DIGITAL IMAGE ANALYSIS:

ANALYTICAL FRAMEWORK FOR

AUTHENTICATING DIGITAL IMAGES

by

Scott Dale Anderson

B.S., University of Colorado Denver, 2001

A thesis submitted to the

University of Colorado Denver

in partial fulfillment

of the requirements for the degree of

Master of Science

Media Forensics

2011

This thesis for the Master of Science

degree by

Scott Dale Anderson

has been approved

by

Catalin Grigoras

Jeff M. Smith

Gregory Walker

Richard W. Vorder Bruegge

Date

Anderson, Scott Dale (M.S., Media Forensics)

Digital Image Analysis: Analytical Framework for Authenticating Digital Images

Thesis directed by Associate Professor Catalin Grigoras

ABSTRACT

Due to the widespread availability of image processing software, it has become easier to produce visually convincing image forgeries. To overcome this issue, there has been considerable work in the digital image analysis field to determine forgeries when no visual indications exist. However, while certain manipulation techniques can elude one or more analyses, it may difficult to elude them all. This thesis proposes an analytical framework to help analysts determine the authenticity of digital images by considering the digital file and the image structure. Since statistical models are not possible for all digital image authentication cases, analytical evaluations are sometimes used to determine how observed features in the image structure compare to known characteristics of digital image creation. Chapter 1 explains digital image creation, tasks of an image analyst, and the principles that outline how forensic science is applied to court and the law. Chapter 2 reviews the current literature on digital image forgery detection and how they apply to an image file. Chapter 3 introduces an analytical framework by using case examples to illustrate how this approach can be used to strengthen conclusions when authenticating a digital image.

This abstract accurately represents the content of the candidate's thesis. I recommend its publication.

Signed _____

Catalin Grigoras

DEDICATION

First, I would like to dedicate this thesis to my father, Robert Dale Anderson, who passed away before he could see his son progress on to bigger and better things. He was always a pillar of support and strength, quick to put a smile on your face, and forever loving. Your absence is deeply felt.

I would also like to dedicate this thesis to my mother who has continued to love me for as long as I have known and gave me this wonderful gift of life.

Last, but certainly not least, I would like to dedicate this thesis to S. C., who has kept me grounded for the past eight years and showed me grace, compassion, love, and beauty, the likes of which I had never imagined possible. Your sacrifice will never go unremembered. I pray you find the peace you seek, and it is worth considerably more than the heavy price that was paid.

ACKNOWLEDGEMENT

First, I would like to thank GOD, through whom all things are possible.

I would like to express my sincerest gratitude to my thesis advisor, Dr. Catalin Grigoras, for his enthusiasm, patience, wisdom and guidance throughout my transition into the world of digital media forensics. He always made himself available to answer any question I had, even if it meant learning Romanian to ask it. The late night master's sessions were instrumental in a thorough understanding of the subject matter relevant to the field of digital media forensics, and the contents of this thesis.

I would also like to thank my thesis committee members for their critique and feedback on a very broad subject. In particular, I would like to thank Richard W. Vorder Bruegge for taking time out of his busy schedule to help ensure that my thesis was thorough, complete, and worthy of his signature.

I would also like to thank my mother and grandmother for supporting me during the final years of my education, so that I could concentrate solely on my education. I would like to express gratitude to Mema and my friends for providing support when I needed it most.

I could not have done this without each and every one of you.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. Introduction

This thesis concerns digital images and how they can be authenticated. While image authentication has been explored in previous research, published findings are often independent of one another, as each focuses on a specific aspect of a digital image. Furthermore, images used in these experiments are created in laboratory-controlled environments, with crudely constructed "manipulations." It is hard to determine how the results of these experiments could be affected if individuals with some level of skill and competency tried to cover their tracks. In instances such as these, an evaluation using multiple analyses and techniques is necessary for a proper assessment of image authentication [1].

For the most part, a digital image is comprised of a finite set of numbers, arranged by a series of mathematical algorithms to create a digital image file. Like all mathematical functions, these algorithms operate in a predefined, predictable way. If the output of one algorithm is altered, the alteration will most likely effect the output of other algorithms. While the effects of certain manipulation techniques can elude one or more analyses, it may be difficult or even impossible to elude them all.

Each image authentication case is unique, and must be approached individually. As a result, a well-defined approach for determining the authenticity of digital images does not exist. This thesis seeks to bridge the gap by presenting an analytical approach to digital image authentication, and providing a platform for interpretation. A framework will be proposed that will incorporate the analysis of many different features of a digital image file.

This work is not intended to supplant the publications referenced in each section. The reader is encouraged to study all literature referenced in this work, which will provide a more in-depth understanding of the mathematics and principles involved for each type of analysis. This thesis discusses how to interpret the results from many different types of techniques. These techniques are explained on a basic level to provide an understanding of the concepts behind each authentication method. The goal of this work is to provide a strong foundation for forensic image analysts, to help them evaluate their findings when making decisions about the authenticity of digital image files. This is especially important when these matters have bearing on a person's civil liberty in a court of law.

1.1 Forensic Principles

A crime is any unlawful act by a person, intentionally or by negligence, to cause serious offence to a person or group of people. When a crime is committed, investigators are left with only fragments of a larger picture. These fragments, known as evidence, are the only indications of the events that transpired. The reliability of evidence in a court of law is dependent upon how the evidence is handled, how it is analyzed, how it is interpreted, and how it is presented. The fundamental principle behind digital and multimedia forensics is maintaining the integrity and provenance of media upon seizure, and throughout the analysis and handling process [2]. To this end, each stage of evidence processing should follow best practices to ensure evidence integrity and admissibility into a court of law. The stages of evidence processing are:

1) Occurrence of the Crime
2) Recovery of Evidence
3) Analysis of the Evidence
4) Interpretation of the Analysis Results
5) Presentation of the findings

In forensic science, evidence is based on a principle known as 'Locard's Exchange Principle,' which states that whenever two objects come in contact, there will always be an exchange [3]. This principle is one of the main foundations of forensic sciences when applied to physical evidence. When two objects interact with each other, they exchange energy and traces are left on the contact surfaces. While the traces may be small, something is always added and something is always removed. The role of forensic analysts is to uncover these traces in order to help reconstruct a larger picture about the activities or events under investigation.

Physical evidence is any material substance in a crime scene expected to help provide clues in an investigation. The significant characteristic of physical evidence is that it involves some physical interaction with an object, which leaves a chemical trace of the objects involved. This can include tire marks, guns, paint, fibers, or biological traces of human interaction on a chemical level like blood, semen, fingerprints, or DNA. In this aspect, Locard's Exchange Principle can be applied very easily because two physical objects have come into direct contact with each other, but in this age of technological development, other principles have to be applied to digital evidence.

