A FORENSIC ANALYSIS OF DIGITAL IMAGE CHARACTERISTICS ASSOCIATED TO

FLICKR® AND GOOGLE PLUS®

by

MELISSA LAWSON

B.S., Metropolitan State College of Denver, 2000

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirement for the degree of

Master of Science

Recording Arts Program

2017

This thesis for the Master of Science degree by

Melissa Lawson

Has been approved for the

Recording Arts Program

by

Dr. Catalin Grigoras, Chair

Professor Jeff M. Smith

Dr. Marcus Rogers

November 30, 2017

Lawson, Melissa (M.S., Recording Arts Program)

A Forensic Analysis of Digital Image Characteristics Associated to Flickr® and Google Plus®

Thesis directed by Assistant Professor, Dr. Catalin Grigoras

ABSTRACT

This study explores the patterns and characteristics given to images after they have been uploaded and downloaded to and from Flickr and Google Plus. The examination not only tests the impact that a mobile phone and desktop have on images during the transference, but also explores different options to download protected images depending on the internet browser and social platform being used. It is discovered that certain methods of downloading protected images reveal varying results, some methods being better than others at retaining authentic information about the file. Largely, metadata in images downloaded from Flickr remain completely intact so long as account users do not protect their images. If images are protected, there are ways of saving them, however their metadata is completely stripped in the process. On the other hand, the capability of downloading protected images in their exact original pixel width and height is possible on Google Plus depending on the download method and internet browser being used. Furthermore, Google Plus preserves unique information embedded in the extension of image files. Social Media is universal and ever present in our daily lives which is how it has become an incredibly powerful tool to help build a case and why it is routinely used to gather evidence in investigations. By understanding the base behavior of these social media platforms, we can yield results to inform best practices for forensic investigations.

Approved:_____

Approved:_____

DEDICATION

I dedicate my thesis paper to my four-year old son. Mastery is self-improvement and personal growth -never stop learning and applying new skills. Be determined. Be self-disciplined. Be courageous. Be compassionate. Be great. I love you too much.

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

Engaging in social media websites is a common activity today for adults and kids around the globe. Usage is highly diverse ranging from favorite platforms such as Facebook, Twitter, Instagram and YouTube, and funneling down to sources that foster gaming, dating, network exchange, virtual worlds or tight-niche blogs. Hundreds of interactive networks are designed to offer communication and entertainment, while in their unique way, being a catchall location for anything that can be shared digitally. Undoubtedly, social Media has become a lively vessel that connects our personal and business lives to networking opportunities, meaningful engagement and pastimes for those willing to enter its uncharted territory. Nevertheless, forces as global as the social media galaxy inevitably begin to form two sides: the light side of the force and the dark side. While the good side entices many enthusiasts, it synchronously builds a world-wide pond for undesirables to fish and create problems. Online threats, stalking, cyberbullying, hacking and fraud, buying illegal things, and posting crimes are reasons why Digital Media experts, Intelligence Analysts and investigators rapidly face a barrage of crime from a new frontier [24].

Digital images and videos play a large role in the bottleneck of investigations. The ability to share images and live feed instantaneously through electronic devices by the masses, stirs up the perfect storm for investigators in search of evidence. When it comes to social media and the inability to prove every step taken to create electronically stored information -the question of authenticity rises. Authentication is key to getting social media evidence admitted in the courtroom [1].

When the integrity and preservation of digital images is not maintained, copyright matters also arise. For example, photographers who share their work on Flickr. Metadata is important in protecting copyright images online. One of many reasons why it is so important to understand the origins of digital media found on social media.

The data that lies behind digital images can provide a wealth of information in the analysis of an image and so can signs of compression. There are different types of information to pay attention to when analyzing an image. One is Metadata; a set of data that describes and gives information about another data [2]. Metadata in an image file, especially EXIF (Exchangeable Image File Format) [13] and XMP (Extensible Metadata Platform), can provide a lot of information about the user, like when a user creates a document, or changes it, or accesses it.

Metadata can be altered as images undergo processing and sharing. This includes uploading and downloading images to social sites. Modifications can provide clues in an investigation involving a potential suspect and their social network account, but can also limit what can be detected and complicate the pursuit. This research centers around alterations made images with Google Plus and Flickr and provides various methods and experimental results to distinguish changes or patterns relative to each social site, as well as electronic devices and internet browsers used. The paper is organized as follows: Related Works, Overview, Proposed Methodology, Results and Conclusion.

Related Works

Recent works have studied images on social networks. In Megan Pippin's research, "Digital Image Recompression Analysis: Facebook" (2016), it was discovered that when an image is pulled from Facebook, the type of device doing the uploading and downloading of the image impacts the patterns in metadata structure, quantization tables and JPEG quality. For

example, a mobile phone that downloads an image repeatedly revealed the following predictable behavior: baseline Discrete Cosine Transform (DCT) encoding, a specific set of quantization tables, and a metadata structure of 65 elements; images uploaded from mobile phone and downloaded to desktop showed progressive encoding and a metadata structure of 67 elements; images uploaded and downloaded to desktop computer were progressive encoded, variable quantization tables with average of 87 JPEG quality and metadata structure of 81 elements [3]. Regarding the outcome of mobile phone pictures and desktop pictures, it is also noted that each have specific width dimension cutoffs. Mobile phone being 1440 pixels and desktop 2048 pixels. Lastly, Facebook has a specific renaming process on images. An image that is given a filename on its way into the social site, is renames as it exits. After download, the image is named after a long string of numbers ending in either _n or _o. The _n is placed on images smaller than 1024 pixels, the _o on files larger than 1024 [3]. These are all good examples of how a social internet site effects images and also the sort of observations to take note of during this particular research.

Also, worth mentioning is Castiglione et al. (2011) analysis that tested the impact on images on three different social sites: Facebook, Badoo and Google Plus. The analysis confirms that JPEG quantization tables, pixel resolution and metadata are indeed altered in predictable ways and demonstrates that images are stripped of original metadata in the process of compression. Furthermore, it was discovered that Google Plus manages metadata differently in that it only resizes an image if the resolution is larger than 2048, and on that occasion, it's metadata is removed [4]. The research herein helps to expand on this previous research, and in addition, provides a variety of experimentations that yield results to inform best practices for forensic investigations on Google Plus and Flickr images.

CHAPTER II

TECHNICAL OVERVIEW

JPEG Compression

It is necessary to understand the process of JPEG compression and its Discrete Cosine Transform (DCT) as it relates to this research. Many people mistake JPEG as a file type, but more specifically it is a type of lossy compression method. Lossy means that information in an image is taken away in order to decrease data size and save space. Lossy is irreversible, once performed, information is lost forever. What we do in JPEG is we split each image into 8 x 8-pixel groups, and each of those pixel groups is separately encoded with its own DCT algorithm. DCT "expresses a finite sequence of data points in terms of a sum of cosine functions at different frequencies" [5]. Each of the 8 x 8-pixel groups within the DTC can be exactly replicated by 64 (8 x 8 cosine waves.) *Figure 1* shows the base cosine waves that produce any image one might like to perform in 8 x 8 pixels.
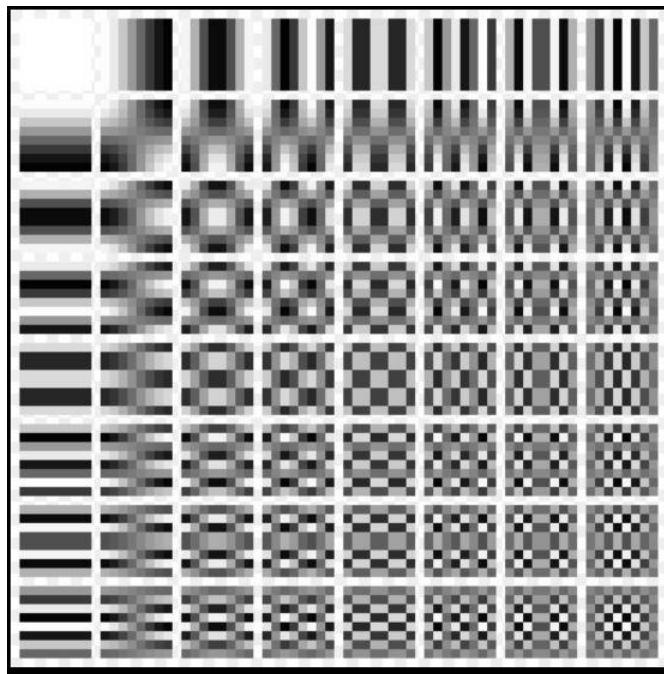

Figure 1. DCT base cosine waves in 8 x8 pixel groups (x64)

High and low frequencies is about the spatial distribution of intensity. High frequencies offer more detail to an image, while low frequencies offer less. Putting this into perspective, a beach of sand would be the low frequency, and each individual piece of sand that makes up the beach, are high frequencies. If the beach was missing, it would be very obvious to the human eye, while a dot of sand is much more difficult to spot. Consequently, the human eye is less sensitive to high frequencies in the pixel group, while low frequencies are more noticeable [17].

The numbers represented in *Figure 2* [6], are the standard luminance quantization table for a JPEG. Basically, each 8 x 8 block of DCT encoded data is quantized based on the values in this table. When an image is compressed, higher frequencies are forfeited so the change goes unnoticed all the while, saving space. This is the "outer layer" description of how lossy JPEG compression works.

$Q =$

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Figure 2. Standard Qauntization Table for JPEG Image

Information Behind Digital Images; Terms, Concepts and Definitions

digital media authentication. The term digital authentication as it pertains to this research, is about putting measures into place to determine if digital images are what they purport to be and whether or not they have been altered. Forensic image analysis helps assist in the determination of authenticity and includes the examination, evaluation and comparison of

image data. As defined in C. Grigoras et al. (2013) digital image enhancement and authentication research:

> In forensic imagery, the primary image consists of the data first recorded onto digital media from which the digital signal or file can be transferred in the native format or exported to another one. The digitally recorded information is stored as a finite set of binary values and exact duplicates or clones can be further made. (p. 305)

Color Filter Array (CFA) and Compression Level Analysis (CLA) are digital image authentication techniques to show demonstrative comparisons between what may or may not be authentic images. Color Filter Array is "a mosaic of tiny color filters (e.g., red, green, blue) placed over the camera sensor to filter and capture color information" [26]. CFA introduces specific correlations and algorithms between subsets of pixels and is considered a digital signature of a camera model [26]. When an image is copied, resaved or downloaded, the original CFA is altered. CFA analysis is an effective and precise way to trace compression and modifications to images.

hash values. In this research the term hash will be used and refers to the use of hash values. Hash values are displayed in a set of letters and numbers to verify that an image is identical to the source media. It is equivalent to a digital fingerprint.

> A hash algorithm is a function that converts a data string into a numeric string output of fixed length. The output string is generally much smaller than the original data. Hash algorithms are designed to be collision-resistant, meaning that there is a very low probability that the same string would be created for different

data. Two of the most common hash algorithms are the <u>MD5</u> (Message-Digest algorithm 5) and the SHA-1 [14].

JPEGmini. The term JPEGmini is referred to during discussions about Flickr. JPEGmini are optimized JPEG files that work to reduce the size of an image: "JPEGmini is a photo optimization technology that reduces the file size of JPEG photos by up to 5X, while preserving the resolution and quality of the original photos" [15].

WebP. WebP is tested during the evaluation of Google Plus. It is an image format output by this social platform and is a format exclusively developed by Google. Specifically, WebP is "a modern image format that provides superior lossless and lossy compression for images on the web. Using WebP, webmasters and web developers can create smaller, richer images that make the web faster" [14].

hexadecimal, binary file and hex editors. Computers work with long strings of binary data represented by a series of zeros and ones, a system that is too strenuous for humans to assimilate. Less complicated, and more compact, the Hexadecimal system was designed to assist in the understanding and the representation of value in a more human friendly way. Hexadecimal is a numeral system that uses 16 symbols, 0 – 9 and A – K [18]. Hex editors are tools used to view file data as a hexadecimal, and are useful for inspecting a binary file for raw content. It can also be used to spot suspicious edits to files or to recover corrupted or deleted files during digital investigations [21]. The structure of a file normally consists of a filename, a file header and footer, and file content. A header is the first few bytes that mark the beginning of the file, while footers contain the ending point of a file. These locations contain signatures or information used to identify the content of the file [23]. This information is translated by means of ASCII

interpretation of hex values, which stands for American Standard Code for Information Interchange [22].

