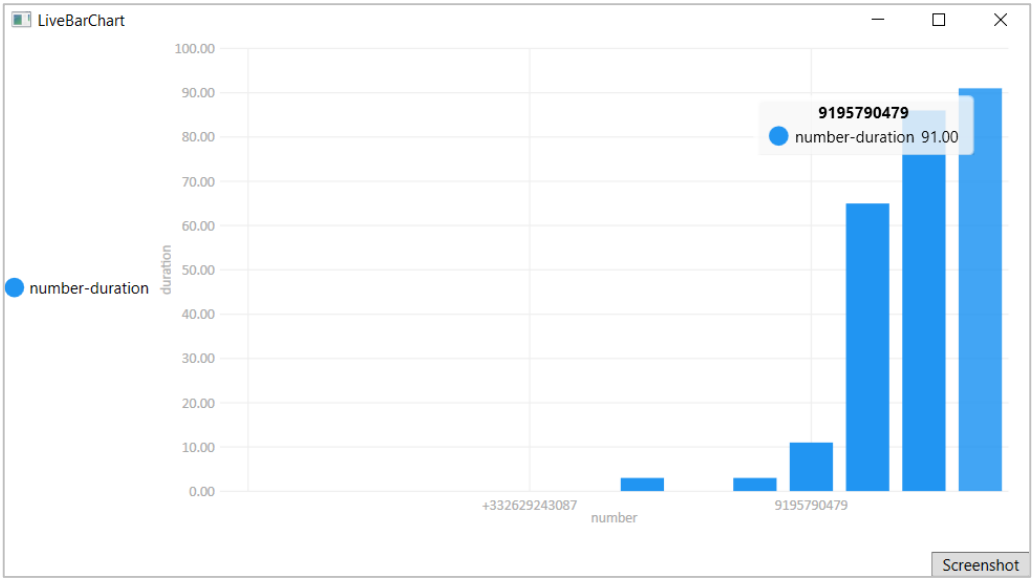


Figure 3: The Bar Chart in Bar Chart Interface [25].

The data in the call log database has been visualized in Figure 3. The number and duration has been used to represent the evidence. The phone number that called is labelled as number while the duration is the calling duration and it has 91 minutes of the calling duration. Potential key points of forensic analysis that can further discussed from Figure 3 such as (1) Who or what numbers are the top contacts?

(2) Which contacts has the longest or shortest call duration? (3) How long is the longest or shortest call duration?

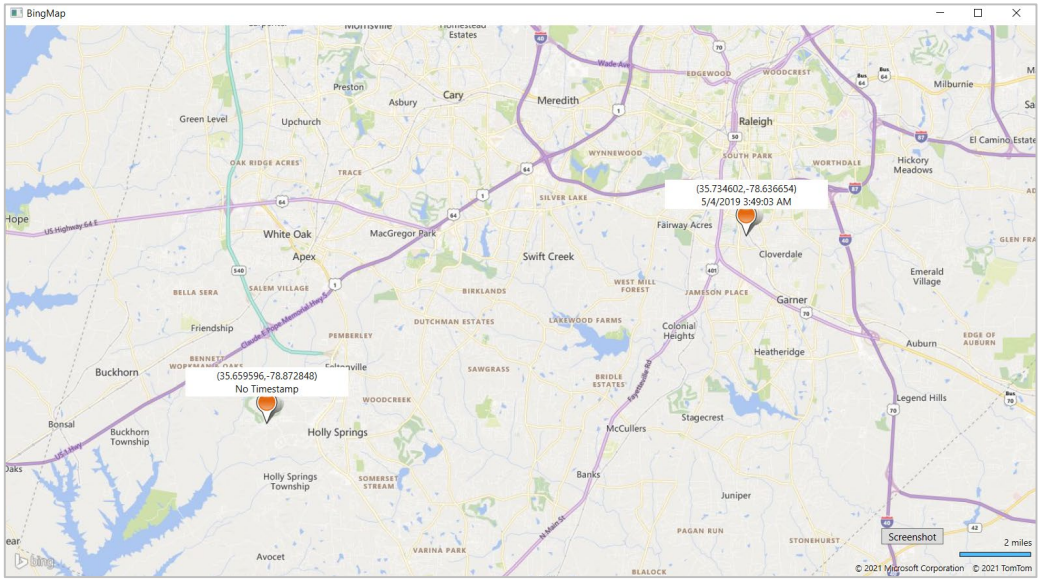


Figure 4: The map and timeline method visualization [25].