Emails, digital photographs, audio files, computer files and other information stored digitally, constitute a growing pool of evidence that is being used to help build cases in court today [4]. Digital evidence is defined as any "information of probative value that is stored or transmitted in a digital form." [5]. While this type of evidence could be considered physical, the fact is that this information is stored in a binary format and is only accessible by computer programs. There are many issues surrounding the admissibility of digital evidence into a court of law, specifically because binary files can be modified and duplicated, largely with no, or very difficult to find, indications [6]. Therefore, special attention must be made concerning the recovery techniques, tools, handling, analysis, and preservation of digital evidence for admittance into a court of law.

During the evidence recovery stage, steps should be taken to ensure that nothing is lost, added, damaged, or altered. Special precautions must also be taken to ensure that the evidence entrusted to forensic analysts is not changed in any way. For example, because digital information can be easily altered, best practices dictate the use of hardware and/or software write blockers when dealing with digital storage mediums [7]. In order to preserve evidence integrity in the face of court challenges, standards must be maintained concerning the handling, storage, and processing of digital evidence. These standards are complicated due to the multitude of devices that store information digitally, and should only be gathered, and processed, by qualified experts.

However, due to the inherent nature of digital evidence, sometimes it may be necessary to alter the evidence, slightly, in order to retrieve the information that is requested. One example is the imaging of a computer hard drive that has been encrypted. If the computer is turned off, the evidence that is being sought may become unattainable. In this instance, the computer would be left on for imaging. The computer will then record all traces of the examiners actions in certain logs. Steps such as these should be fully documented, because the integrity of the evidence could come into question [7].

In court proceedings, the "best evidence" clause in the Federal Rules of Evidence 1003, states that "if data [is] stored in a computer or similar device, any printout readable by sight, shown to reflect the data accurately, is an 'original'" [8]. This rule also extends to the handling and seizing of digital evidence. If it can be shown that a competent digital forensic expert took all precautions to maintain integrity, then the evidence being rejected by a judge can be mitigated.

During the analysis stage, examinations should follow a scientific method using validated techniques that ensure accuracy, repeatability and reproducibility. Accurate findings include analyses that are free from bias introduced by non-calibrated instruments, poorly designed experiments, and examiner expectations. This ensures that precision and exactness are maintained. Also, the scientific method ensures that the same, or similar, results can be repeatable by the same individual, or reproducible by a comparably trained person.

Principles during the interpretation stage are applied to the results of the analyses. Interpretation is the process of evaluating the data, comparing it to known findings, and defining meaning. The principle of individuality states that while two objects, or phenomena, may appear similar in all aspects, no two are exactly the same [9]. When imaging hard drives or copying digital files, accuracy of the copies play a crucial part of admissibility process. Because digital information is represented as a finite set of numbers, digital files can be easily and exactly copied, which makes reproductions indistinguishable from the original. This introduces the principle of comparison, which states that if two objects have no unexplained, forensically significant differences between them, they are said to match [10]. This principle is used largely in image authentication when determining if a digital image file was created using the same process as another image, i.e. from the same digital camera.

The final stage of the evidence processing concerns the presentation of the analyses findings. Conclusions should be supported by appropriate, peer-reviewed literature, and data that is complete. Intentionally withholding data that is at odds with the conclusion is a serious threat to the integrity of the forensic field. For this reason, forensic scientists should follow a code of ethics that is governed, not only by the individual, but also by a group of professionals in the same field [11]. This encourages honesty in scientific analysis, and accountability in the event of serious transgressions. Membership in these professional societies also promotes professionalism, education, competency, and integrity.

Application of these principles to digital evidence presents some challenges, which are being defined in the courts today [6]. In the past, digital evidence could be gathered from a single computer machine, but now digital evidence can be spanned across multiple devices in a corporate network, email, social networking sites, cell phones, GPS devices, digital cameras and much more. This means that any digital system can potentially have data that may be

relevant to a case. This can lead to a massive amount of information that has to be searched, sorted and analyzed for relevancy.

In the case of digital imagery, evidence is being supplied from cell phones, computer cameras, security cameras, and hand held digital cameras, in the form of still and video images. The question of how the principles presented in this section apply to digital photography is addressed in the following sections.

1.2 Digital Photography

Photography has been around since the early 1800's when Joseph Nicephore Niepce made the first photographic image with what was known as a *Camera Obscura* [12]. This device had, at one end, a tiny hole that focused the entering light onto a screen, which at the time was used for viewing, drawing and entertainment purposes. Joseph Niepce made the first photograph by placing an engraving over a metal plate coated with a light sensitive chemical, and then exposed the plate to light. He developed the image by placing the metal plate in a solvent. The areas where the light had been allowed to react with the light sensitive chemical changed color, while the areas that had been covered in the shadow of the engraving did not. This type of photography required many hours of exposure to create an image, and would fade shortly after being developed.

Film-based photography has been the preferred medium for capturing images in the 20th century. Images are captured using a thin a sheet of plastic that has been treated with a silver-halide emulsion. This emulsion is made up of numerous silver halide crystals that are sensitive to light. Each crystal is turned into a small piece of metallic silver as the number of photons that react with it increase. The sizes of the individual crystals determine how much detail will be captured in the process. Larger crystal sizes typically mean that less detail will be recorded onto the film, and vice-versa. After being developed, this film is referred to as a 'negative.' The resultant image is an inversion of the positive, or normal, image, where dark areas appear light and light areas appear dark. Similarly, in color negatives, red appears as cyan, green as magenta and yellow as blue. In order to revert the image to its original, positive state, the negative must be enlarged and projected onto another light sensitive paper. This paper is developed and made into a print.

Towards the end of the 20th century, digital cameras started to replace traditional film based cameras, and around 2003 it was reported by many

camera manufacturers that digital camera sales exceeded those of film cameras [13]. Developments in technology have made digital cameras more cost effective, efficient, and cheaper than its film based counterpart. Digital photography is the process of capturing visible light from the electromagnetic spectrum, and saving it to a digital medium. The primary function of a digital camera is to convert an infinite range of light intensity levels into a digital medium that has a finite amount of binary values.