Flickr

Many readers are already aware of the assortment of social networks made available to users. Back in the day, Flickr used to be the go-to spot to share and organize photos online before other popular sites took over. Still, Flickr remains a viable source to turn to for passionate photo lovers and photographers, with an average of 6.8 million digital photographs uploaded a day [7]. Flickr's aesthetic blog speaks volumes about the community it attracts. Undeniably, the science and technique that goes into creating high-quality photographs is important to this robust crowd. Flickr recognizes this and has a "long-standing commitment to keeping uploaded images-byte-for-byte intact" [8], however, the vast number of large images uploaded to Flickr presents costly storage challenges. In order to achieve more space and save cost, Flickr rolled out what they call 'dynamic generation of thumbnail sizes' and 'perceptual compression' to images that were uploaded to the site after 2014 [8]. The repercussion of these tactics is shown effectively in a test conducted by PetaPixel, where you can see that thumbnail sizes were reduced up to 60% [9].
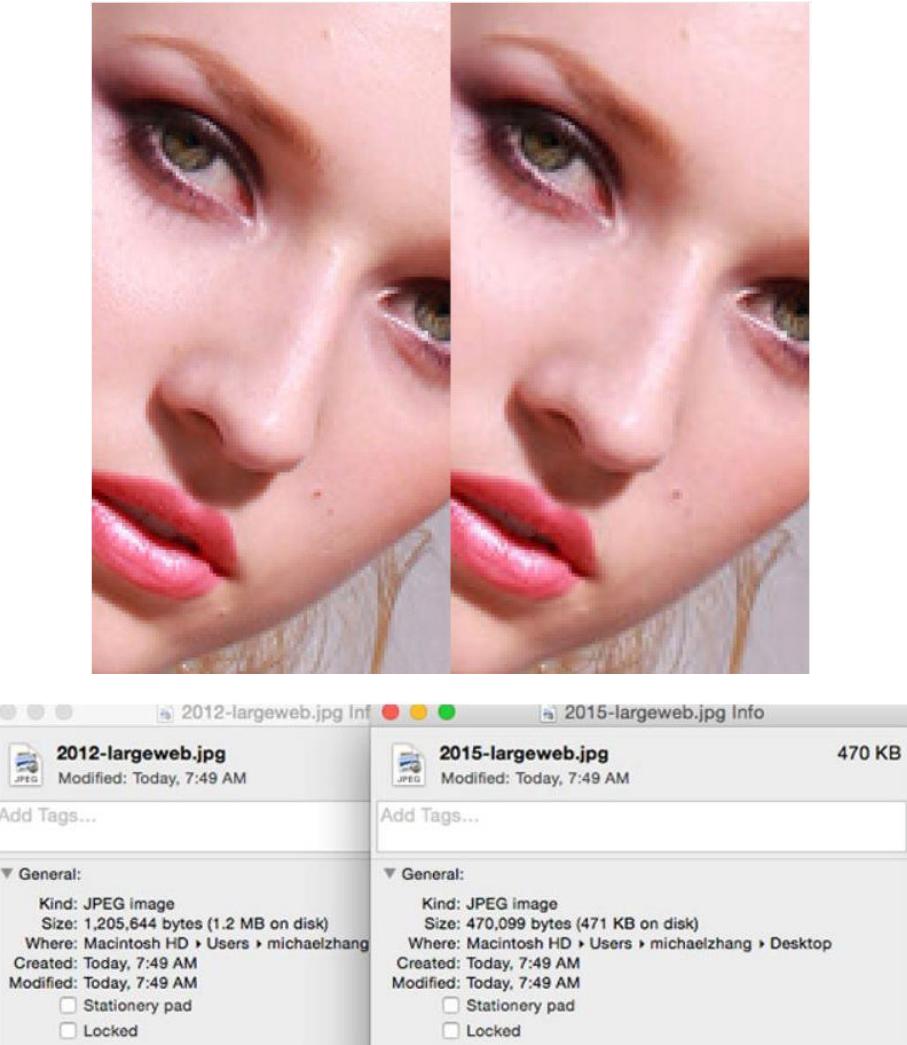
Figure 3. Flickr image comparison test conducted by PetaPixel: Left version (2012) has 60% better resolution than the image on the right (2015) [9].

Flickr offers a variety of size option links from which to download images, including the option to download the actual 'original', which is the catalyst to storage shortage. Here, Flickr uses Lepton, a streaming image compression format that reduces JPEG images by 22% [10]. Suitably, it also provides lossless bit-exact storage. Lossless compression is "a class of data compression algorithms that allows the original data to be perfectly reconstructed from the compressed data" [11]. In that way, the process works much like zipping and extracting a large file.

While Flickr has continued to stay true to their vision under the ownership of Yahoo, it will be interesting to see what happens after the recent acquisition by Verizon in June 2017. Under Verizon's directive, new innovations and features could be on the way.

Google Plus

Google Plus is a social network owned by Google that was introduced in 2011. It initially adopted features from other popular social networks, mixing in its own unique functionality like 'Circles' and 'Hangouts' and acts as a social layer across many of Google's properties. Their concern tends to focus in on the optimization of user accounts, and not so much on the social interaction between others. So, while Google Plus is not the most prevalent of social networks out there, building a presence within America's most popular search engine may be important to some. Much of how the users engage on Google Plus is the same as Twitter and Facebook, however Google Plus is far more customizable and allows a greater level of focused privacy. Not much was found in regards to what types of users engage most on this network, but just perusing through, it seems to be another popular spot for photographers to congregate and share their images.

CHAPTER III

PREPARATION

Flickr and Google Plus accounts were both created to examine changes of images, if any, after being uploaded and downloaded to and from each platform. A Yahoo account was created for Flickr and Gmail for Google Plus. A total of five JPEG test images that were captured by a LG-D850 mobile phone, and then transferred and stored into a Samsung-SM-J320, were used as reference standards to pinpoint what characteristics of an image may have changed. The test evaluated JPEG quantization tables, the resizing of images, and metadata. Exiftool, HxD,, ExactFile and JPEG Snoop are freeware forensic tools that capture these characteristics of an image and were utilized in this study. Exiftool reads metadata in an image's data and indicates information such as image size, GPS and the name of a file [19]. HxD is a hex editor used to recover data and also provides an image's hex analysis as well as the ASCII interpretation of hex values. ExactFile is used to display the hash value of image files and was used to verify the integrity of images after they were uploaded and downloaded to and from the social sites. JPEG Snoop decodes an image to include the image size and some of the EXIF data [20]. Quantization Tables were acquired from JPEG Snoop to evaluate image compression for the purposes of this research.

Images were uploaded and downloaded to both social sites using a Windows 10 Laptop and the Samsung-SM-J320 mobile phone (where the LG D850 mobile phone images were stored.) Electronic devices and the impact they have on transferring images to and from the social platforms were compared. In addition to comparing those differences, varying internet browsers were also explored. The internet browsers tested were Internet Explorer, Chrome, and Mozilla Firefox.

Lastly, it was tested if multiple people extract the same results when downloading images from these social platforms. To test this, a copy of the links to both social site accounts were emailed to Dr. Grigoras and Professor Smith, who were then able to click on the link and visit the accounts directly to download images. The internet browsers they used were Mozilla Firefox with the 'DownloadThemAll' plug-in and Google Chrome Version 62.0.3202.94 (Official Build) (64-bit.) After completing their downloads, their version of the image was emailed back to me for evaluation.

CHAPTER IV

METHOD

In order to understand the cause and effect that Google Plus and Flickr have on images, it is important to look at the metadata of an original image against the same image after it has been uploaded and downloaded to these platforms. Therefore, the analysis started by collecting information and data of both original images and downloaded images. A comparison was then done by equating the metadata including observing distinct characters located in the hex, noting how platforms name images and resize them, and crosschecking hash values. Platform settings create limitations in the ability to see and download protected images from social sites, therefore, a trial and error of download methods were tested and measured and are discussed in more detail below. An overall evaluation is discussed during the conclusion of this thesis paper.

Uploading and downloading images

In the process of uploading and downloading images from Google Plus and/or Flickr from an electrical device, data may be altered. For the purpose of this research, images were transferred and evaluated four different ways: 1. Upload and download via mobile phone 2.

Upload and download via desktop 3. Upload from mobile phone and download to desktop 4. Upload from desktop and download to mobile phone.

Social Network Settings

The ease of customizing privacy settings and tiered controls varies by service. It can also increase the difficulty of downloading an image for the purpose of obtaining meaningful data. However as long as the image is viewable, there are still ways to download them, although a lot of data (if not all) will be stripped. On occasion, some details may remain that could be pertinent to a case. Actual examples of this occurred during the study, when the original size of the image was not changed and the IMEI number and brand of the mobile used to transfer images was exposed. This study experiments with the different ways to grab images from Flickr and Google Plus, evaluates the changes, and attempts downloading images using different internet browsers that may potentially be more successful at acquiring data than others.

Obtaining Protected Images

There are only two sure-fire ways to protect your images on Flickr and Google Plus, 1. Make them private 2. Do not upload them. The bottom line is: if a photo can be seen in a web browser, it can be downloaded, copied or saved.

To get around a restricted image on Flickr, Google Plus and any image on the web, one can simply right click (in some browsers) and select 'inspect image'. This opens a box, and in that box, is an icon on the top left that looks like a square with a pointer in it. When you click on it and hover the mouse over the image (the image will turn blue), the HTML code responsible for displaying that page appears as highlighted -delete that string of commands. That brings down the protection. Right click the image. The image is now able to be saved to computer as a JPEG.

This cannot be done on a mobile phone, and when performed on a desktop or laptop, strips most of an image's metadata.
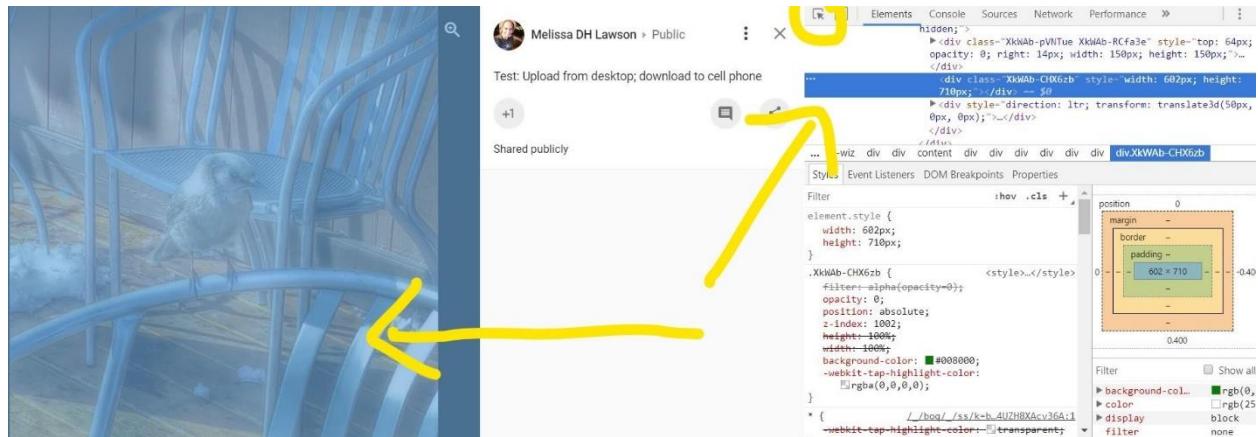

Figure 4. Screen capture: disarming image on Google Plus to save as JPEG

Flickr Settings

By default, all images on Flickr are public. When an image is public, anyone may view and download it. A user may choose to restrict access to their photos by making their accounts private, or alternatively, a user may apply certain restrictions that inhibit a person's ability to download. If this is the case, the option to download will say "owner has disabled downloading of their photos." Photos can also be downloaded in various sizes beginning at 75 x 75 all the way up to the original size.

