Image GPS Analyzer: Using Graphic File Metadata to Plot Geographic Timeline

Andrew Kirby
Department of Computer Science
Sam Houston State University
Huntsville, Texas 77340, USA
amk073@shsu.edu

Dr. Umit Karabiyik
Department of Computer Science
Sam Houston State University
Huntsville, Texas 77340, USA
umit@shsu.edu

***Abstract* – Graphics file formats, such as JPEG, contain information about the camera or device that created it as well as the information about when and where the picture was taken. Investigators may come into possession of these files in multiple ways. A few examples might be mobile data acquisition of a phone, directly off of a SIM card taken from a camera or other device, or from forensically recovering the image from a deleted file in memory on a computer. The images themselves may not be incriminating, however the metadata from those images may help investigators build a timeline of events in the context of a case. This study introduces the Image GPS Analyzer, a forensic modeling application that examines the GPS data from EXIF information and plots each photo as a point on a map in chronological order.**

Index Terms* – *EXIF, GPS, Digital forensics

I. Introduction

In the field of digital forensics (DF), mobile forensics (MF) has become increasingly important due to the rise in prominence of the smart phone. Over 1.5 billion smartphone units shipped in 2017, almost triple the amount from 2011 (See Figure 1) [1].

Smartphones are complex devices that act not only as a cell phone but are actually mini-computers that store a wealth of information about its owner, and information about anyone it's owner interacts with on a day to day basis. This allows investigators to glean a large amount of data related to a network of people.

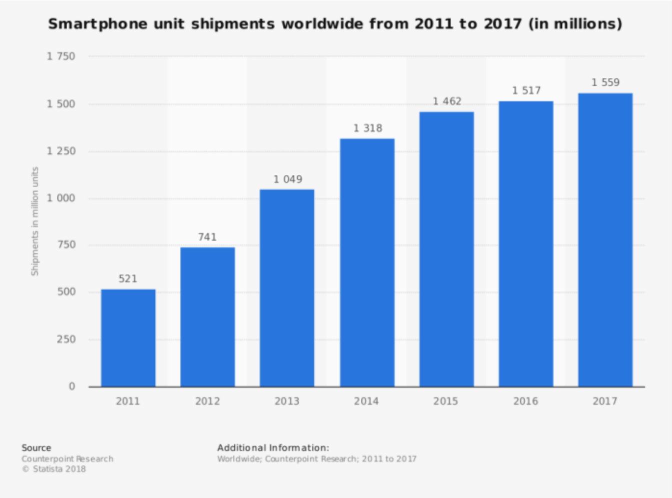


Figure 1:Smartphone Units Shipped Worldwide 2011-2017

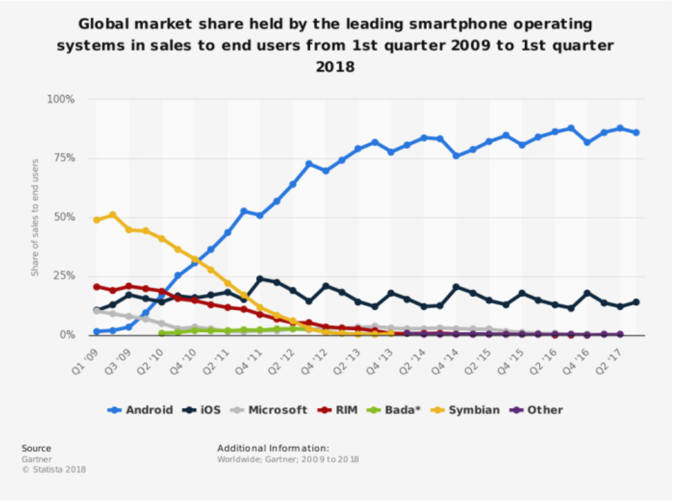
While there are many methods for data acquisition, depending on the make, model and firmware version of the device, there are not as many methods for presentation of the data [2]. Forensically sound presentation tools can serve a few purposes: displaying data to law enforcement personnel in such a way to aid investigations, creating a presentation that is admissible in court as well as intuitive to a typical jury, and creating presentations that will be able to be stored for future litigation and appeals.

EXIF stands for Exchangeable Image File Format, and stores metadata related to the image file [3]. EXIF is managed by the Japan Electronics and Information Technology Industries Association (JEITA), which seeks to advance the electronics and information technology industries [4].