The digital photography model is illustrated in Figure 1. First, light from a scene [Figure 1(a)] is focused into the camera by an optical lens [Figure 1(b)]. Lenses help control how much of the scene and the amount of light that enter the camera. Lenses can be roughly divided into three categories: wide-angle, macro, and telephoto. Wide-angle lenses are small and have short focal lengths. These types of lenses are used to capture a wide arc of the scene content. Macro lenses are long, and have a longer focal length. These are commonly used to capture content that is very close to the lens and help produce an image that is greater than life size. A telephoto lens is another lens, which has a focal length longer than the actual length of the barrel. These lenses are used to capture images from far distances.

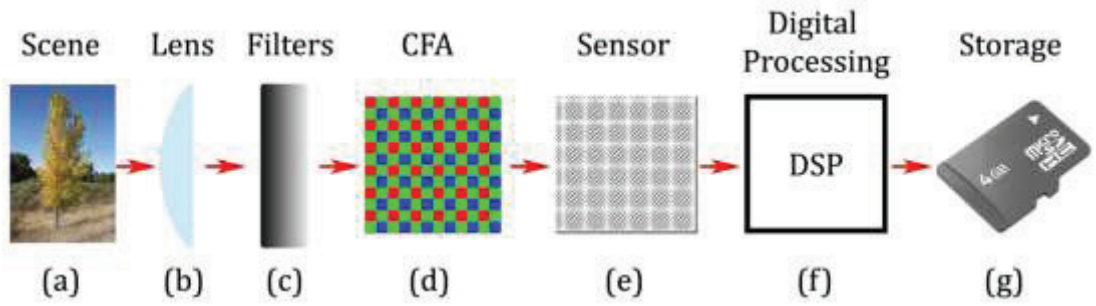


Figure 1 Digital Image Pipeline
This diagram represents the process for creating an image with a digital camera.

Before the focused light is recorded, it passes through a series of filters that prepare it for conversion into the digital domain [Figure 1(c)]. One filter is the anti-aliasing filter, which blurs the image slightly to prevent spatial frequencies greater than the resolution of the sensor from being recorded. The artifacts of aliasing can include distortion or artifacts not present in the original image. In addition to the anti-aliasing filter, an infrared (IR) filter is also needed

because digital sensors are extremely sensitive to IR light [14]. The filtered light continues through to the image sensor.

The imaging sensor is one of the most important components of the digital camera. The two main types of sensors are the Charged-Coupled Device (CCD) and the Complimentary Metal-Oxide-Semiconductor (CMOS). While the underlying technologies of the CCD and CMOS chips differ, the fundamental operation of each is the same. These sensors, which contain many photon-sensitive elements called 'pixels', convert light intensity levels into electronic voltages. However, a pixel can only distinguish between light intensity levels, making it a monochromatic component. Since pixels are not color sensitive elements, a Color Filter Array (CFA) is used to help distinguish between the different frequency ranges of light [Figure 1(d)].

The CFA is a mosaic of color filters that separate the light by color frequency onto each discrete pixel, and is mounted directly onto the image sensor. There are many different types of arrays, but the most commonly used is the Bayer RGB color filter array [Figure 2]. This filter consists of a mosaic of red, green and blue filters. Each sub mosaic covers four pixels in a 2x2 matrix, each containing one red, one blue, and two green filters. After the light passes through the CFA, the digital sensor captures the information [Figure 1(e)].

The sensor consists of many rows and columns of pixels. The number of pixels in each row M, and the number of pixels in column N, is what determines a camera's resolution, expressed as M x N. Digital imaging operates on a binary principle, meaning there are a finite amount of values that can be used to express scene content. The number of possible intensity values available is dependent on the bit depth of the imaging system. For example, an 8-bit system has the ability to express light intensity over a range from 0 to 255, or 256 discrete values. The value of 0 represents total lack of light, or black, while a value of 255 represents a total saturation of light, white. The sensor is responsible for converting light intensity levels from a continuous signal to a finite amount of intensity values through a process known as quantization. The quantization process essentially rounds intensity levels to the nearest integer values. While this process introduces small distortions into the image, finer detail by using larger bit depth can minimize these distortions.

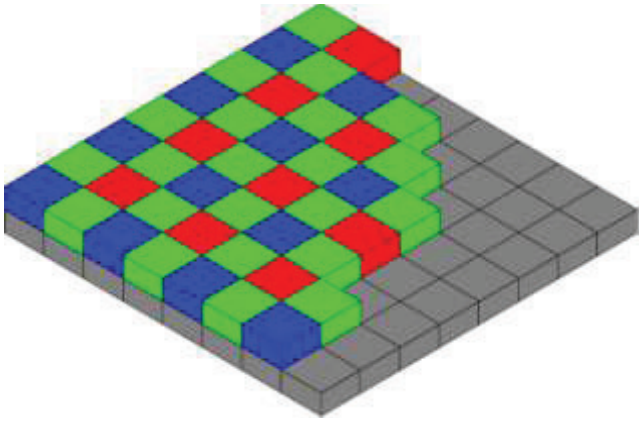


Figure 2 Bayer RGB Color Filter Array
Illustration of the Bayer CFA, which contains a mosaic of red, green, and blue filters mounted onto the sensor matrix. Each pixel of the sensor records intensity values for each color range at each location.

After light intensity is converted by the sensor for each color, a 'demosaicing' algorithm is needed to compute color values for each of the three primary colors represented at each pixel location using a process known as interpolation [Figure 1(f)]. The process converts the 1-layer grayscale matrix, into a color image. After conversion to the RGB color space there are three color layers: one for red, one for green, and one for blue [Figure 3]. The demosaicing algorithm is different between manufacturers and can vary between differing models from the same manufacturer.

Next, a number of operations are performed by the internal camera processor, which can include white balancing and gamma correction [Figure 1(f)]. The camera processes the sensor information and creates a digital image file for storage onto a digital memory device [Figure 1(g)]. Image file types can include raw sensor data (.RAW, .CR2, .NEF, etc…), lossless codecs such as tiff (.TIF or .TIFF) and bit maps (.BMP), or lossy codecs like the JPEG File Interchange Format (.JPEG or .JPG), which uses the JPEG compression standard.