In an attempt to achieve what technique is the best practice to download protected images on Flickr, yielding the best results, the following was tested:

1. Downloading an image while signed into the account.

2. Downloading an image while logged out of the account; public download permitted.

3. Downloading an image while logged out of the account; public download disabled.

4. Dr. Grigoras downloading images from Flickr's optional size links 500 x 292 or larger.

Because Flickr asserts that original images are kept intact bit-per-bit, it was predicted that images allowed to be downloaded from this platform would not be altered while on the contrary, performing special methods to download protected images, would. *Table 1* below compiles the tests conducted and provides a basic summary:

Table 1. Flickr: Proposed Method Summary

Method	Download Capability	Quantization	Exif Data	Summary Notes
U/D via mobile phone logged in	🟢	🟢	🟢	Results = original
U/D via mobile phone; logged out; public download enabled	🟢	🟢	🟢	Results = original
U/D via mobile phone; logged out; public download disabled	🔴			
U/D via desktop while logged in	🟢	🟢	🟢	Results = original
U/D via desktop; logged out; download enabled	🟢	🟢	🟢	Results = original
U/D via desktop; logged out; download disabled	🟡	🔴 Quantization tables altered	🔴 altered	Results = downloading possible via inspecting image
Upload from mobile phone; download to desktop; logged in	🟢	🟢	🟢	Results = original
Upload from mobile phone; download to desktop; logged out; public download enabled	🟢	🟢	🟢	Results = original
Upload from mobile phone; download to desktop; logged out; public download disabled	🟡	🔴 Quantization tables altered	🔴 altered	Results = downloading possible via inspecting image
Upload from desktop; download to mobile phone; logged in	🟢	🟢	🟢	Results = original
Upload from desktop; download to mobile phone; logged out; download enabled	🟢	🟢	🟢	Results = original
Upload from desktop; download to mobile phone; logged out; public download disabled	🔴			
Download by another user from their personal computer and browser; different sizes public download enabled	🟢	🟢	🟢	Results = original

Download images 500 x 292 or larger; downloaded by two different users	🟢	🔴 *Quantization tables altered*	🔴 *altered*	*Results = different than original, however both users rendered same results*
Download by another user from their personal computer and browser; original version public download enabled	🟢	🟢	🟢	*Results = original*

Google Plus Settings

Google Plus sharing features can also be controlled. Account holders can share their posts with everyone or with only those in designated circles. Circles are like categories for a person's connections. While some posts can be selected to share to the public, other posts may be left private for only a selected group to see. This sets limitations as to what is possible in the downloading of images. In order to conduct this research, settings had to be adjusted to allow for public viewing.

There are two ways to download images from Google Plus. One, is by right clicking the image and selecting 'save as'. This saves an image as a WebP format. The second way to download images on Google Plus works similarly to 'inspecting' the image, however you right click outside of the image and select 'view source page' instead. In the page that opens you can search for the keyword 'JPG'. Locate the URL that contains the image and copy / paste the URL into a browser URL bar. That will bring up the image. Right click on the image, and save as a JPG.

,[null,"","",1507774968738,"z12yjtaxlu3vyvglc22uwfmponmxwnf22#1507774968738165",null,"107200714377388860125","z12yjtaxlu3vyvglc22uwfmponmxwnf22",
[null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,false]
,null,null,null,null,null,null,null,false,null,["Tony Funderburk","107200714377388860125",false,false,"https://lh6.googleusercontent.com/-rqLXKI4a
sG6N9Wbk/photo.jpg","./+TonyFunderburk","male",null,null,true]
,["en",false]
,[[0,"And no, I didn't know you live in that area. How about that?! There's not much of that area that looks like it did when I was first there a
interesting things."]
,[1]
]
]
]
,null,null,null,1,null,null,[[null,null,
]
]
,"z12yjtaxlu3vyvglc22uwfmponmxwnf22",   :updates:esshare",null,null,"",null,null,"110489355980503749652",null,"https://lh5.googleusercontent.com
n8/photo.jpg",null,"Test: Upload from D   op; Download to Desktop -by public
user","+MelissaDHLawson/posts/7kuUayuvo7   0.0,"./+MelissaDHLawson",null,null,null,"",false,0,false,true,null,false,true,null,false,150759939895
ull,false,false,null,null,false,null,   ,null,null,false,null,null,null,null,null,null,null,null,null,
[null,null,null,null,null,null,null,n   null,null,null,null,null,null,null,null,false]
,null,null,true,null,true,null,0,null,   ,"social.google.com",false,false,null,null,true,false,null,"",7,false,null,0,[[344,339,437,338,336
,"https://plus.google.com/photos/11048935598   9652/albums/6475090111837474865/6475090111877912434?
authkey\u003dCNOs67i4xIHDMw",null,null,null,   null,"https://plus.google.com/photos/110489355980503749652/albums/6475090111837474865/64750901118
["https://plus.google.com/photos/110489355980503749652/albums/6475090111837474865/6475090111877912434","https://lh3.googleusercontent.com/-
F4sqmlRGKk8/WdwkJncgl3I/AAAAAAAAZPk/ygD0ap05ViEyVkzRDib6w_YRVw_yISIQgCJoC/w2340-h2340/20160315_105057_HDR%257E2.jpg","20160315_105057_HDR~2.jpg",".
user",null,[["//lh3.googleusercontent.com/-F4sqmlRGKk8/WdwkJncgl3I/AAAAAAAAZPk/ygD0ap05ViEyVkzRDib6w_YRVw_yISIQgCJoC/w506-h750/20160315_105057_HDR%
[1,"https://lh3.googleusercontent.com/-F4sqmlRGKk8/WdwkJncgl3I/AAAAAAAAZPk/ygD0ap05ViEyVkzRDib6w_YRVw_yISIQgCJoC/w800-h800/20160315_105057_HDR%257
,null,null,null,null,null,"plus.google.com",null,null,null,null,null,null,null,2340,2340,null,null,null,"110489355980503749652",null,nul
65","6475090111877912434",null,1,null,null,null,null,"photos/110489355980503749652/albums/6475090111837474865/6475090111877912434",null,null,null,
ull,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,n
ll,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,null,nul
[[339,338,336,335,0]
,"https://lh3.googleusercontent.com/-F4sqmlRGKk8/WdwkJncgl3I/AAAAAAAAZPk/ygD0ap05ViEyVkzRDib6w_YRVw_yISIQgCJoC/w2340-h2340/20160315_105057_HDR%257
F4sqmlRGKk8/WdwkJncgl3I/AAAAAAAAZPk/ygD0ap05ViEyVkzRDib6w_YRVw_yISIQgCJoC/w2340-h2340/20160315_105057_HDR%257E2.jpg","https://lh3.googleuserconte
F4sqmlRGKk8/WdwkJncgl3I/AAAAAAAAZPk/ygD0ap05ViEyVkzRDib6w_YRVw_yISIQgCJoC/w2340-h2340/20160315_105057_HDR%257E2.jpg"]
}]
,null,null,null,null,null,null,null,null,"0.6942094","2340","2340"]

Figure 5. Screen capture of view source page

The type of internet browser used during the seizing of images on Google Plus also became of interest in the observation that methods of downloading images using various internet browsers were producing diverse results in the EXIF data. To explore this area further, and to explore the effects that electronic devices may also have, the following was tested on images:

1. Upload and download via desktop; Internet Explorer logged in.

2. Upload and download via desktop; Microsoft Edge; logged out.

3. Upload and download via desktop; Microsoft Edge; logged in.

4. Upload from desktop; download to desktop; Mozilla FireFox; logged out.

5. Upload from desktop; download to desktop; Moxilla FireFox with DownloadThemAll plugin.

6. Upload and download via mobile phone.

7. Upload from desktop and download to mobile phone.

8. Upload from mobile phone; download to desktop; Microsoft Edge; logged out.

9. Upload from mobile phone; download to desktop; Internet Explorer; logged out.

10. Upload from mobile phone; download to desktop; Microsoft Edge; logged in.

11. Upload from mobile phone; download to desktop; Internet Explorer; logged in.

12. Upload from desktop; download to desktop – Dr. Grigoras using Mozilla Firefox.

13. Upload from desktop; download to desktop; Chrome / WebP to JPG.

The table below is a compilation of the techniques applied to download images and a brief summary of those results:

Table 2. Google Plus Method Summary

Method	Download Capability	Quantization	Exif Data	Summary Notes
Upload and Download via desktop; Internet Explorer logged in	🟢	🔴	🔴	*Results = quantization tables and metadata altered*
Upload and download via desktop; Microsoft Edge logged out	🟢	🔴	🔴	*Results = quantization tables and metadata altered*
Upload and download via desktop; Microsoft Edge logged in	🟢	🔴	🔴	*Results = quantization tables and metadata altered*
Upload from desktop; Download to desktop; Mozilla FireFox; logged out	🟢	🔴	🔴	*Results = quantization tables and metadata altered*
Upload from desktop; Download to desktop; Moxilla FireFox with DownloadThemAll plugin	🟢	🟢	🟡	*Results = metadata altered, however image quality and size remained the same and the brand of the device that uploaded image was revealed*
Upload and Download via mobile phone	🟢	🔴	🔴	*Results = quantization tables and metadata altered*

Upload from desktop and download to mobile phone	🟢	🔴	🔴	*Results = quantization tables and metadata altered*
Upload from mobile phone; download to desktop; Microsoft Edge; logged out	🟢	🔴	🔴	*Results = quantization tables and metadata altered*
Upload from mobile phone; download to desktop; Internet Explorer; logged out	🟢	🔴	🔴	*Results = quantization tables and metadata altered*
Upload from mobile phone; download to desktop; Microsoft Edge; logged in	🟢	🔴	🔴	*Results = quantization tables and metadata altered*
Upload from mobile phone; download to desktop; Internet Explorer logged in	🟢	🔴	🔴	*Results = quantization tables and metadata altered*
Upload from Desktop; Download to Desktop -by PROFESSOR using Mozilla Firefox	🟢	🟢	🟡	*Results = metadata altered, however image quality and size remained the same and the brand of the device that uploaded image was revealed*
Upload from desktop; Download to Desktop; Chrome / WebP to JPG	🟢	🟢	🟡	*Results = metadata altered, however image quality and size remained the same and the brand of the device that uploaded image was revealed*

CHAPTER V

FLICKR RESULTS

As predicted, Flickr preserved images as long as two things happened: 1. The account holder allowed downloads of their original images and/or 2. You are the account holder and logged in. This was the case for both accomplishing the task on a mobile phone or desktop by anyone. *Table 3* demonstrates (in partial) an example of preserved metadata in the EXIFtool results after downloading an image with permissions. More importantly, hash values are useful in verifying the integrity of a digital file, and *Figure 6* shows how even hash values match after being downloaded from this platform under these settings. The only notable change in the metadata is renaming of the images. It appears that an image downloaded to a mobile phone is renamed to the date it is being saved, followed by another segment of numbers + .jpg. Images downloaded to desktop are saved as a string of numbers, followed by _o.jpg. While modifying any content *within* an image will change the hash value, *renaming* a file does not.

Table 3. Partial EXIF Data for Two Different Images; One Downloaded to Mobile Phone, One Downloaded to Desktop. Names Were Changed.

UPLD / DNLD Mobile Phone	UPLD / DNLD Desktop
ExifToolVersion = 9.90	ExifToolVersion = 9.90
FileName = 36070463433_2075a560c7_o.jpg	FileName = 36880336625_cea97a3838_o.jpg
Directory = .	Directory = .
FileSize = 1671607	FileSize = 1238212
FileModifyDate = 1503958279.45154	FileModifyDate = 1503964097.71911
FileAccessDate = 1503958283.05747	FileAccessDate = 1503964101.56196
FileCreateDate = 1503958283.05747	FileCreateDate = 1503964101.56196
FilePermissions = 33206	FilePermissions = 33206
FileType = JPEG	FileType = JPEG
MIMEType = image/jpeg	MIMEType = image/jpeg
JPEG APP1 (18133 bytes):	JPEG APP1 (1330 bytes):
ExifByteOrder = MM	ExifByteOrder = MM
+ [IFD0 directory with 9 entries]	+ [IFD0 directory with 10 entries]
| 0) GPSInfo (SubDirectory) -->	| 0) ModifyDate = 2015:05:07 10:30:54
| + [GPS directory with 8 entries]	

| | 0) GPSLatitudeRef = N
| | 1) GPSLatitude = 39 44 8.2269 (39/1 44/1 82269/10000)
| | 2) GPSLongitudeRef = W
| | 3) GPSLongitude = 104 59 20.0646 (104/1 59/1 200646/10000)
| | 4) GPSAltitudeRef = 0
| | 5) GPSAltitude = 1623 (1623000/1000)
| | 6) GPSTimeStamp = 17 56 13 (17/1 56/1 13/1)

| 1) GPSInfo (SubDirectory) -->
| + [GPS directory with 8 entries]
| | 0) GPSDateStamp = 2015:05:07
| | Warning = Tag ID 0x0005 out of sequence in GPS
| | 1) GPSAltitudeRef = 0
| | Warning = Tag ID 0x0003 out of sequence in GPS
| | 2) GPSLongitudeRef = W


```
ExactFile Report        Image Download from Flickr
Copy    Copy All

Checksums generated by ExactFile 1.0.0.15
http://www.exactfile.com
10/24/2017 12:02:19 AM

C:\Users\Melissa\Desktop\Thesis Paper\ORIGINAL PHOTO ANALYSIS\20150116_182
4537855 bytes

MD5: 8bba53e2a5a59ee0c112b66146faef9f
SHA256: 34ecb4999d410249eb027e5d5bdb05207b4aec8b5299126e6d56f4e0ad8e7da9
```

```
ExactFile Report                Original Image
Copy    Copy All

Checksums generated by ExactFile 1.0.0.15
http://www.exactfile.com
10/23/2017 11:47:23 PM

C:\Users\Melissa\Desktop\Thesis Paper\Flickr\Flickr Photo Analysis\Upload
4537855 bytes

MD5: 8bba53e2a5a59ee0c112b66146faef9f
SHA256: 34ecb4999d410249eb027e5d5bdb05207b4aec8b5299126e6d56f4e0ad8e7da9
```