More specifically, investigators are able to acquire graphic files, such as JPEG or TIFF formats, that contain date, time and geographic information in the metadata that can help establish a geographical timeline, which can support a prosecutor's case or create reasonable doubt for the defense. This geographical data is commonly referred to as a geotag. Geotagging is the process of adding geographical data to images or other files as metadata [5]. Geotagging can be configured in device settings to automatically attach it to every photo taken by the device. In short, geographical data from photos may be more important than the images themselves.

To address the concerns of presentation of image data from multiple sources, we propose the Image GPS Analyzer. There are three major goals of the Image GPS Analyzer. First, implement a solution which will read, compile and facilitate analysis of EXIF information that is stored as a part of several commonly used image formats (currently GIF, JPEG, PNG, and TIFF) [6]. Second, this solution must be flexible enough to work with image files generated from any device platform. And finally, this solution must be able to be extensible to integrate with other commonly used industry solutions.

There are two main business cases for this research. First, the Image GPS Analyzer could be used as a part of active investigations if investigators are attempting to track a person's movements over a period of time, or to establish a timeline. Secondly, the Image GPS Analyzer would aid in the presentation of electronic evidence in court proceedings. Jury members may not all be technically savvy and therefore it is important to present information to them in a way that allows for them to intuitively understand the value of the evidence.


Global market share held by the leading smartphone operating systems in sales to end users from 1st quarter 2009 to 1st quarter 2018

Source
Gartner
© Statista 2018

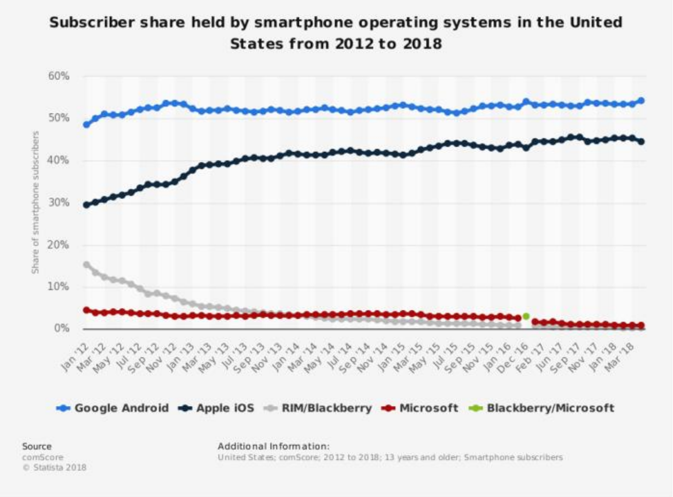
Additional Information:
Worldwide; Gartner; 2009 to 2018

Figure 2: Global Market Share by Mobile Operating System

II. Related Work

Work related to forensic analysis of image data from smartphones has shown that a user may not be required to explicitly take a photo for it to be stored in memory. Saltaformaggio et al. presented a method called Visual Content Recovery (VCR) which allows the investigator to recover and view images that were stored in application memory. Applications that have access to the camera may store images from the camera view screen without the user taking a photo [7].

Another study created a geographical timeline presentation that plots events on a map. Kasiaras et al. developed a program called the Android Forensic Data Analyzer (AFDA) that compiled application events with geotags from Android phones and plotted the events on a geographical map to display a physical timeline showing the likely path that the person took from point to point [8]. While ADFA creates a map presentation in the same fashion that is intended with this study, it is limited to Android implementations.


Subscriber share held by smartphone operating systems in the United States from 2012 to 2018

Source
comScore
© Statista 2018

Additional Information:
United States; comScore; 2012 to 2018; 13 years and older; Smartphone subscribers

Figure 3: US Market Share by Mobile Operating System

ADFA is a useful tool internationally, since Android has a vast majority of market share globally, but domestically in the United States there is a more even distribution of iPhone and Android users. In the global market in Q1 2018, Android held 85.9% with iOS trailing at 14.1%. In the U.S. the figure is much more even with Android holding the lead at 54.3% and iOS at 44.6% (See Figure 2 for Global and Figure 3 for U.S.) [9, 10].

Park et al. created a method to retroactively assign geotags to photos that don't have geographical metadata already embedded in the image [11]. This method uses known landmarks in the photo to triangulate the location of the camera. This may be useful in an investigation if the suspect has disabled geotagging in their device settings.

Mahdian and Saic summarized that using a single technique is not sufficient to determine whether a jpeg image has been modified. They used multiple techniques (EXIF data analysis and JPEG Double Quantization) in conjunction to increase the number of modified files that were found during the process [12].