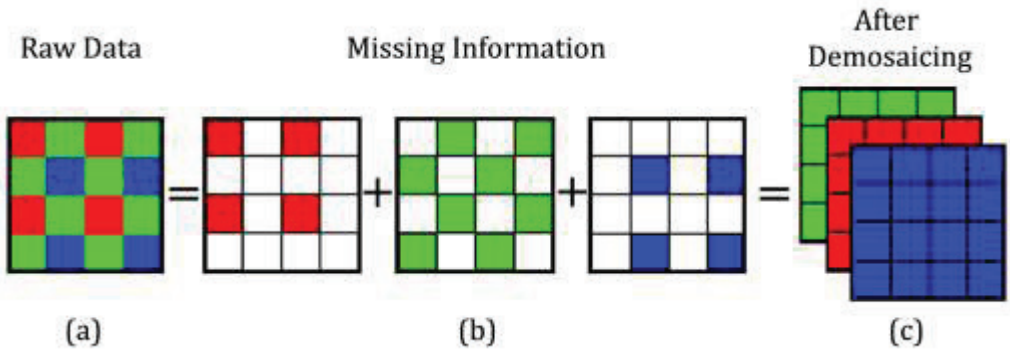


Figure 3 Illustration of Demosaicing
The sensor produces a 1-layer matrix of light intensity values for each color as determined by the CFA (a). The 1-layer matrix is separated by color onto three different matrices with missing values (b). The demosaicing algorithm interpolates missing color values by using the known values of the surrounding pixels captured by the sensor.

Digital images can be quickly copied, transferred, and duplicated with no loss of information or image quality. Digital copies, or clones, can be indistinguishable from the original file. These files can be sent electronically over great distances in a matter of seconds to friends and family. Photographers can quickly process their images using imaging processing software, and no longer have to wait for film stock to be developed and printed in specialized darkrooms. For newsprint, images can be taken and immediately inserted into the paper for late breaking news. Image files can be backed-up on multiple disk drives as opposed to protecting a single negative, which is susceptible to damage and destruction. While there are advantages and disadvantages of film and digital photography, there can be no question that digital photography provides more convenience and flexibility than its film-based counterpart.

However, in a judicial setting, this convenience and flexibility could have serious implications when it comes to evidence admissibility. Digital files can be easily manipulated by almost anyone with access to a computer and some degree of image processing proficiency. While the quality of the manipulation is dependent on the skill and knowledge of an individual, there may be a need to verify the authenticity of images in the absence of an eyewitness for use in legal proceedings [15]. The processes, however, for determining the authenticity of

film based images and digital images are different. As such, the remainder of this thesis will only discuss the processes and techniques relevant to the authentication of digital images.

1.3 Digital Image Analysis

The art of image manipulation is nothing new. Image forgeries were perpetrated shortly after the invention of photography itself. Manipulations are noted as early as 1840 when Hippolyte Bayard produced the first fake picture and caption combination [16], by staging an image and writing a false caption to explain the context. Techniques quickly evolved to include double exposure of the film negative, painting the negative, and compositing multiple images. One of the most famous images of Abraham Lincoln is actually a composite of Lincoln's head atop the southern politician, John Calhoun [17]. In fact the use of photo manipulations by governments for political gain, intelligence and propaganda has been well chronicled [18]. Government officials, particularly in totalitarian regimes, have used photo manipulation techniques to alter historical images to reflect a version of the past that they want remembered. Political leaders such as Stalin, Hitler, and Mao Tse-Tsung are well known to have removed people from images once these individuals had fallen out of favor with them [19].

More recently however, there has been an influx of digital image forgeries submitted to respected publications by photojournalists [16][19]. Journalistic publications have in turn responded with severe punishments to photojournalists who have been found to have tampered or staged images. While many early manipulations dealt with traditional film photography, there was a growing concern as early as 1988 that computer manipulations posed a serious threat to the integrity and credibility of photographic content [20]. This was at a time when specialized equipment, i.e. cameras and computers, was confined to a relatively small group of people with the proper technical skills.

Due to the low cost of manufacturing digital cameras, the market has been flooded with numerous digital devices. Digital cameras are now integrated into almost all mobile phones and personal computers, and people are using them. Facebook alone has over 3 billion photos uploaded to its website every month [21]. The growing use of digital images has also prompted the development of numerous image-processing software programs, many of which are free to the general public. Consumers are now able to make drastic changes

to an image with no perceptual indications of alteration. As more images are used in courtrooms, the necessity to verify the integrity of image files may become more paramount.

Image analysis is defined by SWGIT as "the application of image science and domain expertise to interpret the content of an image and/or the image itself in legal matters" [22]. There are three tasks of image analysis that are presented by the SWGIT document, two of which are relevant in this thesis. The first, *interpretation*, is the application of image science and expertise to derive a conclusion about the contents in an image, or the image itself. This is usually accomplished by using a combination of statistical and cognitive evaluations of analysis data [1]. While statistical analyses can be of great use in image authentication, they are not possible for all aspects of an image. Therefore, conclusions are determined using a cognitive evaluation. This process is accomplished by combining observed features with *a priori* knowledge to determine the best explanation of the image elements.

The second task of image analysis critical to this paper is the *examination*, which is used to extract information, characterize image features, and interpret the image structure. Largely, the quality of images given to image analysts is not ideal due to lighting conditions, distance of an object from the camera, resolution, camera focus and others variables. For these reasons, the majority of work done by an image analyst starts with the processing of an image to help bring out details so an investigator can extract data from it. This type of processing is usually referred to as "enhancement" throughout the field, which implies a changing of the evidence for improvement. Given that enhancement is used to help make content more clear and easier to understand, "clarification" is sometimes used as an alternative description to express this idea.

The services provided by an image analyst are photogrammetry, comparison, content analysis, and authentication [1]. Photogrammetry is the process of making precise measurements from objects in a photograph. It is commonly used, but not exclusively, to determine the height of suspects by comparing their geometric relationship to objects with known dimensions in an image. Photogrammetry can also be used to demonstrate alteration in an image by determining lighting, shadow and perspective inconsistencies [23-25]. The use of photogrammetry for detection of alteration lies in the physics based area of image authentication, which is outside the scope of this paper and will not be discussed. Although, it should be noted that physics based techniques are very valuable tools for image analysts.

Image comparison is the examination of two or more objects to ascertain the correspondences and/or discordances between them to determine identification or elimination [1]. Image comparison can be used to evaluate the similarities shared, or differences between objects in two or more images, which can include suspect faces, weapons, clothing, vehicle descriptions and others. Image comparison is also used to exclude objects from a suspect pool, narrowing down the possible number of objects that share similar characteristics. Comparative analysis is achieved by classifying similarities into three descriptors: class, limited-class, and unique. The *Class* descriptor is a characteristic that is shared by a large number of objects in a group. An example of a class descriptor would be a red sedan. Obviously, this description could describe a multitude of different makes and models of red sedans, which make this type of identification a very general one. *Unique* descriptors are characteristics that are individual to an object within a class or group, and can be used to uniquely identify a particular object. An example of a unique characteristic is the vehicle identification number for the red sedan. This now makes it possible to identify the exact car, because no two cars share the same vehicle identification number.