Figure 6. Flickr version (top) and original image (bottom) hash values match

When optimal scenarios for preserving images is not the case, and hash values did not agree, further analysis was necessary. Downloading protected images was not possible, and the

following circumstances and platform settings created dissimilarities:
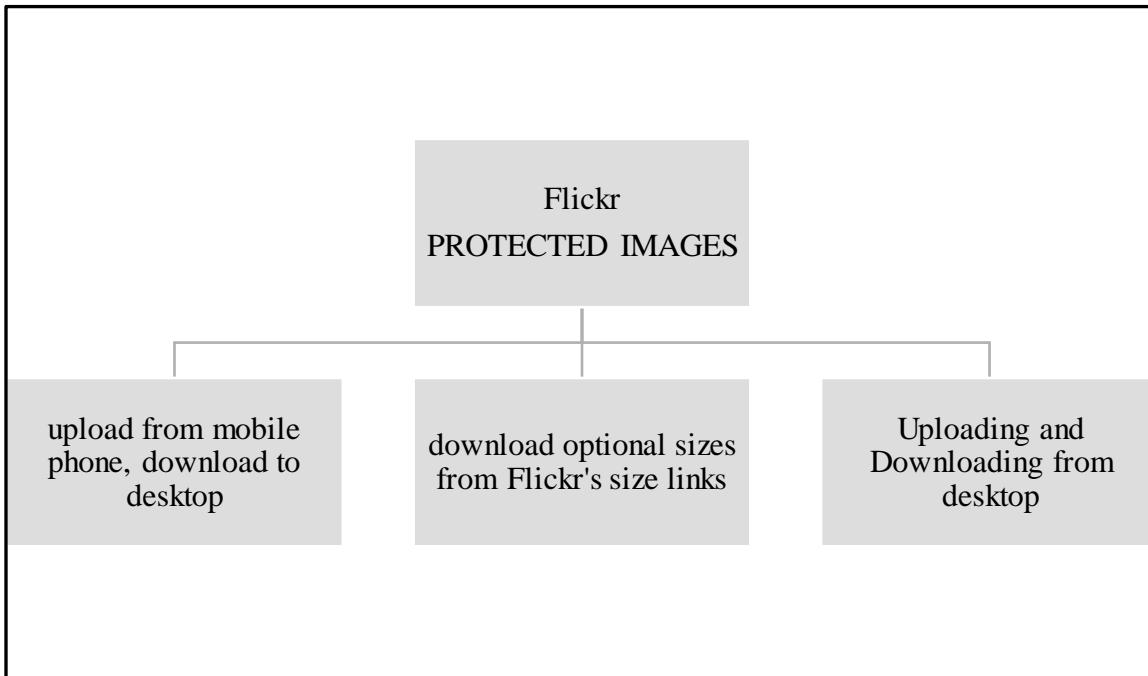
Figure 7. Using These Methods to Download Images from Flickr Strips Data

In proposed methods it is explained that to save restricted images to a desktop, certain steps are necessary to get past the protection. After these steps, the image is able to be saved as a jpeg. However, the process stripped the metadata from the original and the quality of the image was decreased. Tables A-1 through A-4 in Appendix A show the results. Tables A-1 and A-2 demonstrate the difference in metadata and quality factor of an image before (left column) and after (right column) it has been uploaded and downloaded to and from Flickr. Tables A-3 and A-4 (name and file size changes are highlighted in the tables) illustrate that downloading an image from Flickr's optional size links vs obtaining protected images via the view source page (and deleting the HTML) is indistinguishable, implying neither method is better than the other and infers the attempt of preserving any sort of authentic metadata from protected images on Flickr, is not possible.

So what observations should one consider when analyzing images from Flickr? Underneath is a simplified summary of the results and what observations stood out -some of which relate and are similar to the discoveries exposed in Pippin, M. (2016) research as it pertains to image resizing, image renaming and desktop vs mobile phone downloads (p. 28-31.) Following the summary, a diagram (*Figure 8*) has been provided that further summarizes Flickr image storage and processing in more detail.

Summary of Images Permitting Downloads

1. Are renamed to a string of numbers followed by _o.jpg when downloaded to desktop

2. Images downloaded to mobile phone are renamed to the date in the which the image was saved, followed by more segmented groups of numbers + .jpg

3. All original metadata is intact whether downloaded to mobile phone or desktop

Summary of Images Protected From Public Downloads

4. It is not possible to download images to mobile phone if they are protected.

5. When an image is uploaded from mobile phone, the image results in 1024 pixels in width and retains the original image aspect ratio.

6. When an image is uploaded from desktop, the image results in 2048 pixels in width and retains the original image aspect ratio.

7. If an image was 1024-pixel width, EXIF data revealed: optimized by JPEGmini 3.13.3.15 0xad6b4f35

8. If an image was 2048-pixel width, EXIF data revealed: Optimized by JPEGmini 3.13.3.15 0x3cf38bdf

9. The size of the image also appears to influence the name change: images in 2048-pixel width were renamed as a string of numbers ending in _k.jpg, 1024-pixel width were renamed as a string of numbers ending in _b.jpg.

10. The renaming of images (string of numbers and all) match whether or not the image was downloaded via Flickr's 'size option' link OR if the image was saved via 'inspecting' the original image to get past the protection. For example, image *20140930_115613_Burst04.jpg* was renamed to 36070463433_f58a95a391_k.jpg when saving an image via Flickr's '1152 x 2048' size link or by way of inspecting the image (which by default saved the image to size 1152 x 2048.)

SUMMARY	Images with download permissions OR account holder logged in; both mobile phone and desktop
	RESULTS
DESKTOP	• *Name Change** 36070463433_2075a560c7_o.jpg
MOBILE PHONE	• *Name Change** 2017-08-28_05-19-45.jpg
DESKTOP AND MOBILE PHONE	• No alterations in Metadata, Quantization Tables (hash value = match)

SUMMARY	Upload / download via desktop, download not permitted

RESULTS

- Cannot save unless protection (HTML) is deleted
- Download default size: 1152 x 2048
- Optimized by JPEGmini 3.13.3.15 0x3cf38bdf
- Name Change* 36070463433_f58a95a391_k.jpg
- Altered* Metadata, Quantization Tables (hash values = no match)
- Original 4160 x 2340 / Download 1152 x 2048 = 1.77 Aspect Ratio

SUMMARY	Upload from mobile phone, download to desktop, download not permitted

RESULTS

- Cannot save unless protection (HTML) is deleted
- Download default size: 775 x 1024
- Optimized by JPEGmini 3.13.3.15 0xad6b4f35
- Name Change* 36880336625_11ff6133fe_b.jpg
- Altered* Metadata, Quantization Tables (hash values = no match)
- Original 1719 x 2270 / Download 775 x 1024 = 1.32 Aspect Ratio

SUMMARY	Download *20140930_115613_Burst04.jpg* via Flickr size link 1152 x 2048 **VS.** delete HTML to save protected image onto desktop
	RESULTS
DOWNLOAD IMAGE VIA FLICKR DOWNLOAD 'SIZE' LINK **VS** **SAVE PROTECTED IMAGE TO DESKTOP VIA INSPECT IMAGE**	• Optimized by JPEGmini 3.13.3.15 0x3cf38bdf • Name Change* 36070463433_f58a95a391_k.jpg **VS** • Saved to default size: 1152 x 2048 • Optimized by JPEGmini 3.13.3.15 0x3cf38bdf • Name Change* 36070463433_f58a95a391_k.jpg (name change corresponds with image downloaded from Flickr 'size' link)

SUMMARY	Download *20150507_093353_HDR~2~2.jpg* via Flickr size link 775 x 1024 **VS.** delete HTML to save protected image onto
	RESULTS
DOWNLOAD IMAGE VIA FLICKR DOWNLOAD 'SIZE' LINK **VS** **SAVE PROTECTED IMAGE TO DESKTOP VIA INSPECT IMAGE**	• Optimized by JPEGmini 3.13.3.15 0xad6b4f35 • Name Change* 36880336625_11ff6133fe_b.jpg **VS** • Saved to default size 775 x 1024 • Optimized by JPEGmini 3.13.3.15 0xad6b4f35 • Name Change* 36880336625_11ff6133fe_b.jpg (name change corresponds with image downloaded from Flickr 'size' link)

Figure 8. Summary of Flickr Analysis Results

CHAPTER VI

GOOGLE PLUS RESULTS

Unlike Flickr, the option to download original images and/or images in optional sizes is not available on Google Plus. As it is, how a user obtains images from Google Plus widely depends on what internet browser is being used, which incidentally yields varying results in the quality and metadata of images. Like Flickr, some approaches for downloading images off Google Plus strip metadata and reduce the pixel size and therefore the quality and authenticity of the image is lost. These approaches all seemed to have a commonality of obtaining an image by right clicking it and selecting 'save as' on Internet Explorer, Microsoft Edge and Chrome (see Appendix B, Table B-1 and Table B-2 for their EXIF data results.) It is important to note that saving an image from Google Plus on Chrome in this way, will save an image as a WebP format. This only occurs on Chrome as the format has yet to be supported by a solid majority of programs and browsers [25]. WebP comes in lossy and lossless formats and offers a dramatic drop in file size with very little reduction in image quality [16]. Perhaps the best file type for procuring images with the highest quality, is WebP. The file format can be viewed in both Exiftool (*Table 4*) and Hex reader tools (*Figure 9*). It looks like this:

Table 4. Example EXIF Data for WebP File Format of Test Image

```
ExifToolVersion  = 9.90
FileName  = 20160315_105057_HDR-2.webp
Directory  = .
FileSize  = 78946
FileModifyDate  = 1509245302.85195
FileAccessDate  = 1509245227.10217
FileCreateDate  = 1509245301.97813
FilePermissions  = 33206
FileType  = WEBP
MIMEType = image/webp
RIFF 'VP8X' chunk (10 bytes of data):
```

```
VP8X (SubDirectory) -->
+ [BinaryData directory
| ImageWidth  = 2516583062
| ImageHeight = 169472
RIFF 'VP8 ' chunk (78866 bytes of data):
VP8Bitstream (SubDirectory) -->
+ [BinaryData directory
| VP8Version  = 0
| ImageWidth  = 663
| HorizontalScale  = 0
| ImageHeight = 663
| VerticalScale  = 0
RIFF 'EXIF' chunk (34 bytes of data):
EXIF (SubDirectory) -->
+ [TIFF directory]
| ExifByteOrder  = II
| + [IFD0 directory with 1 entries]
| | 0)  Software = Google
```
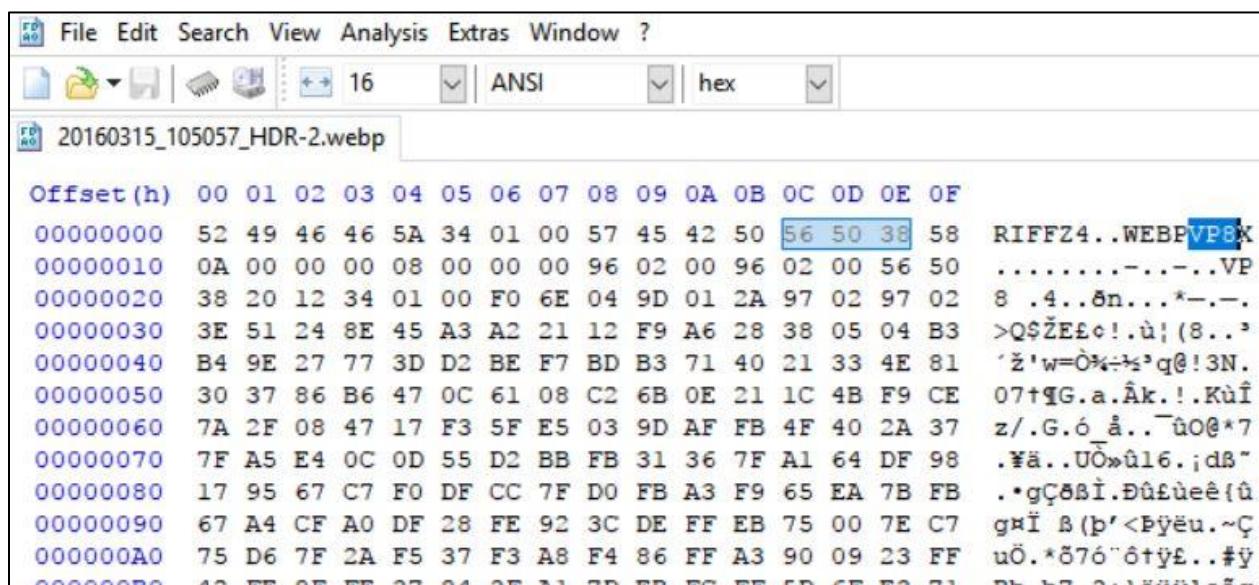

Figure 9. Example of Hex Reader WebP File Format of test Image

The WebP file formats do not contain any of the embedded data in an original image, and in the figure above highlighted in blue, VP8 corresponds to a lossy version of a WebP file format. If the version was lossless, it would read VP8L. Retaining a lossless image from Google Plus

was not possible, although visually the downloaded outcome was not noticeable compared to the original.