III. Methodology

The Image GPS Analyzer is a desktop application intended to be run on a forensic computer. It is written in C#.NET, an object-oriented programming language developed by Microsoft [13]. The program is separated into modules. The graphical user interface (GUI) is encapsulated in ImageGPSAnalyzer.exe, while two custom assemblies are used for parsing the image data and exporting the case file reports (ImageAnalyzer.dll and ImageGPSPackager.dll respectively). In the forthcoming sections, we will discuss in depth the operation of each of the assemblies. The full source code for the Image GPS Analyzer can be found on GitHub (https://github.com/akirby/ImageGpsAnalyzer) [14].

A. Image GPS Analyzer Front End

The Image GPS Analyzer GUI consists of a Windows Presentation Foundation (WPF) form [15]. In Figure 4, a screenshot of the application is shown. The major components are a data grid that displays a list of all the images selected, an image preview control that displays a thumbnail of the image and a world map control which uses the Bing Maps API with a WPF control produced by Microsoft [16]. Working with the Bing Maps API requires an internet connection, which could be a hurdle for some if their forensic machine is not connected to the internet. Due to time constraints, no other map control or API was evaluated, but this aspect may need to be revisited. Furthermore, the Bing Maps API has free developer licensing, but distributing the application with this license would be a violation of the terms, and therefore licensing for deploying the solution would likely require a paid subscription to the Bing Maps API service.

To begin analysis of a group of images, the images must first be collected in a single directory on the forensic machine. The user will select "New" from the menu and is prompted to enter the case number and select the directory in which the image files are located.