While general and unique descriptors describe two very different characteristics of an image, another identifier has been proposed to bridge the gap. A *Limited-class* descriptor is a characteristic that exists somewhere between the class and unique descriptors. Limited-class identifies a subset of a class but is not unique enough for individual identification. An example of this would be a low-resolution image of a red sedan with a large dent in the passenger door. While the number of sedans that share the characteristics of being red and having a large dent in the passenger side have been greatly reduced, positive identification of the car is still not possible. However, when two objects start to share a larger quantity of limited-class characteristics, the weight of each characteristic on the comparison also increases.

In addition to corresponding attributes, items may also contain discordances. These characteristics are used to determine the true nature of differences between two objects [1]. *Explainable* differences are the result of identifiable features, like those caused by image processing or changes in lighting conditions, which cause known features to seem different under dissimilar circumstances. *Unexplainable* differences are discordances that exist but the reason for, or significance of is unknown. Unexplainable differences can be explained, but further research or expertise is usually required. *Exclusionary* differences are those characteristics that represent a true discrepancy between

two objects that preclude them from being the same. Such differences establish elimination. An example of an exclusionary difference would be an image of a red sedan compared to an image of a red corvette. The differences, such as the body shape and contour, are exclusionary differences.

Content analysis is the evaluation of an image for the purposes of drawing a conclusion about it. This analysis can be applied to the process of how a digital image was created by evaluating attributes and characteristics of the media information and digital file. Analyzing the content of an image can be helpful in determining if a particular camera could have created an image file.

The fourth category, *authenticity*, is used to verify that the "information content of the analyzed material is an accurate rendition of the original data by some defined criteria" [1]. Authenticity is used to ensure the integrity of digital images by determining that the image content has not been altered to modify the perception or meaning of the events depicted. For courtroom purposes, image authentication can be determined by witness testimony claiming that the photo is a fair and accurate portrayal of the scene [15]. The person who took the image, or one who witnessed the scene that was depicted can provide this testimony. In the absence of such a witness, an expert can be called upon to evaluate the provenance of the image by analyzing the image content and the details surrounding its acquisition.

Forensic image authenticity is defined by SWGIT as "the application of image science and domain expertise to discern if a questioned image or video is an accurate representation of the original data by some defined criteria" [26]. Furthermore, the authenticity of a digital image can be defined as '*an image made simultaneously with the visual events it purports to have recorded, and in a manner fully and completely consistent with the method of recording claimed by the party who produced the image; an image free from unexplained artifacts, alterations, additions, deletions, or edits'* (adapted from the Audio Engineering Society's definition of audio authenticity [27]).

'*An image made simultaneously with the visual events it purports to have recorded.'* What this statement refers to is when the image file was created and the accuracy of the events depicted in the image. It infers that the digital image file was created at the exact same moment the content was captured. Digital files, however, can be copied with no loss of information, making the copies indistinguishable from the primary image. While these copies may not have been made simultaneously with the actual events, the integrity of the image content is still maintained because digital files can be copied exactly with no loss of information. Another inference made by the above statement is about the

events it purports to have recorded. This is cause for concern if the events depicted were falsified, or fabricated, to show incorrect or misleading information. While the events in the image recorded this way would not be an authentic representation of the scene depicted, the image file itself would be authentic because it was created simultaneously with the falsified events. The integrity of the events recorded is an area of image analysis not covered in this publication.

'*In a manner fully and completely consistent with the method of recording claimed by the party who produced the image.*' This statement refers to the equipment that created the image and the digital file that is comprised of the digital information. Camera specifications differ between manufacturers and even between models of the same manufacturer. As such, different cameras create digital images differently. File format and file structure are key in this area of authenticity. Characteristics of interest would be the image resolution, bit depth, image file type, file size, and information embedded into the digital file. When the image acquisition device is known, exemplar images should be made for comparison. The effects of the camera settings on exemplar images should be checked against the unknown image to determine if the suspected camera could have produced the image files.

'*An image free from unexplained artifacts, alterations, additions, deletions, or edits.*' The key word in this phrase is 'unexplained.' There can be artifacts such as JPEG compression, or date and time stamps, which are explainable as being caused by the normal and routine operation of the image device. JPEG blocking for instance (explained in chapter 2.2.1), is cause by the compression process, and while it does not thrust doubt onto the integrity of evidence, it can however obfuscate fine details in an image. Malicious alterations, edits, and deletions are used to change the interpretation of an image. This can be especially important in criminal and civil cases when the events depicted have been changed to modify the perception of the events in order to mislead the judge and/or jury.

One important aspect to note is that analytical techniques can only provide positive proof of image tampering. The simple fact is that is extremely difficult (if not impossible [28]) to prove a negative, i.e. prove that an image is free of modification. Even if an analyst were to apply all known techniques to uncover all known forms of manipulation in an image, the possibility may exist that an unknown manipulation technique was applied to the image that left no trace. In which case, there would be no positive indication that the image was manipulated. The best that an analyst can do in these types of cases is to search

for indications that support the proposition that the image was generated without modification. Nevertheless, if indications of alteration do exist, it is important that analysts look at the image from every conceivable angle to reduce the possibility that these traces go uncovered. The simple fact is, however, that it may not be feasible, or even possible, to perform every technique on every image due to limited manpower, time, or financial resources. Therefore, it is important to apply what resources an analyst has in the most efficient way possible. In order to do this effectively and analyst must be resourceful with the tools and knowledge that are available to them.

1.4 Summary

There can be no doubt that technology is integrating itself into the everyday life of virtually everyone around the world. As new technology emerges, the cost of the equipment will decrease even further than it is now, making the most complex computer devices, available to the general public. In the case of digital images alone, there are billions of photos uploaded to the Internet every month. With so many images, some are bound to make their way into the courtroom. While traditional and digital images both serve the purpose of recreating a snapshot of time, digital images exist by a set of rules and regulations that are not just grounded in the physical world, but in the binary language of computers and mathematics. Traditional film photography has firm roots in the legal arena when it comes to admissibility and authenticity, however, digital images are being subject to the same rules and regulations even though they exist in a digital format. With image processing software readily available to anyone with a computer, alterations can easily be made.