Figure 4 shows the map visualization via Microsoft Bing map with Android 10 data sample from digital corpora [25]. In the Figure, two locations are pinned by the application with the latitude, longitude coordinates and timestamp from the data in the table. This could help the investigator visualize the evidence from geolocations and timestamps that recorded by applications.

The most essential part of the Map Visualization would be the set pin for the map. As the database may contain some null value for geolocation that may be due to the failure of synchronization with the GPS module. It is important to filter the null value from the Map Visualization. Therefore, the source code will check whether the latitude and longitude coordinates are null. The result will assign to a variable flag.

If the flag is true, that geolocation will process to set pins. It would start with declaring the background, font size, width and other for the labels. Then, the geolocation will pin on the Map. The process will continue until the data finishes fetching all of the coordinates.

This feature is important to assist investigators find key locations faster, as they do not have to go for other map tools manually.

```
for (int j=0;j<i;j++)
{
        if(templongtitude==0|| templatitude==0)
        {
                flag = false;
        }
}
if (flag==true)
{
        locations[count] = new Location(LatitudeArray[i], LongtitudeArray[i]);
        pushpin[count] = new Pushpin()
        {
                Location = new Location(LatitudeArray[i], LongtitudeArray[i]),
                FontSize = 50.0,
        };
        var txt = new System.Windows.Controls.TextBlock()
        {
                Text = "(" + LatitudeArray[i] + "," + LongtitudeArray[i] + ")",
                Background = new SolidColorBrush(Colors.White),
                Foreground = new SolidColorBrush(Colors.Black),
                Width = 150,
                TextWrapping = TextWrapping.WrapWithOverflow,
                TextAlignment = TextAlignment.Center,
                Padding = new Thickness(2)
        };
        pushpinLayer.AddChild(pushpin[count], locations[count]);
        pushpinLayer.AddChild(txt, locations[count], new System.Windows.Point(-txt.Width / 2, -
height-30));
}
```

Figure 5: The source code that sets the pin on the Bing Map.

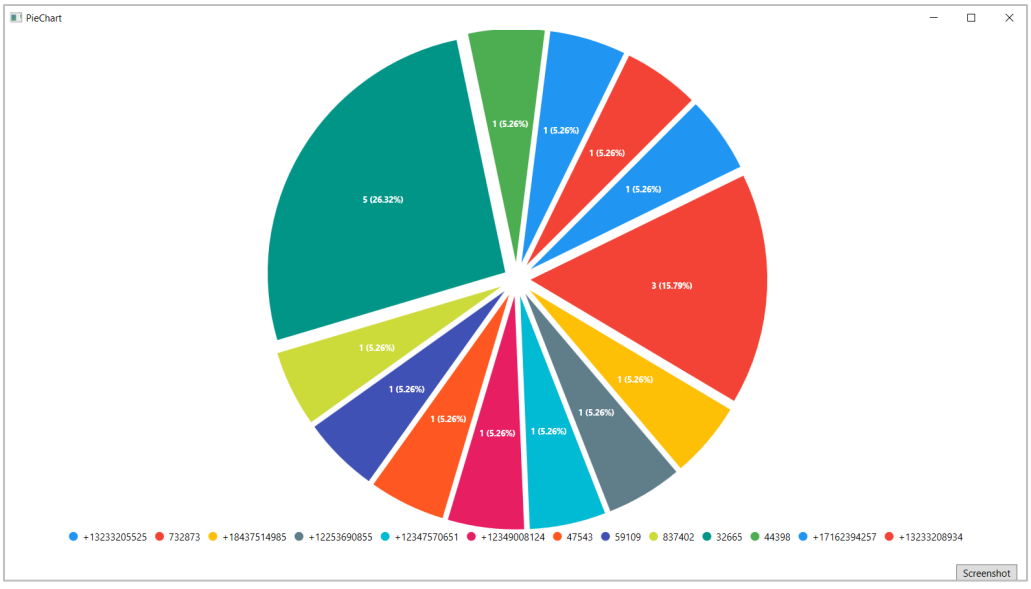


Figure 6: The pie chart of contact.db via Live Chart [25].

Figure 6 shows the percentages received from the other phone number. Pie Chart is visualized using the Live Chart Framework. Pie Chart has the advantages that it can clearly show the result in percentages. Figure 6 presents that phone number 32665 has the highest percentages. This feature would assist investigators to rapidly identify the most and least frequent contacts.

Figure 7: The Word Cloud of message.db [25].