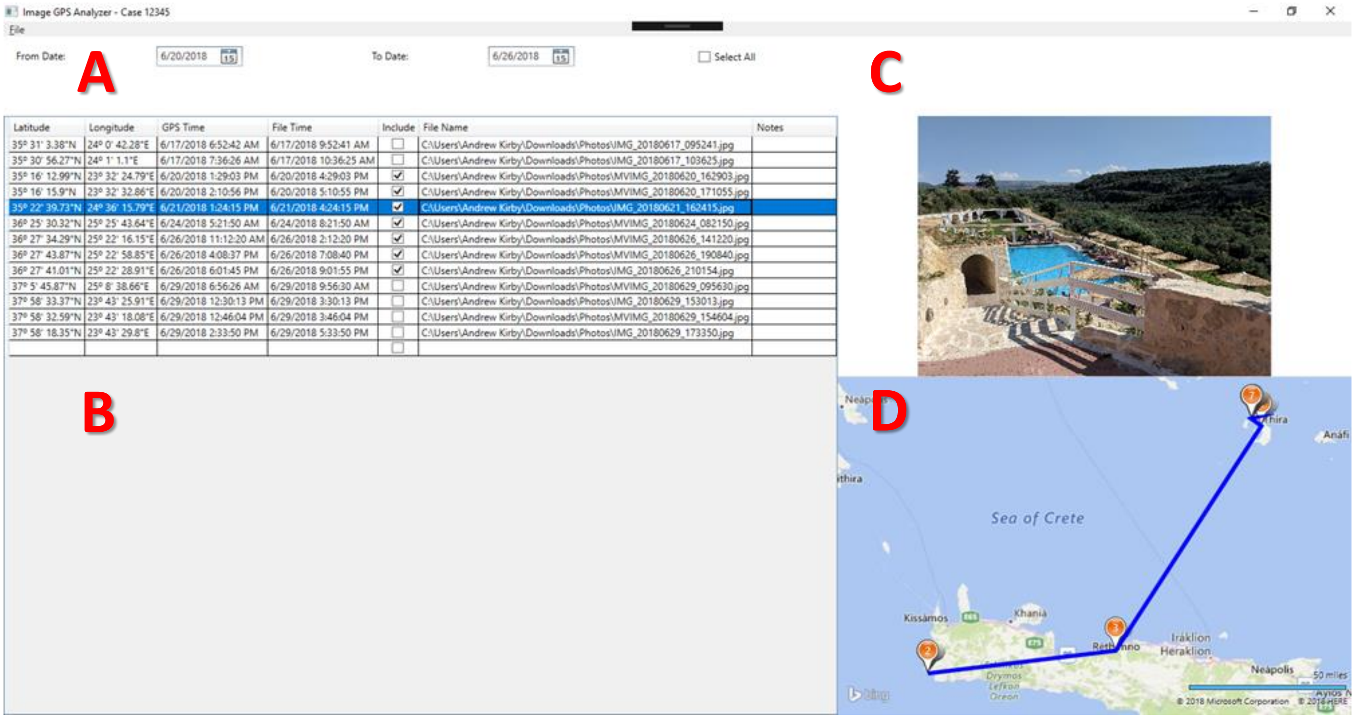


Image GPS Analyzer - Case 12345

File

From Date: 6/20/2018 To Date: 6/26/2018 □ Select All

Latitude	Longitude	GPS Time	File Time	Include	File Name	Notes
35° 31' 3.38"N	24° 0' 42.28"E	6/17/2018 6:52:42 AM	6/17/2018 9:52:41 AM	□	C:\Users\Andrew Kirby\Downloads\Photos\IMG_20180617_095241.jpg	
35° 30' 56.27"N	24° 1' 1.1"E	6/17/2018 7:36:26 AM	6/17/2018 10:36:25 AM	□	C:\Users\Andrew Kirby\Downloads\Photos\IMG_20180617_103625.jpg	
35° 16' 12.99"N	23° 32' 24.79"E	6/20/2018 1:29:03 PM	6/20/2018 4:29:03 PM	☑	C:\Users\Andrew Kirby\Downloads\Photos\MVIMG_20180620_162903.jpg	
35° 16' 15.9"N	23° 32' 32.86"E	6/20/2018 2:10:56 PM	6/20/2018 5:10:55 PM	☑	C:\Users\Andrew Kirby\Downloads\Photos\MVIMG_20180620_171055.jpg	
35° 22' 39.73"N	24° 36' 15.79"E	6/21/2018 1:24:15 PM	6/21/2018 4:24:15 PM	☑	C:\Users\Andrew Kirby\Downloads\Photos\IMG_20180621_162415.jpg	
36° 25' 30.32"N	25° 25' 43.64"E	6/24/2018 5:21:50 AM	6/24/2018 8:21:50 AM	☑	C:\Users\Andrew Kirby\Downloads\Photos\MVIMG_20180624_082150.jpg	
36° 27' 34.29"N	25° 22' 16.15"E	6/26/2018 11:12:20 AM	6/26/2018 2:12:20 PM	☑	C:\Users\Andrew Kirby\Downloads\Photos\MVIMG_20180626_141220.jpg	
36° 27' 43.87"N	25° 22' 58.85"E	6/26/2018 4:08:37 PM	6/26/2018 7:08:40 PM	☑	C:\Users\Andrew Kirby\Downloads\Photos\MVIMG_20180626_190840.jpg	
36° 27' 41.01"N	25° 22' 28.91"E	6/26/2018 6:01:45 PM	6/26/2018 9:01:55 PM	☑	C:\Users\Andrew Kirby\Downloads\Photos\IMG_20180626_210154.jpg	
37° 5' 45.87"N	25° 8' 38.66"E	6/29/2018 6:56:26 AM	6/29/2018 9:56:30 AM	□	C:\Users\Andrew Kirby\Downloads\Photos\MVIMG_20180629_095630.jpg	
37° 58' 33.37"N	23° 43' 25.91"E	6/29/2018 12:30:13 PM	6/29/2018 3:30:13 PM	□	C:\Users\Andrew Kirby\Downloads\Photos\IMG_20180629_153013.jpg	
37° 58' 32.59"N	23° 43' 18.08"E	6/29/2018 12:46:04 PM	6/29/2018 3:46:04 PM	□	C:\Users\Andrew Kirby\Downloads\Photos\MVIMG_20180629_154604.jpg	
37° 58' 18.35"N	23° 43' 29.8"E	6/29/2018 2:33:50 PM	6/29/2018 5:33:50 PM	□	C:\Users\Andrew Kirby\Downloads\Photos\IMG_20180629_173350.jpg	
				□		

Figure 4: Screenshot of Image GPS Analyzer (A - Filter Controls, B - Data grid, C - Image Preview, D - Bing Map Control)

Once the image data has been parsed and loaded, the user can filter the images plotted on the map by either checking the "Include" checkbox in the rows of the data grid, or by changing the date boundaries. This gives a user flexibility to start with a large set of files, but narrow to only relevant images or dates.

B. *Parsing Image Data*

EXIF information stores GPS data in a series of byte pairs that represent rational numbers [17]. When using the .NET Framework to access this information, this information is accessible as an item in a collection of properties of the image. These items are identified using the tags from the Property Item Descriptions [18]. Not all properties will be represented in every image file, but the properties used by the ImageAnalyzer assembly are listed in Table I.