Digital images are a product of computers and mathematics, they behave in a predefined and predictable way. Any alteration will affect the balance and change these mathematical relationships. An image analyst just needs to know what these traces look like, and where to look for them. Chapter 2 of this thesis discusses the current literature from the image authentication community, presenting strengths, limitations and interpretation of each technique. A framework that combines many of the aforementioned techniques will be presented in chapter 3. Case studies will also be included using the proposed methodology to illustrate the benefits of a global approach towards authenticating a digital image. Finally, chapter 4 will contain the concluding remarks.

2. Review of Image Authentication Techniques

This chapter serves as a review of some of the most modern digital image authentication techniques in use today. Techniques that determine content authentication, such as lighting inconsistencies, historical inaccuracies, anachronisms, and physics based forensics will not be discussed. Instead, this paper focuses on techniques that determine authenticity by investigating the digital information that represents a digital image file. While this thesis does not go into great detail about the methods proposed in the papers discussed a general understanding of the mathematics and concepts are explained in this chapter. Analysts who desire a deeper understanding of the math and science are encouraged to read the papers reviewed in this document. Furthermore, this is in no way a comprehensive review of all analyses. If every technique were to be discussed, this thesis would contain volumes of information, which is far beyond the scope of this paper. While many important papers are reviewed in this section, due to the vast quantity of work available, the omittance of important papers is inevitable, but was in no way done intentionally.

When a digital image is acquired, the information is stored onto a storage medium, like a hard drive or memory card. This information can only be translated into a visual image people can make sense of by displaying it onto a monitor. However, there are more characteristics to a digital image than just the image information. Digital images are a product of mathematics and computer language, both of which operate in an expected way. Image authentication is about determining if any aspects of this order have been disturbed. Alterations can be made to the media information, the digital file, or inconsistencies in the events surrounding its capture. Therefore, the author proposes that the analysis of digital image files be divided into four categories: file structure, global structure, local structure, and source identification. The techniques useful in each category will be discussed in the following sections.

File Structure analyses investigate the format of the digital information such as the file type, EXIF, hex data, and MAC stamps. Digital cameras create files in a particular way, each with its own unique structure. Information is embedded into image files, which can be distinct between manufacturers and cameras. When computers, or image processing software, interact with the file, this structure could be altered in some way. While this type of alteration does not necessarily mean that image content has been altered, it can raise concern about the authenticity of the file.

One step deeper into the digital file is the content that constitutes the actual image media information. *Global Structure* analyses investigate the content that represents the digital image information. A digital camera operates much like a traditional camera, with one big difference. Light is converted to electrical energy by a camera's sensor, and then the camera's internal image algorithms process the information to form an image file. Thus, a digital image is a product of a mathematical process. As such, numerical relationships in the binary digits of an image are formed. Because of the way color images are created, the three color layers of an image are highly correlated to each other. When an image is manipulated, these correlations could be lost, in which case new ones will be created. By comparing suspect images to exemplars taken from the suspected camera, inconsistencies or similarities can be identified. In addition, many images are saved with the JPEG compression standard. This standard can be used in a variety of image formats including JFIF and TIFF files. The compression process introduces more relationships to an image, which can be used to evaluate authenticity. Therefore, a section on the JPEG compression process is reviewed, as well as techniques that examine the characteristics of the compression standard. These analyses take a general approach in determining alteration, but cannot identify the exact part of an image that has been altered.

To help identify areas that have been altered, the *local structure* is examined. These techniques identify alterations that corrupt the relationship between the pixel values themselves. While mathematically more complex than the other analyses, these can be powerful tools in identifying malicious fabrication.

The final chapter examines traces that are created by the digital image sensor. These traces tend to be unique to a specific camera, and can therefore be used as a digital fingerprint to identify the source of the image, or if two images were created by the same imaging device. These traces can be identified from defective pixels or by using the photo response non-uniformity of the image sensor. Characteristics observed from these types of analyses are used to determine *Source Image Identification.*

2.1 File Structure Analyses

There are many different digital image file formats in use today. Each format encodes digital information differently for storage as a computer file. Some formats, such as TIFF files, are a flexible file container that allows a wide range of compression schemes and color spaces. Since computers store information in a binary form, 1's and 0's, the file format determines how these files will be encoded, stored and retrieved from a storage medium, like a hard drive or USB drive. The image file not only contains the digital information representative of the image, it also contains a list of the contents of the file, the address of the contents contained within, instructions on how to reassemble the image, information about the image, and information about the file itself. The structure of the file can also include important information about when it was created, accessed and what programs have interacted with it.

For a forensic image analyst, the structure of the file, and the information that resides within, can provide important clues in determining authenticity and verification of events concerning the image. This section will discuss the information about the image, which includes EXIF, hex data, metadata, file structure and MAC stamps. This information is created by the equipment used to create the image and can potentially be altered by devices or software that interacts with the digital image file.

An important concept to understand is that a digital image file contains more information than just the actual media information [Figure 4]. A digital file is a container that the image information resides in, much like a can of soup. The soup is the image content, while the can is the digital container. A digital file contains information about the image that is inside, much like the label on the outside of a can. It states the name of the image, when it was packaged, the size of the file, the ingredients that make up the digital image information, and directions on how to reassemble the image. Therefore, there is an important distinction between an authentic file and an authentic image.

An authentic file is one in which the container is consistent with a file made by the imaging device, while an authentic image is one in which the image content is free from malicious alteration. The implication being, any alterations or inconsistencies in the file container do not necessarily mean that malicious alteration of the image content has occurred. For example, if you dent the can of soup, it is possible that the soup inside has not been compromised. However, any alteration in the container brings suspicion onto the content of the image.

Nevertheless, any alteration in either the container, or the image structure, is an indication that a secondary program has altered the digital image file.

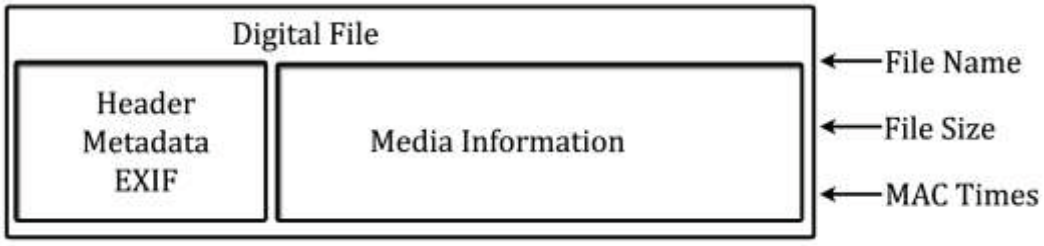


Figure 4 Digital File Structure
A digital file is a container that is made up of more than just the media information. The header, EXIF, and metadata include information about the image content. The file name, size, and MAC times are characteristics not embedded in the digital file.