Figure 7 shows the Word Cloud that output from the "message.db". The Word Cloud is very suitable with the messages that have large amounts of heterogenous text data and the tool will be able to give a simple overview of the message, as consistent with a study by Kucher and Kerren [26]. From Figure 7, it can summarize that Facebook, com, code, http, FB has the highest frequencies and would provides clues of the popular contents of received and sent messages.

Figure 8 shows the reporting interface that extracted from the com.google.android.photos/gphotos-1.db. The reporting is allowed user to edit the header and footer text for describe the evidences. From Figure 8, we can obtained the evidences of two geolocation and timestamp and this indicates the phone owner is using the phone camera to capture photo in that area.

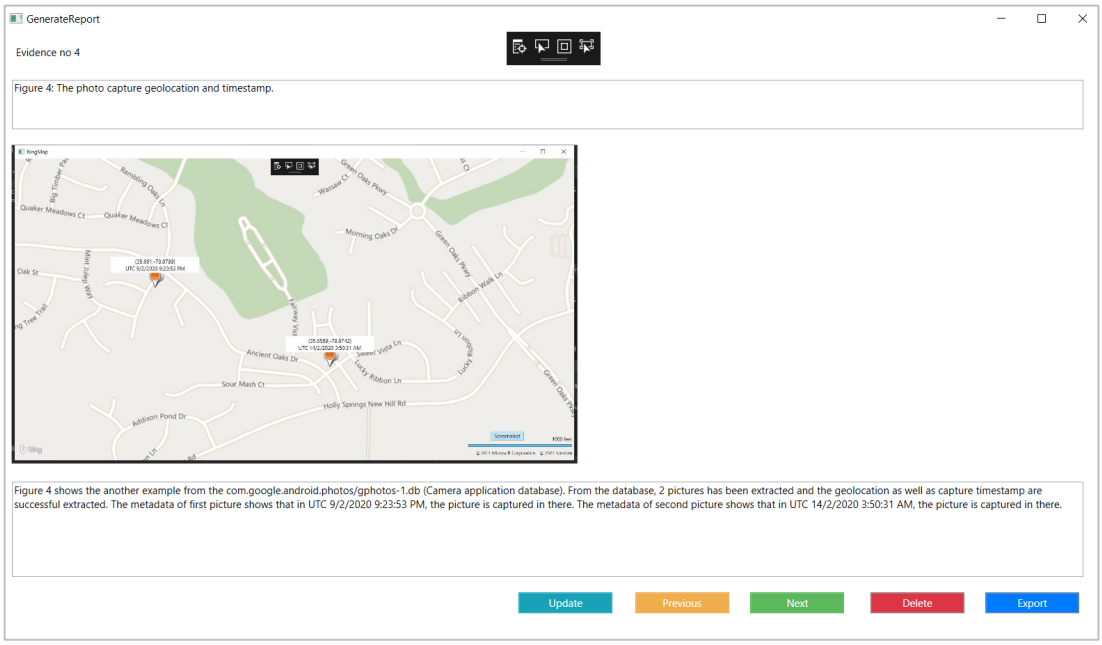


Figure 8: Reporting interface of photo capture geolocation and timestamp [25].

4.4 Testing

The system testing is the mandatory phase to implement when the system is completed. Therefore, there are testing cases that have been designed to determine whether all of the functionality of the tool is done.

This section will discuss the user acceptance test. The user acceptance test was carried out with 20 potential users of the system which are the third-year students of Bachelor of Computer Science. As the tool has the requirement that the tester need to have some digital forensic knowledge, the students who have taken the Digital Forensics course would be the target respondents.

Table 5: Testing table for Visualization Module.

Test Case	Expected Output	Actual Output
Display data	Display the data with correctly	Display the data with correctly
Adjust the scale	Able to adjust the scale	Able to adjust the scale
Screenshot Module	Able to screenshot the current chart and save as files.	Able to screenshot the current chart and save as files.
Select Data	Able to select data and shows the data in details.	Able to select data and shows the data in details.

From Table 5, the test cases of the visualization interface are all passed. The visualization part is mainly to test whether the proposed tool is able to process data and interact with the data visualization framework to produce correct output. The same results also occurred on the other test cases.

The user review discusses the opinion of users to the proposed tool. There are two statements that will be asked for the overall review from respondents.

Table 6: Testing table for Visualization Module.

No	Description	1	2	3	4	5
1	The data visualization of the proposed tool can facilitate the investigator.	0	3	5	10	2
2	The data visualization to the different charts of the proposed tool could suit different requirements of forensic analysis.	0	2	3	15	0

From Table 6, there are 2 respondents (10%) fully agreed with the statement that data visualization of the proposed tool can facilitate the investigator while 10 respondents (50%) agree with the statements. There are 5 respondents (25%) choosing neutral with the statements and 2 respondents (10%) did not agree with the statement.