TABLE I
IMAGE PROPERTY TAGS

Tag Names (Comma Separated)	Descriptions
PropertyTagGpsLatitudeRef, PropertyTagGpsLongitudeRef	Represented by a 2-byte value indicating the hemisphere of the relative cardinal directions.
PropertyTagGpsLatitude, PropertyTagGpsLongitude, PropertyTagGpsGpsTime	Represented by 3 pairs of 32-bit integers, for degrees, minutes and seconds.
PropertyTagGpsGpsTime	Represented by 3 pairs of 32-bit integers, for hours, minutes and seconds. Time is local time of the GPS location.
PropertyTagDateTime	UTC time stored as ASCII values and formatted as "yyyy:MM:dd H:mm:ss"

For example, when parsing the bytes for PropertyTagGpsLatitude in Table II, the input is broken down into 3 pairs of 4-byte values. Those 4-byte values are then converted to 32-bit integers and each pair of 32-bit integers is divided with the first integer being the numerator and the second integer being the denominator. In the Latitude and Longitude properties, the first pair represents degrees, the second pair represents Minutes and the last pair represents seconds.

$$\begin{gathered}\text{Degrees} = 35\,/\,1 = 35^\circ \\ \text{Minutes} = 31/1 = 31' \\ \text{Seconds} = 338/100 = 3.38'' \\ \mathit{Latitude} = 35^\circ\,31'\,3.38''\end{gathered}$$

With the context of the PropertyTagGpsLatitudeRef from this image, we are able to determine that this is in the Northern hemisphere as well.

TABLE II
24 BYTE VALUE FOR PROPERTYTAGGPSLATITUDE

35	0	0	0	1	0	0	0
31	0	0	0	1	0	0	0
82	1	0	0	100	0	0	0

C. *Exporting Data to Case Files*

Another important part of the Image GPS Analyzer is its ability to save the analysis that the user has done to files that can be read later. One benefit of is feature is that the analysis can be preserved for future reference. If this analysis were to ever be used in a court case or law enforcement investigation, then it would be subject to procedures for chain of custody of electronic evidence as well as the scrutiny of judicial review.

The Image GPS Analyzer saves the data to two file formats as of this writing (XML and DOCX). The XML file is based on a data model that was created for the Image GPS Analyzer. This file can be reloaded into the Image GPS Analyzer to change the selection of images plotted on the map or for review.

The DOCX file is saved using OpenXML formatting through the Open XML SDK, version 2.8, by Microsoft [19]. The word document export contains a table with the information related to each image that was included in the analysis as well as a copy of the map as a PNG image embedded into the document. The Bing Maps API does not provide a method of exporting the map to an image, so the Image GPS Analyzer creates a bitmap image from the GUI map control and then saves that image to the file.

IV. Experiment

A. *Setup*

To test the Image GPS Analyzer, images taken on an iPhone 8+, a Google Pixel 2, and a Samsung Galaxy S8 were compiled into a single folder. All these files are JPEG format, which is a file format that is supported by EXIF.

B. *Results*

On the start-up of the Image GPS Analyzer, the application prompted for a case number. For the test, the case number 12345 is used. Then the application prompts for a directory that contains the image files. After selecting the folder created in the setup step, the Image GPS Analyzer loads all the GPS data into the data grid and automatically selects all rows to be represented as points on the map. The original data set contains images from 6/17/2018 6:52:42 UTC to 6/29/2018 14:33:50 UTC

After the initial load, we saved the case file reports, which exports the DOCX and XML file. The DOCX contains a table with all the images listed and the map image showing all 13 points on the map. The XML also has all the images listed. The XML can be used later to reopen the case and make changes. Figure 5 depicts the map after the initial load step. This map is the same map that was exported to the DOCX file.