2.1.1 File Format

There are many different types of image files in use today. Each is a standardized format that defines how a file is stored and retrieved from a storage medium. A digital image file can be large and the amount of disk space required to store this information is dependent on color space, bit depth, compression, and resolution. To be as efficient as possible in storing the information, a computer, or camera processor, compresses the information into as small as size as possible onto the storage medium. Consider packing a suitcase for a vacation. It is helpful to organize the contents in a way that will maximize how many objects can fit inside for the trip. There will be more space inside the suitcase if clothes are folded neatly and organized, as opposed to them being thrown in randomly and unfolded. Computers do this type of packing very efficiently. Compression schemes are divided into two classifications: lossless and lossy.

A lossless compression scheme, as the name implies, does not discard any information about the original file. It determines the most effective and resourceful way to pack the file while making no compromise on data accuracy. In other words, exactly what goes in is exactly what comes back out. For larger images files, such as color images with high resolutions, the amount of space

needed to store this information can be quite large. These types of files are preferred when detail and accuracy retention are more important than reducing file size. Common lossless file types are .BMP, and .TIF.[1] These flexible file formats can contain large amounts of image data.

A subset of the lossless compression schemes is the proprietary file type. Many of the major camera manufacturers like Canon and Nikon, create a file that is considered a digital negative. These 'raw' files, as they are commonly called, contain the raw sensor data as recorded by the digital sensor. These files require specialized software interfaces to convert the data into a useable image file. Common proprietary formats are .DNG (Adobe), .CR2 and .CRW (Canon), .NEF (Nikon), and .SRF (Sony).

A lossy compression scheme produces smaller files sizes, at the cost of image degradation. The amount of degradation depends on the file type and the amount of compression applied [Figure 5]. Lossy compression schemes can be used when quicker transmission or loading times are necessary. Browsing the Internet would not be so quick if the image files on the page took minutes to load. Compression schemes use algorithms to determine what information can be discarded without severely effecting image accuracy. However, too much compression can cause severe loss in image detail and can introduce unwanted artifacts into an image. Figure 5 shows the effect of JPEG compression on the uncompressed image "Lena." Photoshop allows the saving of an image using 13 JPEG quality levels ranging from 0 to 12, with higher numbers (level 12) producing less compressed images. Depending on the severity of the compression, the algorithm could produce an image that has unrecognizable features. The JPEG compression standard, which is used in multiple image file formats, is the most common method for compressing image data.

When authenticating an image, it is useful to determine if a particular camera can create the image format in question. Some cameras, such as some models made by Nikon, have the ability to save images as uncompressed .TIF files, whereas some other cameras cannot. In addition, the size of the suspect file should be compared to exemplar images from the suspected camera on a comparable setting under similar lighting conditions.

[1] .TIF files can also be saved using compression schemes such JPEG.

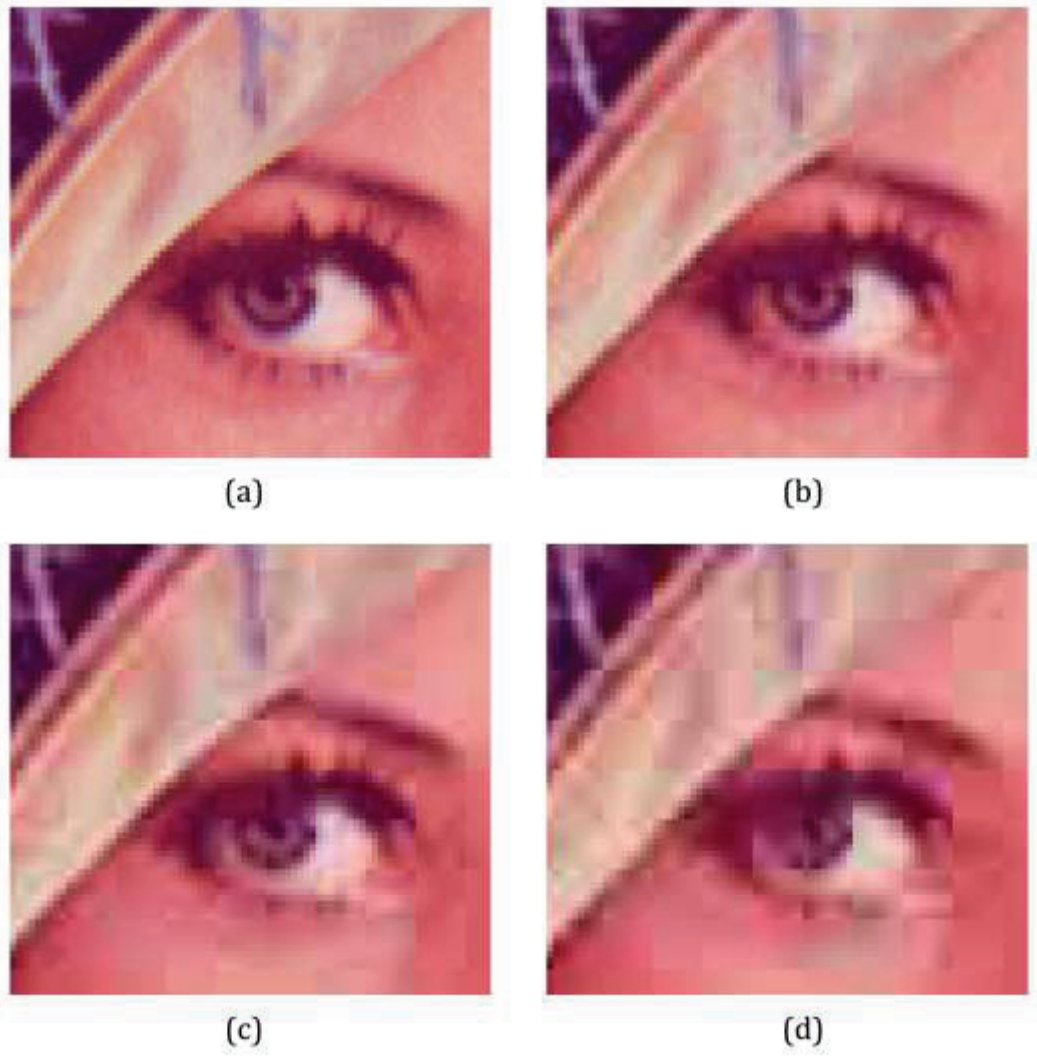


Figure 5 Degradation of Image Quality Caused by JPEG Compression
The effects of JPEG compression on a cropped portion of the image, Lena, using Adobe Photoshop. Uncompressed Image (a), JPEG Quality 6 (b), JPEG Quality 2 (c), and JPEG Quality 0 (d). Note the blocking artifacts and loss of detail as compression is increased.