For the second statement, there are 15 respondents (75%) who fully agree while there are 3 respondents neither agree nor disagree with the statements. Besides, there are 2 respondents who disagree with the second statement.

5. Conclusion

In this study, the design and development of MF Visualizer and testing results are discussed. The MF Visualizer can extract, analyze, visualize and report the image source of Android data partition. Therefore, objectives of this study are successfully achieved. The MF Visualizer could facilitate the forensic analysis of the crimes.

There are also limitations in this study. First limitation of MF Visualizer is only being able to visualize data in a few types of charts. Second limitation is that the chart of MF Visualizer is lacks of interactive control.

For future work, there are a few of improvements can be considered:

- The system can include more options of visualization models.
- The system can include more interactive control for more functionalities.
- Involve more respondents, especially digital forensics partitioners to get more useful feedback.

Acknowledgement

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support and encouragement throughout the process of conducting this study.

References

[1] C. Tassone, B. Martini, and K.-K. Choo, "Chapter 11 - Forensic visualization: survey and future research directions," in Contemporary Digital Forensic Investigations of Cloud and Mobile Applications, Elsevier, 2017, pp. 163–184.

[2] A. Dowling, "Digital forensics: A demonstration of the effectiveness of the sleuth kit and autopsy forensic browser," Master dissertation, University of Otago, 2006.

[3] T. Vidas, C. Zhang, and N. Christin, "Toward a general collection methodology for Android devices," Digit. Investig., vol. 8, pp. S14--S24, 2011.

[4] Y. M. Hasheem and K. M. M. Ahmad, "Mobile Forensic Triage for Damaged Phones Using M_Triage,", PhD dissertation, Universiti Tun Hussein Onn Malaysia, 2016.

[5] F. Carbone, Computer forensics with FTK. Packt Publishing Ltd, 2014.

[6] A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone, Handbook of applied cryptography. CRC press, 2018.

[7] H. Wijayanto, I. Riadi, and Y. Prayudi, "Encryption EXIF metadata for protection photographic image of copyright piracy," Int. J. Res. Comput. Commun. Technol., vol. 5, no. 5, pp. 237–242, May. 2016.

[8] P. Feng, Q. Li, P. Zhang, and Z. Chen, "Logical acquisition method based on data migration for Android mobile devices," Digit. Investig., vol. 26, pp. 55–62, May. 2018.

[9] Q. Do, B. Martini, and K.-K. R. Choo, "Enforcing file system permissions on android external storage: Android file system permissions (afp) prototype and owncloud," in 2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications, pp. 949–954, Sept. 2014.

[10] H. Shidek, N. Cahyani, and A. A. Wardana, "WhatsApp Chat Visualizer: A Visualization of WhatsApp Messenger's Artifact Using the Timeline Method," Int. J. Inf. Commun. Technol., vol. 6, no. 1, pp. 1–9, Jun. 2020.

[11] M. Yates, "Practical investigations of digital forensics tools for mobile devices," in 2010 information security curriculum development conference, pp. 156–162, Jan. 2010.

[12] N. Scrivens and X. Lin, "Android digital forensics: data, extraction and analysis," in Proceedings of the ACM Turing 50th Celebration Conference-China, pp. 1–10, May. 2017.

[13] J. Lessard and G. Kessler, "Android Forensics: Simplifying Cell Phone Examinations.," Small Scale Digital Device Forensics Journal, Jan. 2010.

[14] C. F. R. Tassone, B. Martini, and K.-K. R. Choo, "Visualizing digital forensic datasets: a proof of concept," J. Forensic Sci., vol. 62, no. 5, pp. 1197–1204, Feb. 2017.

[15] I. V Kotenko, M. Kolomeets, A. Chechulin, and Y. Chevalier, "A visual analytics approach for the cyber forensics based on different views of the network traffic.," J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl., vol. 9, no. 2, pp. 57–73, Jun. 2018.

[16] J. Oetting, "Data visualization 101: how to choose the right chart or graph for your data," 2016. [Online]. Available: https//blog. hubspot. com/marketing/datavisualization-choosing-chart. [Accessed Jul. 15, 2021].

[17] S. Lohmann, F. Heimerl, F. Bopp, M. Burch, and T. Ertl, "Concentri cloud: Word cloud visualization for multiple text documents," in 2015 19th International Conference on Information Visualisation, 2015, pp. 114–120, doi: 10.1109/iV.2015.30.