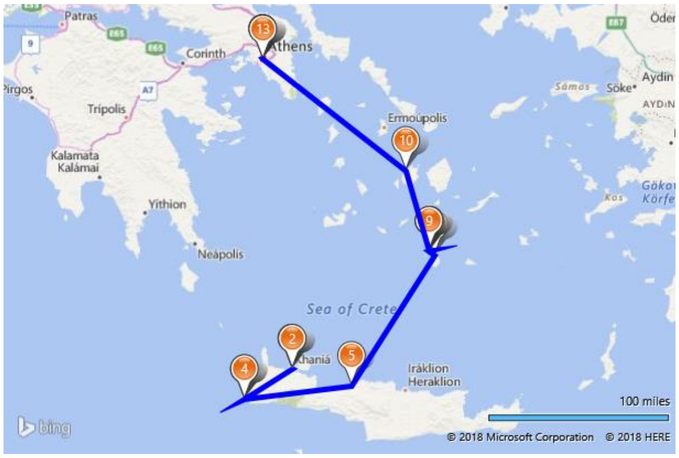
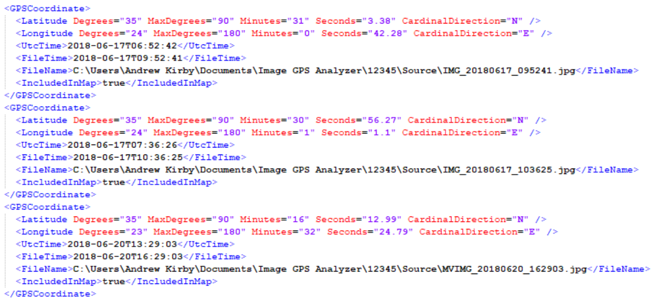


Figure 6: Initial Map points after test load



```
<GPSCoordinate>
  <Latitude Degrees="35" MaxDegrees="90" Minutes="31" Seconds="3.38" CardinalDirection="N" />
  <Longitude Degrees="24" MaxDegrees="180" Minutes="0" Seconds="42.28" CardinalDirection="E" />
  <UtcTime>2018-06-17T06:52:42</UtcTime>
  <FileTime>2018-06-17T09:52:41</FileTime>
  <FileName>C:\Users\Andrew Kirby\Documents\Image GPS Analyzer\12345\Source\IMG_20180617_095241.jpg</FileName>
  <IncludedInMap>true</IncludedInMap>
</GPSCoordinate>
<GPSCoordinate>
  <Latitude Degrees="35" MaxDegrees="90" Minutes="30" Seconds="56.27" CardinalDirection="N" />
  <Longitude Degrees="24" MaxDegrees="180" Minutes="1" Seconds="1.1" CardinalDirection="E" />
  <UtcTime>2018-06-17T07:36:26</UtcTime>
  <FileTime>2018-06-17T10:36:25</FileTime>
  <FileName>C:\Users\Andrew Kirby\Documents\Image GPS Analyzer\12345\Source\IMG_20180617_103625.jpg</FileName>
  <IncludedInMap>true</IncludedInMap>
</GPSCoordinate>
<GPSCoordinate>
  <Latitude Degrees="35" MaxDegrees="90" Minutes="16" Seconds="12.99" CardinalDirection="N" />
  <Longitude Degrees="23" MaxDegrees="180" Minutes="32" Seconds="24.79" CardinalDirection="E" />
  <UtcTime>2018-06-20T13:29:03</UtcTime>
  <FileTime>2018-06-20T16:29:03</FileTime>
  <FileName>C:\Users\Andrew Kirby\Documents\Image GPS Analyzer\12345\Source\MVIMG_20180620_162903.jpg</FileName>
  <IncludedInMap>true</IncludedInMap>
</GPSCoordinate>
```

Figure 7: Example of XML Nodes in exported file

To test the filtering options, we changed the From Date to 6/20/2018 and the To Date to 6/26/2018. This filters the list down to seven images. This is represented in the data grid by checked checkboxes in the rows for the images that are selected. The user may also manually include or exclude an image through the use of the checkboxes. After the filter step is complete we again save the case to DOCX Format and XML format. Figure 6 shows the updated map after the filtering step.

The XML file is slightly different from the DOCX in terms of the data contained. The DOCX is intended for presentation purposes or offline use, it only contains the image data from images included in the report after the filtering applied. The XML, however, includes all files in the dataset, whether they are included in the report or not. In the XML, each record has an attribute to indicate whether the image is included in the report. A sample of the XML nodes is shown in Figure 7. Also, in Figure 8, is a screenshot of the format of the DOCX file.