While the above methods to download images from Google Plus strip metadata and reduce the quality of images, the option to save images through the 'view page source' on Internet Explorer and Chrome preserves exact pixel dimensions of images. This was also the case using Mozilla Firefox with the 'DownloadThemAll' plug-in. While lossy compression alters some original information embedded in an image, the ability to download images without compressing and reducing their size makes this method of downloading images from Google Plus a best practice in the seizure of evidence (Appendix B, Table B-3.) In addition to being able to obtain actual pixel size through this method, further exploration gave rise to another very important finding. Unique information, "Samsung Trailer / Backup Restore_Data" with associated numbers, appeared in the extension of one of the test image's EXIF data (see *Table 5*).

Table 5. Unique Data located in EXIF Data for image 20160315_105057_HDR~2.jpg

```
ExifToolVersion  = 9.90
FileName = 20160315_105057_HDR-2.jpg
Directory = .
FileSize = 1244487
FileModifyDate  = 1509252181.91304
FileAccessDate  = 1509252176.48793
FileCreateDate  = 1509252181.39927
FilePermissions  = 33206
FileType = JPEG
MIMEType = image/jpeg
JPEG APP0 (14 bytes):
 + [BinaryData directory
 | JFIFVersion  = 1 1
 | ResolutionUnit  = 0
 | XResolution  = 1
 | YResolution  = 1
JPEG APP1 (106 bytes):
 ExifByteOrder  = II
 + [IFD0 directory with 3 entries]
```

```
| 0)  Orientation  = 1
| 1)  Software = Google
| 2)  ExifOffset  (SubDirectory)  -->
| + [ExifIFD  directory  with 3 entries]
|| 0)  ExifVersion  = 0220
|| 1)  ExifImageWidth  = 2340
|| 2)  ExifImageHeight  = 2340
JPEG DQT (65 bytes):
JPEG DQT (65 bytes):
JPEG SOF0 (15 bytes):
 ImageWidth  = 2340
 ImageHeight  = 2340
 EncodingProcess  = 0
 BitsPerSample  = 8
 ColorComponents  = 3
JPEG DHT (29 bytes):
JPEG DHT (179 bytes):
JPEG DHT (29 bytes):
JPEG DHT (179 bytes):
JPEG SOS
Samsung trailer (77 bytes at offset 0x12fcfa):
 SamsungTrailer_0x0a41Name  = BackupRestore_Data
 SamsungTrailer_0x0a41  = 355022071107283_698
```

Because this signature was not showing up in any other test image, it was apparent that something about this particular digital image was unique and therefore was investigated more in the HxD hex editor tool. In the footer of the hex data, "BackupRestore_Data" was also present as well as the same associated numbers that proved to be the IMEI number of the Samsung phone used to conduct the transfer of images in this study (*Figure 10* and *Figure 11.*)

Figure 10. Unique Content Found in Footer of Hex Data 'BackupRestore_Data'
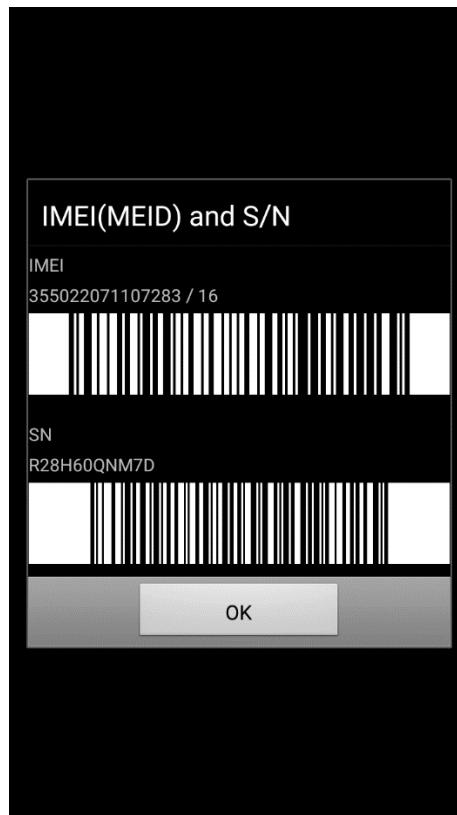
Figure 11. Screen Shot of IMEI number from Samsung Phone Used in the Research

Not a lot could be found about the signature "BackupRestore_Data" and why it is rooted

in only certain images over others, however there are online discussions about it. In a dialog on

the XDA Developers website, it is indicated that a possible reason for this signature to occur is

due to the Samsung Gallery Application, that when turned on, groups photos automatically and

creates event folders for them. When turned on, the signature appears in the hex. When turned off, it goes away [27].

The fact that Google Plus was retaining data embedded in the footer of this file (unique evidence that identifies a specific mobile phone), was an important fiding and could be extremely useful for investigations involving images on Google Plus. To see if the platform saves any and all data embedded in any digital image, a test image that did not contain unique content in its original footer, was opened in the HxD hex editor tool and edited by entering '..M.e.l.i.s.s.a.H.e.l.l.o.0' at the end of the file structure. The edited version was then saved as a new file. Next, the edited image was uploaded to and from Google Plus and the downloaded version was observed in Exiftool and HxD. Indeed, Google Plus still preserved the original size and altered data implanted in the footer. The only difference was that the edit is only visible in hex, not in the EXIF data (see *Figure 12*, *13* and *14*.) This was the same circumstance for the image containing unique data in its original footer after editing it for testing. Last but not least, these tests were also explored on Flickr. Flickr did not preserve any of this information and all metadata was stripped.

Figure 12. Original Hex Data of Image: 20150305_145611_HDR~2~3~2.jpg Does not Contain Unique Data in the footer

```
00046990  E0 9E E3 D6 B6 22 73 B6 76 3F 31 14 51 5F AF 70  àžãÖ¶"s¶v?1.Q ¯p
000469A0  DA 4B 0C AC 7E 69 9E 36 EA EA 46 97 5E 48 CA 20  ÚK.¬~iž6êêF—^HÊ
000469B0  07 CD C5 49 75 FE 91 75 0C 2F F7 37 C7 45 15 F5  .ÍÅIuþ'u./÷7ÇE.õ
000469C0  C7 C9 75 3C F3 E2 74 BF 6D D7 52 07 1F BA 47 DA  ÇÉu<óât¿m×R..ºGÚ
000469D0  AA 3D 01 C0 15 99 E0 FD 2E 09 3C 49 A8 CC C3 32  ª=.À.™àý..<I¨ÌÃ2
000469E0  C7 1A A4 6D DD 41 07 38 A2 8A F8 FC F9 B5 86 76  Ç.¤mÝA.8¢Šøüùµtv
000469F0  3E DB 27 4B 42 E2 21 46 70 CC 5D 81 3F 39 EB DE  >Û'KBâ!FpÌ].?9ëÞ
00046A00  AD DB CE F0 5F 5A 00 41 59 48 8D 81 1D B3 45 15  .ÛÎð_Z.AYH...'E.
00046A10  F9 E6 03 5A C8 FA 8C 47 F0 66 75 B7 F3 32 5C CB  ùæ.ZÈúŒGðfu·ó2\Ë
00046A20  1A F0 A8 8A 07 E1 8A A6 D2 36 48 CF 71 45 15 FB  .ð¨Š.áŠ¦Ò6HÏqE.û
00046A30  86 1B F8 48 FC 97 13 FC 49 8D F3 7D A8 A2 8A A3  †.øHü—.üI.ó}¨¢Š£
00046A40  23 FF D9 2E 2E 4D 2E 65 2E 6C 2E 69 2E 73 2E 73  #ÿÙ..M.e.l.i.s.s
00046A50  2E 61 2E 48 2E 65 2E 6C 2E 6C 2E                 .a.H.e.l.l.o.0
```

Figure 13. Edited Hex is Preserved in the Footer of the Edited Version After Being Downloaded from Google Plus

Figure 14. EXIF Data Results for Edited Version: 20150305_145611_HDR~2~3~2.jpg

In addition to the observations made in the application of different download methods on various internet browsers, here are some other Google Plus behaviors to be aware of:

1. Images that retained their original pixel dimensions after downloading them from Google Plus, are indeed an exact pixel replica of the original as shown in their quantization tables (*Figure 15*), CFA Analysis (*Figure 16*), and Compression Level Analysis (*Figure 17*).

Original Image Quantization Table

3	2	2	3	5	8	10	12
2	2	3	4	5	12	12	11
3	3	3	5	8	11	14	11
3	3	4	6	10	17	16	12
4	4	7	11	14	22	21	15
5	7	11	13	16	21	23	18
10	13	16	17	21	24	24	20
14	18	19	20	22	20	21	20
3	4	5	9	20	20	20	20
4	4	5	13	20	20	20	20
5	5	11	20	20	20	20	20
9	13	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20

Downloaded Image Quantization Table

3	2	2	3	5	8	10	12
2	2	3	4	5	12	12	11
3	3	3	5	8	11	14	11
3	3	4	6	10	17	16	12
4	4	7	11	14	22	21	15
5	7	11	13	16	21	23	18
10	13	16	17	21	24	24	20
14	18	19	20	22	20	21	20
3	4	5	9	20	20	20	20
4	4	5	13	20	20	20	20
5	5	11	20	20	20	20	20
9	13	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20

Figure 15. Quantization Table Comparison of Image Downloaded from Google Plus Using Best Practices

Original Image CFA Analysis
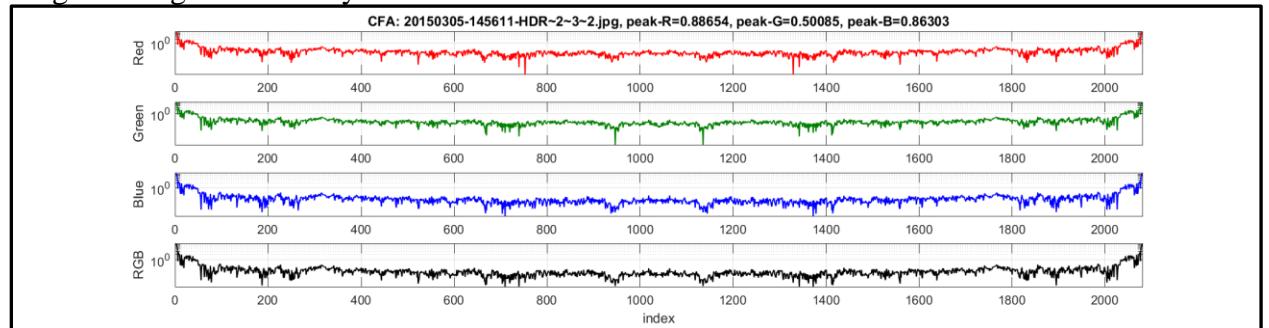
Downloaded Image CFA Analysis
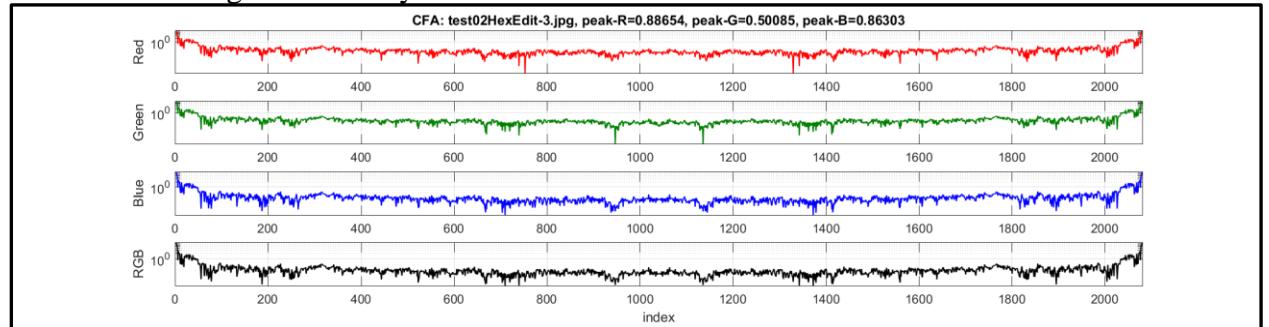
Figure 16. CFA Comparison Analysis Match Exactly Using Best Practices