[18] B. Martini, Q. Do, and K.-K. R. Choo, "Mobile cloud forensics: An analysis of seven popular Android apps," The Cloud Security Ecosystem, 2015, doi:10.1016/B978-0-12-801595-7.00015-X.

[19] J. Choi and S. Lee, "A study of user relationships in smartphone forensics," Multimed. Tools Appl., vol. 75, no. 22, pp. 14971–14983, Nov. 2016.

[20] N. A. Aziz, F. Mokhti, and M. N. M. Nozri, "Mobile Device Forensics: Extracting and Analysing Data from an Android-Based Smartphone," in 2015 Fourth International Conference on Cyber Security, Cyber Warfare, and Digital Forensic (CyberSec), pp. 123–128, Oct. 2015.

[21] A. K. Agrawal, A. Sharma, and P. Khatri, "Digital Forensic Analysis of Facebook App in Virtual Environment," in 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom), 2019. pp. 660–664.

[22] T.-I. Kitsaki, A. Angelogianni, C. Ntantogian, and C. Xenakis, "A forensic investigation of Android mobile applications," in Proceedings of the 22nd Pan-Hellenic Conference on Informatics, pp. 58–63, Nov. 2015.

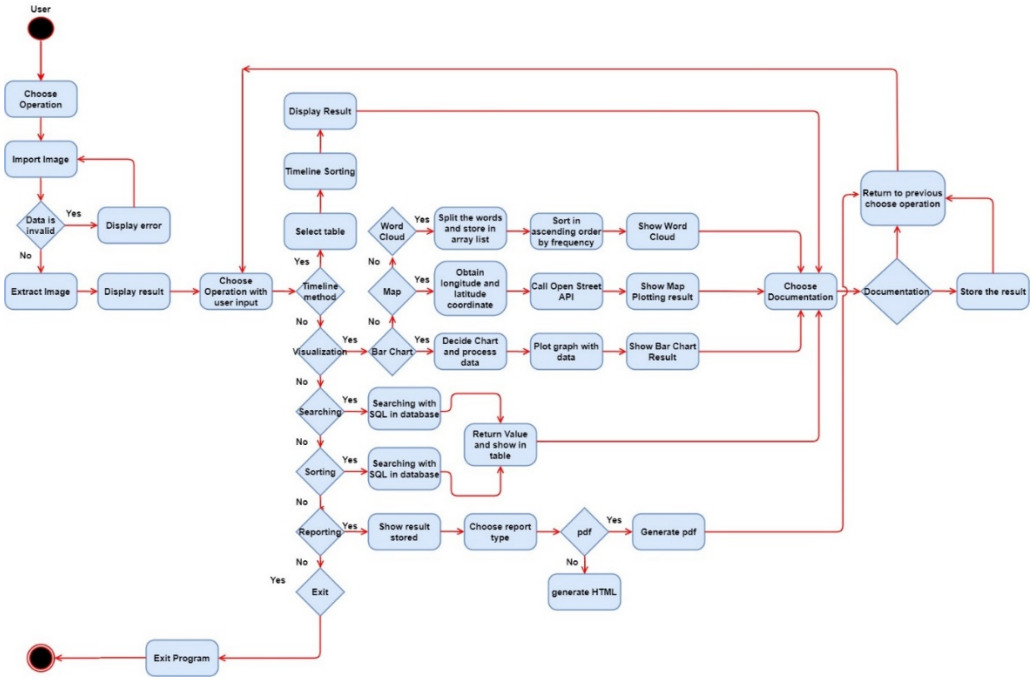
[23] B. Martini, Q. Do, and K.-K. R. Choo, "Conceptual evidence collection and analysis methodology for Android devices," The Cloud Security Ecosystem: Technical, Legal, Business and Management Issues, 2015, doi:10.1016/B978-0-12-801595-7.00014-8.

[24] B. Bruegge and A. A. Dutoit, Object-oriented software engineering; conquering complex and changing systems. Prentice Hall PTR, 1999.

[25] J. Hickman, "Android 10 Image Source," Digital Corpora, 2020. [Online]. Available: https://digitalcorpora.org/corpora/cell-phones/android-10/. [Accessed Jun. 08, 2021].

[26] K. Kucher and A. Kerren, "Text visualization techniques: Taxonomy, visual survey, and community insights," in 2015 IEEE Pacific visualization symposium (pacificVis), 2015, pp. 117–121, doi: 10.1109/PACIFICVIS.2015.7156366.

Appendix A (Activity Diagram)



Appendix B (Use Case Diagram)