IV. Conclusion

The Image GPS Analyzer is a desktop application that takes as input a collection of images in a single directory and outputs analysis on when and where the pictures were taken. Since the Image GPS Analyzer is a platform agnostic tool for investigators to use, they will be able to analyze data from multiple sources in an investigation. During this experiment the

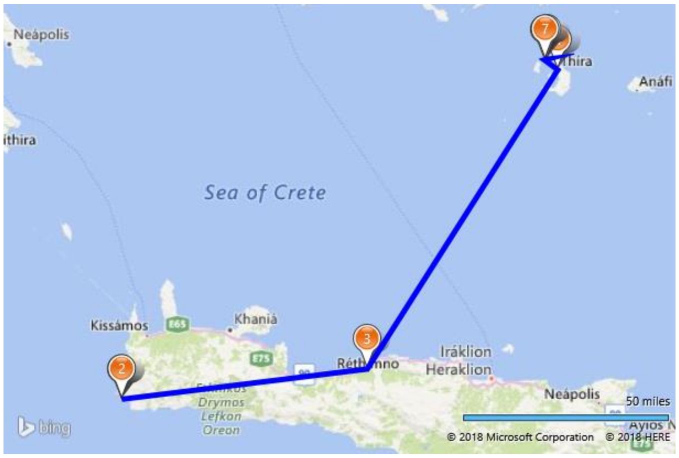
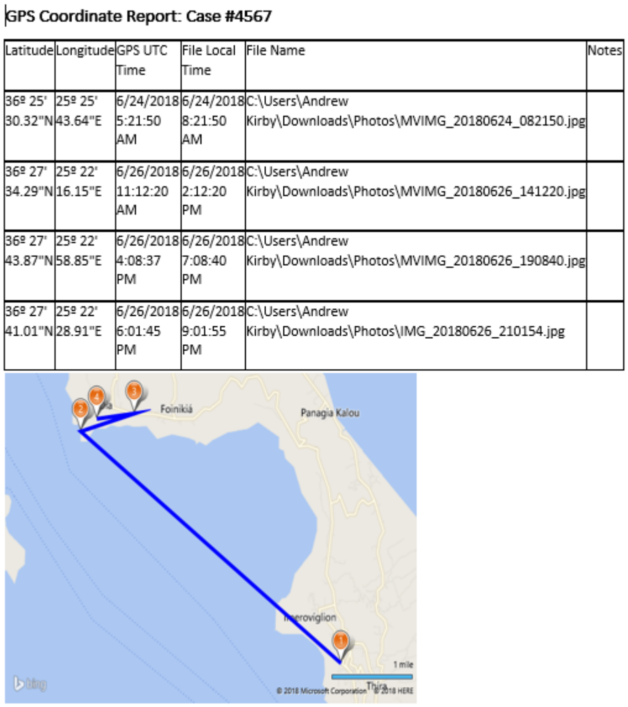


Figure 5: Map points after filtering by dates

Image GPS Analyzer compiled data from images taken on 3 different devices.

It may help investigators to find patterns in the movement of a suspect, or even to track movements of larger criminal organizations involved with human trafficking, drug trade and other major organized crimes.

The Image GPS Analyzer accomplishes the goals originally set out as defined in the introduction. Through parsing the image property tag data, also more commonly known as EXIF data, the Image GPS Analyzer compiles and facilitates the analysis of GPS data from image files. It is modular, isolating main functions into separate assemblies, therefore allowing it the flexiblity to integrate with other solutions. By using the EXIF data, rather than depending on OS specific information, the Image GPS Analyzer is able to analyze images from every platform.



GPS Coordinate Report: Case #4567

Latitude	Longitude	GPS UTC Time	File Local Time	File Name	Notes
36º 25' 30.32"N	25º 25' 43.64"E	6/24/2018 5:21:50 AM	6/24/2018 8:21:50 AM	C:\Users\Andrew Kirby\Downloads\Photos\MVIMG_20180624_082150.jpg	
36º 27' 34.29"N	25º 22' 16.15"E	6/26/2018 11:12:20 AM	6/26/2018 2:12:20 PM	C:\Users\Andrew Kirby\Downloads\Photos\MVIMG_20180626_141220.jpg	
36º 27' 43.87"N	25º 22' 58.85"E	6/26/2018 4:08:37 PM	6/26/2018 7:08:40 PM	C:\Users\Andrew Kirby\Downloads\Photos\MVIMG_20180626_190840.jpg	
36º 27' 41.01"N	25º 22' 28.91"E	6/26/2018 6:01:45 PM	6/26/2018 9:01:55 PM	C:\Users\Andrew Kirby\Downloads\Photos\IMG_20180626_210154.jpg	

Figure 8: Screenshot of DOCX Output

A. Future Work

This application has not reached its full potential. First, enhancements can be made to add export types of different formats. DOCX is not best format because of how easy it is to edit the file, but it does accomplish the initial goal of making it easy to present. Encryption should also be added to the exported files so in order to prevent tampering. Additionally, the software could be updated to directly integrate with other forensic software.

A major dependency that the Image GPS Analyzer has is on the Bing Maps API. This solution does not work offline since the API requires verification with the server using a device key. The solution should ideally work offline.