Original Image CLA Analysis
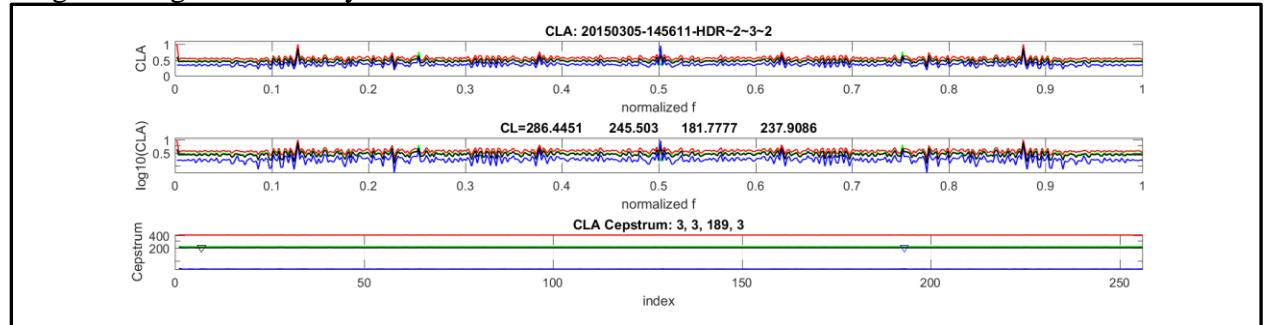
Downloaded Image CLA Analysis
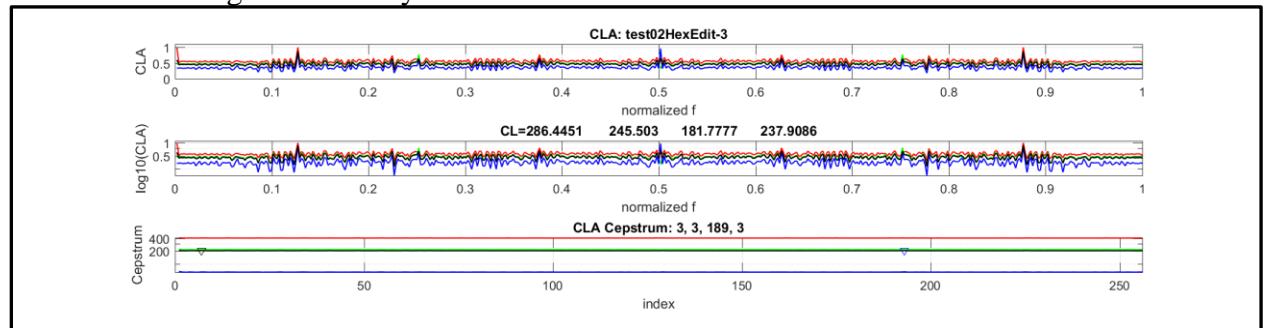
Figure 17. CLA Comparison Shows No Sign of Compression Using Best Practices

2. Google Plus saved images as either Photo.jpg and/or Photo-Google+.jpg as a public user that does not own the account. The diagrams provided in this research paper list when images were saved as their original names, however this will never be the case during real investigations. The computer conducting the test was naming the images by their original name due to information being stored to its browser and not being cached.

3. All downloaded images had the same aspect ratio as the originals.

4. Images that were 1060 pixel-width, correlate with images downloaded to mobile phone. Images that were 663 pixel-width, correlate with images downloaded to desktop.

5. Images downloaded from Google Plus contain: *Software = Google* in the EXIF and hex data (Diagram of results in *Figure 18*.)

SUMMARY	Upload from desktop; Download by desktop; Moxilla FireFox with DownloadThemAll plugin
	RESULTS • Retained same Quantization Tables as original image • 2340 x 2340 = same as the original image • Name Change* saves as Photo-Google+.jpg • Software: Identifier 'Google' in EXIF Data

SUMMARY	Upload from desktop; Download to Desktop; Using chrome and saving to JPG
	RESULTS • Retained same Quantization Tables as original image • 2340 x 2340 = same as the original image • Saves as original name (20160315_105057_HDR-2.jpg) • Software: Identifier 'Google' in EXIF Data

SUMMARY	Upload and Download from desktop Internet Explorer
	RESULTS • Original 4160 x 2340 / Download 663 x 1178 = 1.77 Aspect Ratio • Saved as original name (20140930_115613_Burst04.jpg) • Altered* Quantization Tables • Software: Identifier 'Google' in EXIF Data
SUMMARY	Upload and Download to cell Phone
	RESULTS • Original 4160 x 2340 / Download 1060 x 596 = 1.77 Aspect Ratio • Saved as original name (20150116_182100_HDR.jpg) • Altered* Quantization Tables • Software: Identifier 'Google' in EXIF Data
SUMMARY	Upload from cell phone; download to desktop
	RESULTS • Original 1719 x 2270 / Download 663 x 875 = 1.32 Aspect Ratio • Saved as original name (20150507_093353_HDR~2~2.jpg) • Altered* Quantization Tables • Software: Identifier 'Google' in EXIF Data
SUMMARY	Upload from Desktop and download to Cell Phone
	RESULTS • Original 865 x 1221 / Download 1060 x 1496 = 1.4 Aspect Ratio • Saved as original name (20150305_145611_HDR-2-3-2.jpg) • Altered* Quantization Tables • Software: Identifier 'Google' in EXIF Data

Figure 18. Summary of Google Plus Analysis Results

CHAPTER VI

CONCLUSION AND FUTURE RESEARCH

To recap, digital image evidence from social media networks is useful in many types of

criminal investigations and court proceedings. Investigative leads, for instance, could be solving

timelines of an event or learning details about the account holder. This work aimed to perform an experimental study on Flickr and Google Plus to understand the base behavior of these social media platforms and inform best practices for forensic investigations. It illustrated the potential ways for acquiring information about digital images and interpreting the metadata and, in the process, concludes two core findings: first being that Flickr preserves images 100% as long as the account user allows public downloads of their original images. If the account user does not permit this, it is not possible to obtain authentic information and data from images downloaded from Flickr other than the minor observations listed in this research.

The second vital finding is that Google Plus preserves exact pixel dimensions of original images *and* any unique data that may be embedded in the extension of image files *providing* best practices are followed:

1. Use Internet Explorer or Chrome to download images from Google Plus.

2. Right click alongside an open image in Google Plus, select 'view page source' and find the URL containing the JPG image, copy and paste URL in browser, right click the image that opens and select 'save as'. At this point, the image will be saved as a JPG and in its exact size as the original. Additionally, should an image contain unique data in the footer of its file structure, the information will appear in the hex data as well as in the EXIF data.

3. Saving an image using Mozilla Firefox with the 'DownloadThemAll' plug-in will easily do all of this in less steps and acquire the same optimal results more quickly.

   Being aware of these core findings and following these best practices will help spot evidence more efficiently and assure that investigators and forensic analysts are obtaining images from these platforms in those most optimal way.

The process of downloading an image will usually result in stripping meaningful data. Should an investigation demand further interpretation of behaviors particular to Flickr and Google Plus in order to build a case, how images are resized and renamed are other key factors to consider. This study confirms that Flickr works similar to Facebook according to related research conducted by M. Pippins (2016.) Denoting that, the size of the image influences how an image is named: images in 2048-pixel width were renamed as a string of numbers ending in _k.jpg and 1024-pixel width were renamed as a string of numbers ending in _b.jpg. Finally, image settings that allowed public downloads were saved as a string of numbers followed by _o.jpg. To determine whether or not this extension is related to a certain size, future research using a bigger pool of test images would need to be analyzed. Google Plus, renames images either as 'Photo.jpg' or the obvious giveaway, 'Photo-Google+.html' and/or '.webp'.

One last suggestion is to look for the tagline 'Software – Google" in the image EXIF data. This is an obvious indicator that the images are associated with the Google Plus platform.

Future Research

WebP is an emerging image format that merits future exploration especially as new browsers begin to adopt the format. Because it is potentially the best format to obtain high image quality, it may assist investigations involving online images. Specifically, is it possible to obtain lossless versions of WebP formatted images from online platforms and if so how?

Flickr, previously owned by Yahoo, was acquired by Versizon in June 2017. While Flickr has had a long history of preserving the quality of images, Versizon may make changes that would impact findings in future research. Future testing on Flickr may be necessary.

REFERENCES

1. Josh Gilliland. (2013). The admissibility of social media evidence. Litigation, 39(1), 20-21. Retrieved from http://www.jstor.org/stable/24396778.

2. En.wikipedia.org. (2017). Metadata. [online] Available at: https://en.wikipedia.org/wiki/Metadata [Accessed 21 Oct. 2017].

3. Pippin, M. (2016). Digital image recompression analysis : Facebook University of Colorado Denver.

4. Castiglione, A., Cattaneo, G., & De Santis, A. (2011). A forensic analysis of images on online social networks. Paper presented at the 679-684. doi:10.1109/INCoS.2011.17.

5. En.wikipedia.org. (2017). *Discrete cosine transform*. [online] Available at: https://en.wikipedia.org/wiki/Discrete_cosine_transform [Accessed 21 Oct. 2017].

6. En.wikipedia.org. (2017). *JPEG*. [online] Available at: https://en.wikipedia.org/wiki/JPEG [Accessed 21 Oct. 2017].

7. Flickr. (n.d.). *How many public photos are uploaded to Flickr every day, month, year?* [online] Available at: https://www.flickr.com/photos/franckmichel/6855169886 [Accessed 21 Oct. 2017].

8. Mellina, C., Russell, A., Patel, M. and Travlos, X. (2017). *code.flickr.com*. [online] Code.flickr.net. Available at: https://code.flickr.net/ [Accessed 21 Oct. 2017].

9. PetaPixel. (n.d.). *Flickr Decreased Quality and Increased Compression, and Users Aren't Happy*. [online] Available at: https://petapixel.com/2015/06/08/flickr-decreased-quality-and-increased-compression-and-users-arent-happy/ [Accessed 21 Oct. 2017].

11. Dropbox Tech Blog. (2017). *Lepton image compression: saving 22% losslessly from images at 15MB/s*. [online] Available at: https://blogs.dropbox.com/tech/2016/07/lepton-image-compression-saving-22-losslessly-from-images-at-15mbs/ [Accessed 21 Oct. 2017].

12. En.wikipedia.org. (2017). *Lossless compression*. [online] Available at: https://en.wikipedia.org/wiki/Lossless_compression [Accessed 21 Oct. 2017].

13. En.wikipedia.org. (2017). *Exif*. [online] Available at https://en.wikipedia.org/wiki/Exif [Accessed 29 Oct. 2017].

14. Google Developers. (n.d.). *A new image format for the Web | WebP | Google Developers*. [online] Available at: https://developers.google.com/speed/webp/ [Accessed 29 Oct, 2017].

15. Digitalizationguidelines.gov. (n.d.). *Hash algorithm – Glossary – Federal Agencies Digitization Guidelines Initiative*. [online] Available at: http://www.digitizationguidelines.gov/term.php?term=hashalgorithm [Accessed 29 Oct, 2017].

16. Says, J.B., Says, D., Says, R.B., Says, M., & Says, T. (2017, January 25). *Is there any benefit to using WebP images on my website?* [online] Available at: https://havecamerawilltravel.com/photographer/webp-website/ [Accessed November 11, 2017].

17. Shi, Y. Q., & Sun, H. (2008). *Image and video compression for multimedia engineering: Fundamentals, algorithms, and standards* (2nd ed.) Boca Raton: CRC Press.

18. En.wikipedia.org. (2017). *Hexadecimal*. [online] Available at: https://en.wikipedia.org/wiki/Hexadecimal [Accessed 19 Nov. 2017].

19. En.wikipedia.org. (2017). *Exiftool*. [online] Available at: https://en.wikipedia.org/wiki/ExifTool [Accessed 19 Nov. 2017].

20. JPEGsnoop (moved to GitHub). (n.d.). Available at: https://sourceforge.net/p/jpegsnoop/wiki/Home/ [Accessed 19. Nov. 2017].