Another shortcoming of the current solution is that the line between each of the points on the map does not follow the roads on the map, to which people have grown accustomed. One possible enhancement would be to add a way to snap the line to a possible route between the points. This is possibly doable in the Bing Maps API, but if an alternative API is found based on the offline requirement then this would be dependent on that solution.

Another valuable enhancement would require further analysis of the work done by Saltaformaggio et al to integrate the Visual Content Recovery data into the Image GPS Analyzer [7]. It is not clear at this time whether the digital artifacts recovered through VCR would contain EXIF information or simply the image bitmap.

References

[1] "Global smartphone shipments 2011-2017 | Statistic," *Statista*. [Online]. Available: https://www.statista.com/statistics/687475/global-smartphone-shipments/. [Accessed: 24-Jul-2018].

[2] K. Barmpatsalou, T. Cruz, E. Monteiro, and P. Simoes, "Current and Future Trends in Mobile Device Forensics: A Survey," *ACM Comput Surv*, vol. 51, no. 3, pp. 46:1–46:31, May 2018.

[3] "EXIF.org – EXIF and related resources." .

[4] "JEITA / About JEITA / What is JEITA?" [Online]. Available: https://www.jeita.or.jp/english/about/what/index.htm. [Accessed: 25-Jul-2018].

[5] Jiebo Luo1, D. Joshi dhiraj. joshi@kodak. co., Jie Yu1, and A. Gallagher, "Geotagging in multimedia and computer vision-a survey," *Multimed. Tools Appl.*, vol. 51, no. 1, pp. 187–211, Jan. 2011.

[6] "Geotagging Images – EXIF.org." .

[7] B. Saltaformaggio, R. Bhatia, Z. Gu, X. Zhang, and D. Xu, "VCR: App-Agnostic Recovery of Photographic Evidence from Android Device Memory Images," in *Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security*, New York, NY, USA, 2015, pp. 146–157.

[8] D. Kasiaras, T. Zafeiropoulos, N. Clarke, and G. Kambourakis, "Android Forensics: Correlation Analysis," in *2014 9th International Conference for Internet Technology and Secured Transactions, ICITST 2014*, 2014.

[9] "Mobile OS market share 2018 | Statista." [Online]. Available: https://www.statista.com/statistics/266136/global-market-share-held-by-smartphone-operating-systems/. [Accessed: 24-Jul-2018].

[10] "Mobile OS market share in the U.S. 2018," *Statista*. [Online]. Available: https://www.statista.com/statistics/266572/market-share-held-by-smartphone-platforms-in-the-united-states/. [Accessed: 24-Jul-2018].

[11] M. Park, Y. Chen, and K. Shafique, "Tag Configuration Matcher for Geo-tagging," in *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, New York, NY, USA, 2013, pp. 384–387.

[12] B. Mahdian and S. Saic, "Image Tampering Detection Using Methods Based on JPEG Compression Artifacts: A Real-life Experiment," in *Proceedings of the 4th International Symposium on Applied Sciences in Biomedical and Communication Technologies*, New York, NY, USA, 2011, pp. 176:1–176:5.

[13] B. Wagner, "Introduction to the C# Language and the .NET Framework." [Online]. Available: https://docs.microsoft.com/en-us/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework. [Accessed: 24-Jul-2018].

[14] A. Kirby, *ImageGpsAnalyzer*. 2018.

[15] "Introduction to WPF in Visual Studio." [Online]. Available: https://docs.microsoft.com/en-us/dotnet/framework/wpf/getting-started/introduction-to-wpf-in-vs. [Accessed: 25-Jul-2018].

[16] "Bing Maps WPF Control." [Online]. Available: https://msdn.microsoft.com/en-us/library/hh750210.aspx. [Accessed: 25-Jul-2018].

[17] “How to: Read Image Metadata.” [Online]. Available: https://docs.microsoft.com/en-us/dotnet/framework/winforms/advanced/how-to-read-image-metadata. [Accessed: 24-Jul-2018].

[18] M. Satran, “Property Item Descriptions.” [Online]. Available: https://docs.microsoft.com/en-us/windows/desktop/gdiplus/-gdiplus-constant-property-item-descriptions. [Accessed: 25-Jul-2018].

[19] o365devx, “Getting started with the Open XML SDK 2.5 for Office</td>.” [Online]. Available: https://docs.microsoft.com/en-us/office/open-xml/getting-started. [Accessed: 25-Jul-2018].