21. Warlock. (2013). *File carving*. [online] Available at: http://resources.infosecinstitute.com/file-carving/ [Accessed 20. Nov. 2017].

22. Vangie Beal. (n.d.). *ASCII*. [online] Available at: https://www.webopedia.com/TERM/A/ASCII.html [Accessed 20. Nov. 2017].

23. J. Holt, Thomas & Bossler, Adam & Seigfried-Spellar, Kathryn. (2015). Cybercrime and Digital Forensics: An Introduction.

24. CBSNews.com. (n.d.). *Social Media Related Crimes*. [online] Available at: https://www.cbsnews.com/pictures/social-media-related-crimes/ [Accessed 20. Nov. 2017].

25. Optimus (June 15, 2017). *WebP support – it's more than you think.* [online] Available at: https://optimus.keycdn.com/support/webp-support/ [Accessed 27. Nov. 2017].

26. Koenig, B. E., Lacey, D. S., Grigoras, C., Price, S. G., & Smith, J. M. (2013). Evaluation of the average DC offset values for nine small digital audio recorders. *Journal of the Audio Engineering Society*, *61*(6), 439-448.

27. XDA Developers (May 11, 2017). Photos modified days after taken (exif metadata added). [online] Available at: https://forum.xda-developers.com/galaxy-s7/help/photos-modified-days-t3451600. [Accessed 28. Nov. 2017].

"Appendix A"

Flickr: EXIF Data Analysis

Table A-1.

EXIF Data and Quality Factor: Original Image vs Image Uploaded and Downloaded Via Desktop, Download Not Permitted

Original Image	Image After Download
ExifToolVersion = 9.90	ExifToolVersion = 9.90
FileName = 20140930_115613_Burst04.jpg	FileName = 36070463433_f58a95a391_k.jpg
Directory = .	Directory = .
FileSize = 1671607	FileSize = 354532
FileModifyDate = 1503888971.87222	FileModifyDate = 1505105943.89992
FileAccessDate = 1503299726	FileAccessDate = 1505105938.72692
FileCreateDate = 1503299726	FileCreateDate = 1505105943.5694
FilePermissions = 33206	FilePermissions = 33206
FileType = JPEG	FileType = JPEG
MIMEType = image/jpeg	MIMEType = image/jpeg
JPEG APP1 (18133 bytes):	JPEG APP0 (14 bytes):
ExifByteOrder = MM	+ [BinaryData directory
+ [IFD0 directory with 9 entries]	| JFIFVersion = 1 1
| 0) GPSInfo (SubDirectory) -->	| ResolutionUnit = 1
| + [GPS directory with 8 entries]	| XResolution = 72
|| 0) GPSLatitudeRef = N	| YResolution = 72
|| 1) GPSLatitude = 39 44 8.2269 (39/1 44/1 82269/10000)	JPEG COM (43 bytes):
|| 2) GPSLongitudeRef = W	Comment = Optimized by JPEGmini 3.13.3.15 0x3cf38bdf
|| 3) GPSLongitude = 104 59 20.0646 (104/1 59/1 200646/10000)	JPEG DQT (65 bytes):
|| 4) GPSAltitudeRef = 0	JPEG DQT (65 bytes):
|| 5) GPSAltitude = 1623 (1623000/1000)	JPEG SOF0 (15 bytes):
|| 6) GPSTimeStamp = 17 56 13 (17/1 56/1 13/1)	ImageWidth = 1152
|| 7) GPSDateStamp = 2014:09:30	ImageHeight = 2048
| Warning = Tag ID 0x0112 out of sequence in IFD0	EncodingProcess = 0
| 1) Orientation = 6	BitsPerSample = 8
| Warning = Tag ID 0x0110 out of sequence in IFD0	ColorComponents = 3
| 2) Model = LG-D850	JPEG DHT (27 bytes):
| 3) ExifOffset (SubDirectory) -->	JPEG DHT (90 bytes):
| + [ExifIFD directory with 19 entries]	JPEG DHT (25 bytes):
|| 0) InteropOffset (SubDirectory) -->	JPEG DHT (45 bytes):
|| + [InteropIFD directory with 2 entries]	JPEG DRI (2 bytes):
||| 0) InteropIndex = R98	JPEG SOS
	(END)

```
  ||| 1) InteropVersion = 0100
  || Warning = Tag ID 0x9004 out of sequence in
ExifIFD
  || 1) CreateDate = 2014:09:30 11:56:13
  || 2) ColorSpace = 1
  || Warning = Tag ID 0x9003 out of sequence in
ExifIFD
  || 3) DateTimeOriginal = 2014:09:30 11:56:13
  || 4) FlashpixVersion = 0100
  || Warning = Tag ID 0x9207 out of sequence in
ExifIFD
  || 5) MeteringMode = 2
  || Warning = Tag ID 0x9204 out of sequence in
ExifIFD
  || 6) ExposureCompensation = 0 (0/1)
  || 7) ExifImageHeight = 2340
  || Warning = Tag ID 0x9000 out of sequence in
ExifIFD
  || 8) ExifVersion = 0220
  || 9) ExifImageWidth = 4160
  || Warning = Tag ID 0x920a out of sequence in
ExifIFD
  || 10) FocalLength = 3.97 (3970/1000)
  || 11) DigitalZoomRatio = 1 (100/100)
  || Warning = Tag ID 0x9209 out of sequence in
ExifIFD
  || 12) Flash = 0
  || 13) WhiteBalance = 0
  || Warning = Tag ID 0x829a out of sequence in
ExifIFD
  || 14) ExposureTime = 0.001131221719 (1/884)
  || 15) ISO = 50
  || 16) ComponentsConfiguration = 1 2 3 0
  || Warning = Tag ID 0x829d out of sequence in
ExifIFD
  || 17) FNumber = 2.4 (240/100)
  || 18) MakerNoteUnknown (SubDirectory) -->
  ||| Warning = [minor] Unrecognized MakerNotes
 | Warning = Tag ID 0x0213 out of sequence in
IFD0
 | 4) YCbCrPositioning = 1
 | Warning = Tag ID 0x011b out of sequence in
IFD0
 | 5) YResolution = 72 (72/1)
 | 6) ResolutionUnit = 2
 | Warning = Tag ID 0x011a out of sequence in
IFD0
```

```
| 7)  XResolution  = 72 (72/1)
| Warning = Tag ID 0x010f out of sequence in IFD0
| 8)  Make = LG Electronics
+ [IFD1 directory with 7 entries]
| 0)  YResolution  = 72 (72/1)
| 1)  ResolutionUnit  = 2
| Warning = Tag ID 0x011a out of sequence in
IFD1
| 2)  XResolution  = 72 (72/1)
| Warning = Tag ID 0x0112 out of sequence in
IFD1
| 3)  Orientation  = 6
| Warning = Tag ID 0x0103 out of sequence in
IFD1
| 4)  Compression  = 6
| 5)  ThumbnailOffset  = 1316
| 6)  ThumbnailLength  = 16811
JPEG DQT (130 bytes):
JPEG SOF0 (15 bytes):
  ImageWidth  = 4160
  ImageHeight  = 2340
  EncodingProcess  = 0
  BitsPerSample  = 8
  ColorComponents  = 3
JPEG DHT (416 bytes):
JPEG SOS
```

Quality Factors of Original JPEG Vs. Quality Factors of JPEG Compression After Flickr

Original Luminance	Post Download Luminance
2 2 1 2 3 6 7 9	4 3 1 4 6 9 12 15
2 2 2 3 4 8 8 8	3 3 3 5 6 14 14 13
2 2 2 3 6 8 10 8	3 3 4 6 9 14 17 13
2 2 3 4 7 12 11 9	3 4 5 7 12 21 19 15
3 3 5 8 10 15 14 11	4 5 9 13 16 26 24 18
3 5 8 9 11 15 16 13	6 8 13 15 19 24 27 22
7 9 11 12 14 17 17 14	12 15 18 21 24 29 29 24
10 13 13 14 16 14 14 14	17 22 23 23 27 24 24 24
Original Chrominance	Post Download Chrominance
2 3 3 7 14 14 14 14	4 4 6 11 24 24 24 24
3 3 4 9 14 14 14 14	4 5 6 15 24 24 24 24
3 4 8 14 14 14 14 14	6 6 13 24 24 24 24 24
7 9 14 14 14 14 14 14	11 15 24 24 24 24 24 24
14 14 14 14 14 14 14 14	24 24 24 24 24 24 24 24
14 14 14 14 14 14 14 14	24 24 24 24 24 24 24 24
14 14 14 14 14 14 14 14	24 24 24 24 24 24 24 24
14 14 14 14 14 14 14 14	24 24 24 24 24 24 24 24

Table A-2.

EXIF Data and Quality Factor: Original Image vs. Image Uploaded from Mobile Phone,
Downloaded to Desktop, Public Download Not Permitted

Original Image	Image After Download
ExifToolVersion = 9.90	ExifToolVersion = 9.90
FileName = 20150507_093353_HDR~2~2.jpg	FileName = 36880336625_11ff6133fe_b.jpg
Directory = .	Directory = .
FileSize = 1238212	FileSize = 390291
FileModifyDate = 1503888970.69028	FileModifyDate = 1507741038.286
FileAccessDate = 1503299680	FileAccessDate = 1507741033.3443
FileCreateDate = 1503299680	FileCreateDate = 1507741038.00875
FilePermissions = 33206	FilePermissions = 33206
FileType = JPEG	FileType = JPEG
MIMEType = image/jpeg	MIMEType = image/jpeg
JPEG APP1 (1330 bytes):	JPEG APP0 (14 bytes):
ExifByteOrder = MM	+ [BinaryData directory
+ [IFD0 directory with 10 entries]	| JFIFVersion = 1 1
| 0) ModifyDate = 2015:05:07 10:30:54	| ResolutionUnit = 1
| 1) GPSInfo (SubDirectory) -->	| XResolution = 72
| + [GPS directory with 8 entries]	| YResolution = 72
|| 0) GPSDateStamp = 2015:05:07	JPEG COM (43 bytes):
|| Warning = Tag ID 0x0005 out of sequence in GPS	Comment = Optimized by JPEGmini 3.13.3.15 0xad6b4f35
|| 1) GPSAltitudeRef = 0	JPEG DQT (65 bytes):
|| Warning = Tag ID 0x0003 out of sequence in GPS	JPEG DQT (65 bytes):
|| 2) GPSLongitudeRef = W	JPEG SOF0 (15 bytes):
|| 3) GPSLongitude = 104 53 45.7068 (104/1 53/1 457068/10000)	ImageWidth = 775
	ImageHeight = 1024
|| Warning = Tag ID 0x0001 out of sequence in GPS	EncodingProcess = 0
|| 4) GPSLatitudeRef = N	BitsPerSample = 8
|| 5) GPSTimeStamp = 15 33 51 (15/1 33/1 51/1)	ColorComponents = 3
|| Warning = Tag ID 0x0006 out of sequence in GPS	JPEG DHT (28 bytes):
|| 6) GPSAltitude = 1691 (1691000/1000)	JPEG DHT (92 bytes):
|| Warning = Tag ID 0x0002 out of sequence in GPS	JPEG DHT (26 bytes):
|| 7) GPSLatitude = 39 39 18.2983 (39/1 39/1 182983/10000)	JPEG DHT (60 bytes):
| Warning = Tag ID 0x0110 out of sequence in IFD0	JPEG SOS
| 2) Model = LG-D850	**(END)**
| 3) YCbCrPositioning = 1	
| Warning = Tag ID 0x0128 out of sequence in	

```
IFD0
 |4) ResolutionUnit = 2
 | Warning = Tag ID 0x011b out of sequence in
IFD0
 |5) YResolution = 72 (72/1)
 | Warning = Tag ID 0x0112 out of sequence in
IFD0
 |6) Orientation = 1
 |7) ExifOffset (SubDirectory) -->
 |+ [ExifIFD directory with 19 entries]
 ||0) ColorSpace = 1
 || Warning = Tag ID 0x9004 out of sequence in
ExifIFD
 ||1) CreateDate = 2015:05:07 09:33:53
 || Warning = Tag ID 0x829d out of sequence in
ExifIFD
 ||2) FNumber = 2.4 (240/100)
 ||3) FocalLength = 3.97 (3970/1000)
 ||4) WhiteBalance = 0
 || Warning = Tag ID 0xa002 out of sequence in
ExifIFD
 ||5) ExifImageWidth = 4160
 || Warning = Tag ID 0x9207 out of sequence in
ExifIFD
 ||6) MeteringMode = 2
 || Warning = Tag ID 0x9003 out of sequence in
ExifIFD
 ||7) DateTimeOriginal = 2015:05:07 09:33:53
 ||8) ComponentsConfiguration = 1 2 3 0
 ||9) ExifImageHeight = 2340
 || Warning = Tag ID 0x9209 out of sequence in
ExifIFD
 ||10) Flash = 0
 || Warning = Tag ID 0x9000 out of sequence in
ExifIFD
 ||11) ExifVersion = 0220
 ||12) InteropOffset (SubDirectory) -->
 ||+ [InteropIFD directory with 2 entries]
 |||0) InteropIndex = R98
 |||1) InteropVersion = 0100
 || Warning = Tag ID 0x9204 out of sequence in
ExifIFD
 ||13) ExposureCompensation = 0 (0/1)
 || Warning = Tag ID 0x8827 out of sequence in
ExifIFD
 ||14) ISO = 50
 ||15) FlashpixVersion = 0100
```

|| Warning = Tag ID 0x927c out of sequence in
ExifIFD
|| 16) MakerNoteUnknown (SubDirectory) -->
||| Warning = [minor] Unrecognized MakerNotes
|| 17) DigitalZoomRatio = 1 (100/100)
|| Warning = Tag ID 0x829a out of sequence in
ExifIFD
|| 18) ExposureTime = 0.01666666667 (1/60)
| Warning = Tag ID 0x011a out of sequence in
IFD0
| 8) XResolution = 72 (72/1)
| Warning = Tag ID 0x010f out of sequence in
IFD0
| 9) Make = LG Electronics
+ [IFD1 directory with 5 entries]
| 0) YResolution = 72 (72/1)
| Warning = Tag ID 0x0112 out of sequence in
IFD1
| 1) Orientation = 6
| Warning = Tag ID 0x0103 out of sequence in
IFD1
| 2) Compression = 6
| 3) ResolutionUnit = 2
| Warning = Tag ID 0x011a out of sequence in
IFD1
| 4) XResolution = 72 (72/1)
JPEG APP0 (14 bytes):
+ [BinaryData directory
| JFIFVersion = 1 1
| ResolutionUnit = 0
| XResolution = 1
| YResolution = 1
JPEG DQT (65 bytes):
JPEG DQT (65 bytes):
JPEG SOF0 (15 bytes):
  ImageWidth = 1719
  ImageHeight = 2270
  EncodingProcess = 0
  BitsPerSample = 8
  ColorComponents = 3
JPEG DHT (29 bytes):
JPEG DHT (179 bytes):
JPEG DHT (29 bytes):
JPEG DHT (179 bytes):
JPEG SOS

Quality Factors of Original JPEG Vs. Quality Factors of JPEG Compression After Flickr

Original Luminance	Post Download Luminance
3 2 2 3 5 8 10 12	3 1 1 3 5 8 11 13
2 2 3 4 5 12 12 11	3 3 3 2 5 13 13 11
3 3 3 5 8 11 14 11	3 3 3 5 8 12 15 12
3 3 4 6 10 17 16 12	3 3 5 6 11 19 17 13
4 4 7 11 14 22 21 15	4 5 8 12 14 23 22 16
5 7 11 13 16 21 23 18	5 8 11 14 17 22 24 19
10 13 16 17 21 24 24 20	10 14 16 19 22 26 26 21
14 18 19 20 22 20 21 20	15 19 20 21 24 21 22 21
Original Chrominance	**Post Download Chrominance**
3 4 5 9 20 20 20 20	3 4 5 10 21 21 21 21
4 4 5 13 20 20 20 20	4 5 5 14 21 21 21 21
5 5 11 20 20 20 20 20	5 5 12 21 21 21 21 21
9 13 20 20 20 20 20 20	10 14 21 21 21 21 21 21
20 20 20 20 20 20 20 20	21 21 21 21 21 21 21 21
20 20 20 20 20 20 20 20	21 21 21 21 21 21 21 21
20 20 20 20 20 20 20 20	21 21 21 21 21 21 21 21
20 20 20 20 20 20 20 20	21 21 21 21 21 21 21 21

Table A-3

EXIF Data for Image 20140930_115613_Burst04.jpg: Saving Image via Flickr's Size Option Link 1152 x 2048 vs. Deleting HTML to Save Image to Desktop

Download from Flickr	Download via deleting HTML
ExifToolVersion = 9.90	ExifToolVersion = 9.90
FileName = 36070463433_f58a95a391_k.jpg	FileName = 36070463433_f58a95a391_k.jpg
Directory = .	Directory = .
FileSize = 354532	FileSize = 354532
FileModifyDate = 1506045673.60251	FileModifyDate = 1505105943.89992
FileAccessDate = 1506137043.53627	FileAccessDate = 1505105938.72692
FileCreateDate = 1506137043.53627	FileCreateDate = 1505105943.5694
FilePermissions = 33206	FilePermissions = 33206
FileType = JPEG	FileType = JPEG
MIMEType = image/jpeg	MIMEType = image/jpeg
JPEG APP0 (14 bytes):	JPEG APP0 (14 bytes):
+ [BinaryData directory	+ [BinaryData directory
| JFIFVersion = 1 1	| JFIFVersion = 1 1
| ResolutionUnit = 1	| ResolutionUnit = 1
| XResolution = 72	| XResolution = 72
| YResolution = 72	| YResolution = 72
JPEG COM (43 bytes):	JPEG COM (43 bytes):
Comment = Optimized by JPEGmini 3.13.3.15 0x3cf38bdf	

JPEG DQT (65 bytes): JPEG DQT (65 bytes): JPEG SOF0 (15 bytes): ImageWidth = 1152 ImageHeight = 2048 EncodingProcess = 0 BitsPerSample = 8 ColorComponents = 3 JPEG DHT (27 bytes): JPEG DHT (90 bytes): JPEG DHT (25 bytes): JPEG DHT (45 bytes): JPEG DRI (2 bytes): JPEG SOS	Comment = Optimized by JPEGmini 3.13.3.15 0x3cf38bdf JPEG DQT (65 bytes): JPEG DQT (65 bytes): JPEG SOF0 (15 bytes): ImageWidth = 1152 ImageHeight = 2048 EncodingProcess = 0 BitsPerSample = 8 ColorComponents = 3 JPEG DHT (27 bytes): JPEG DHT (90 bytes): JPEG DHT (25 bytes): JPEG DHT (45 bytes): JPEG DRI (2 bytes): JPEG SOS

Table A-4

EXIF Data: for Image 20150507_093353_HDR~2~2.jpg: Saving Image Via Flickr's Size Option Link 775 x 1024 vs. Deleting HTML to Save Image to Desktop

Download from Flickr	Download via Deleting HTML
ExifToolVersion = 9.90 FileName = 36880336625_11ff6133fe_b.jpg Directory = . FileSize = 390291 FileModifyDate = 1506045257.60762 FileAccessDate = 1506137043.72379 FileCreateDate = 1506137043.72379 FilePermissions = 33206 FileType = JPEG MIMEType = image/jpeg JPEG APP0 (14 bytes): + [BinaryData directory | JFIFVersion = 1 1 | ResolutionUnit = 1 | XResolution = 72 | YResolution = 72 JPEG COM (43 bytes): Comment = Optimized by JPEGmini 3.13.3.15 0xad6b4f35 JPEG DQT (65 bytes): JPEG DQT (65 bytes):	ExifToolVersion = 9.90 FileName = 36880336625_11ff6133fe_b.jpg Directory = . FileSize = 390291 FileModifyDate = 1507741038.286 FileAccessDate = 1507741033.3443 FileCreateDate = 1507741038.00875 FilePermissions = 33206 FileType = JPEG MIMEType = image/jpeg JPEG APP0 (14 bytes): + [BinaryData directory | JFIFVersion = 1 1 | ResolutionUnit = 1 | XResolution = 72 | YResolution = 72 JPEG COM (43 bytes): Comment = Optimized by JPEGmini 3.13.3.15 0xad6b4f35 JPEG DQT (65 bytes): JPEG DQT (65 bytes):

JPEG SOF0 (15 bytes):	JPEG SOF0 (15 bytes):
ImageWidth = 775	ImageWidth = 775
ImageHeight = 1024	ImageHeight = 1024
EncodingProcess = 0	EncodingProcess = 0
BitsPerSample = 8	BitsPerSample = 8
ColorComponents = 3	ColorComponents = 3
JPEG DHT (28 bytes):	JPEG DHT (28 bytes):
JPEG DHT (92 bytes):	JPEG DHT (92 bytes):
JPEG DHT (26 bytes):	JPEG DHT (26 bytes):
JPEG DHT (60 bytes):	JPEG DHT (60 bytes):
JPEG SOS	JPEG SOS

Appendix B

Google Plus: EXIF Data Analysis

Table B-1

EXIF Data Results: Method of Downloading - Right Click Image, Select 'Save As', Using
Internet Explorer. EXIF Data Results

```
 ExifToolVersion  = 9.90
  FileName  = 20160315_105057_HDR~2.jpg
  Directory  = .
  FileSize  = 122760
  FileModifyDate  = 1507606360.91573
  FileAccessDate  = 1509253906.66322
  FileCreateDate  = 1509253906.39457
  FilePermissions  = 33206
  FileType  = JPEG
  MIMEType = image/jpeg
JPEG APP0 (14 bytes):
 + [BinaryData directory
 | JFIFVersion  = 1 1
 | ResolutionUnit  = 0
 | XResolution  = 1
 | YResolution  = 1
JPEG APP1 (40 bytes):
  ExifByteOrder  = II
 + [IFD0 directory with 1 entries]
 | 0)  Software = Google
JPEG DQT (130 bytes):
JPEG SOF0 (15 bytes):
  ImageWidth  = 663
  ImageHeight  = 663
  EncodingProcess  = 0
  BitsPerSample  = 8
  ColorComponents  = 3
JPEG DHT (27 bytes):
JPEG DHT (89 bytes):
JPEG DHT (25 bytes):
JPEG DHT (56 bytes):
 JPEG SOS
```

Table B-2

EXIF Data Results: Method of Downloading – Right Click Image, Select 'Save As' Using Microsoft Edge.

```
   ExifToolVersion  = 9.90
   FileName  = 20160315_105057_HDR~2.jpg
   Directory  = .
   FileSize  = 122760
   FileModifyDate  = 1509254820.74757
   FileAccessDate  = 1509254820.73194
   FileCreateDate  = 1509254820.469
   FilePermissions  = 33206
   FileType  = JPEG
   MIMEType = image/jpeg
 JPEG APP0 (14 bytes):
  + [BinaryData directory
  | JFIFVersion  = 1 1
  | ResolutionUnit  = 0
  | XResolution  = 1
  | YResolution  = 1
 JPEG APP1 (40 bytes):
   ExifByteOrder  = II
  + [IFD0 directory with 1 entries]
  | 0)  Software = Google
 JPEG DQT (130 bytes):
 JPEG SOF0 (15 bytes):
   ImageWidth  = 663
   ImageHeight  = 663
   EncodingProcess  = 0
   BitsPerSample  = 8
   ColorComponents  = 3
 JPEG DHT (27 bytes):
 JPEG DHT (89 bytes):
 JPEG DHT (25 bytes):
 JPEG DHT (56 bytes):
   JPEG SOS
```

Table B-3

EXIF Data Results: Method of Downloading – "View Page Source" on Chrome and Internet Explorer OR Mozilla Firefox with Plug-in

```
        ExifToolVersion = 9.90
```

```
      FileName = 20150507_093353_HDR_2_2.jpg
      Directory = .
      FileSize = 1236988
      FileModifyDate = 1510466131.77645
      FileAccessDate = 1510466127.50872
      FileCreateDate = 1510466131.34912
      FilePermissions = 33206
      FileType = JPEG
      MIMEType = image/jpeg
JPEG APP0 (14 bytes):
 + [BinaryData directory
 | JFIFVersion = 1 1
 | ResolutionUnit = 0
 | XResolution = 1
 | YResolution = 1
JPEG APP1 (106 bytes):
 ExifByteOrder = II
 + [IFD0 directory with 3 entries]
 | 0) Orientation = 1
 | 1) Software = Google
 | 2) ExifOffset (SubDirectory) -->
 | + [ExifIFD directory with 3 entries]
 | | 0) ExifVersion = 0220
 | | 1) ExifImageWidth = 1719
 | | 2) ExifImageHeight = 2270
JPEG DQT (65 bytes):
JPEG DQT (65 bytes):
JPEG SOF0 (15 bytes):
 ImageWidth = 1719
 ImageHeight = 2270
 EncodingProcess = 0
 BitsPerSample = 8
 ColorComponents = 3
JPEG DHT (29 bytes):
JPEG DHT (179 bytes):
JPEG DHT (29 bytes):
JPEG DHT (179 bytes):
JPEG SOS
```