2.1.2 Hex Data

In arithmetic, we are taught to use the decimal numbering system that consists of 10 digits, 0 through 9. While this type of numbering system may be easy for some of us to comprehend mathematically, this type of arithmetic is very complicated and time consuming for a computer. Therefore, computers use information in a binary fashion and work with data known as bits. A bit is short for binary digit, and consist of one of two values: 0 or 1. Bits are combined to form longer strings of data that make up a digital file. Depending on the file size, the string of bits in a file can be extremely long. In addition, this binary information is the representation of the raw digital data. It is important to note that software programs are needed to interpret this binary information. Therefore, programs require the file information to be constructed in a specific way in order for programs to properly decode them. This includes the location of markers and identifiers, in the form of specific binary sequences, which helps a program reconstruct the file.

While great for a software program, this binary information is not useable to human beings. To help interpret the binary data, the information can be converted to hex data. Hex data, short for hexadecimal notation data, is used universally in computing and is a number based system of 16 characters: 0 through 9 and A through F. A hex digit is comprised of 4 bits, and 2 hex digits represent one byte, or 8-bits of data. A byte of information has 256 possible character combinations and range from a value of 00 to FF. While hex data can still be considered raw data, the ease at which it can be navigated is increased.

Each hex string can represent a decimal number, a letter, an operation, command, or many other attributes relevant to a computer program. This raw data can be accessed by using a hex viewer, which is a software program that represents the hex values as a text display. The hex viewer converts the hexadecimal notation into text information, some of which can be understood and read by a human. Depending on the character-encoding scheme, some hex bytes are defined specific letter and symbols. While there are many character-encoding schemes, one commonly used standard is the American Standard Code for Information Interchange (ASCII) [29]. This standard includes definitions for 128 characters including numbers, symbols and the letters of the English alphabet. Most hex viewers contain a "character area" which displays the ASCII representations of each of the hex bytes.

Depending on the file size, the string of characters in a file can be extremely long. When software programs interact with this data, they

sometimes embed information in the hex data. While some programs may leave traces in the EXIF (as shown in section 2.1.3), evidence of tampering may also be found in other areas of the digital file. Photoshop, for example, leaves detailed notes about the processes used on an image, and embed that information along with the image data [Figure 6].

For instance, the symbols that make up the word 'Photoshop,' are converted to hexadecimal notation and embedded into the image file. For this reason, a search of the hex information is often useful to uncover traces left by imaging software. Depending on how the software interacts with the file, this information could be anywhere. Searching the hex data manually is not realistic since the amount of information that make up the file can be enormous and would take days if one were to search it manually. In addition, humans are prone to error when subject with such a mundane task.

A computer, on the other hand, can quickly search through the multitude of characters and return exact locations in a more efficient manner. The search criteria, however, must be defined by the user and input into a hex search algorithm. See Appendix A for a list of common search terms. The computer will then return any values that match the criteria [Figure 7]. Any return will indicate that software may have interacted with the file at some point.

Using a hex editor, hex and text data can be easily manipulated. These programs allow an individual to change the binary data that make up the computer file. However, a clear understanding of what is being changed is needed so a file does not become corrupted and unrecognizable by the computer or software. A deletion of a single hex byte can cause addresses of the information to shift. Depending on the format of the file, locations like the start of the image section, or directions on how to reconstruct the image can be lost, and therefore make the image unreadable by a computer.

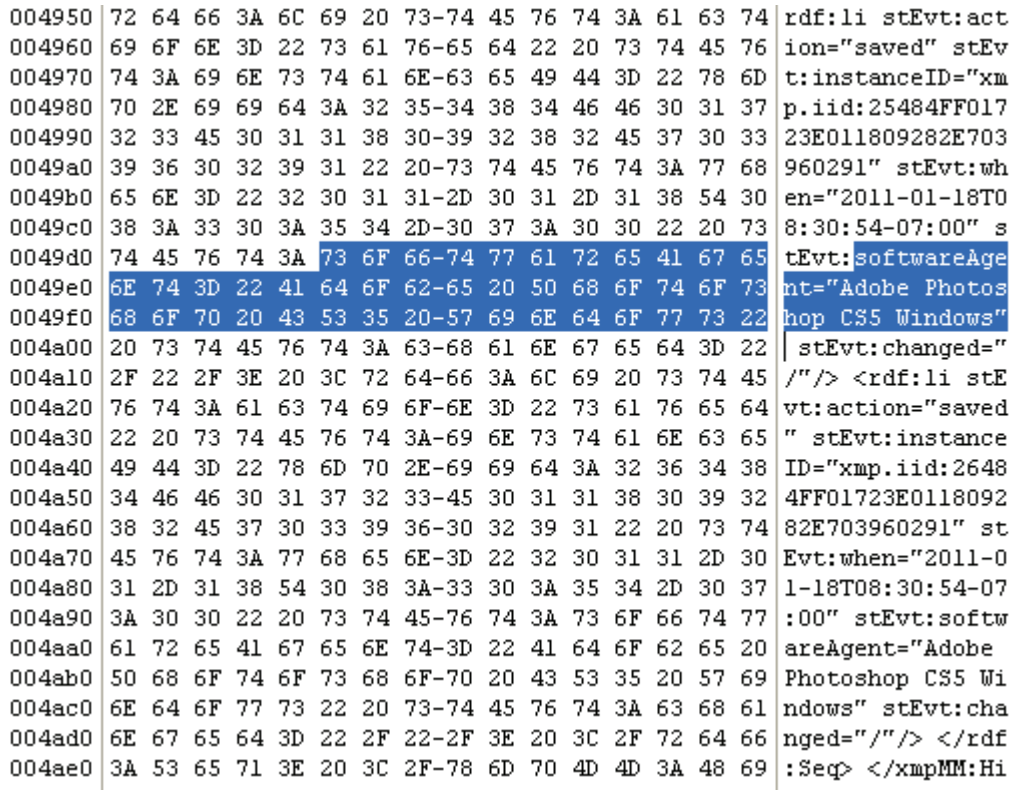


Figure 6 Hex Data From a Manipulated Image

This hex data is from an image that was processed using Adobe Photoshop CS5 and resaved. While traces of Photoshop were removed from the EXIF, additional indications were found scattered throughout the digital file.

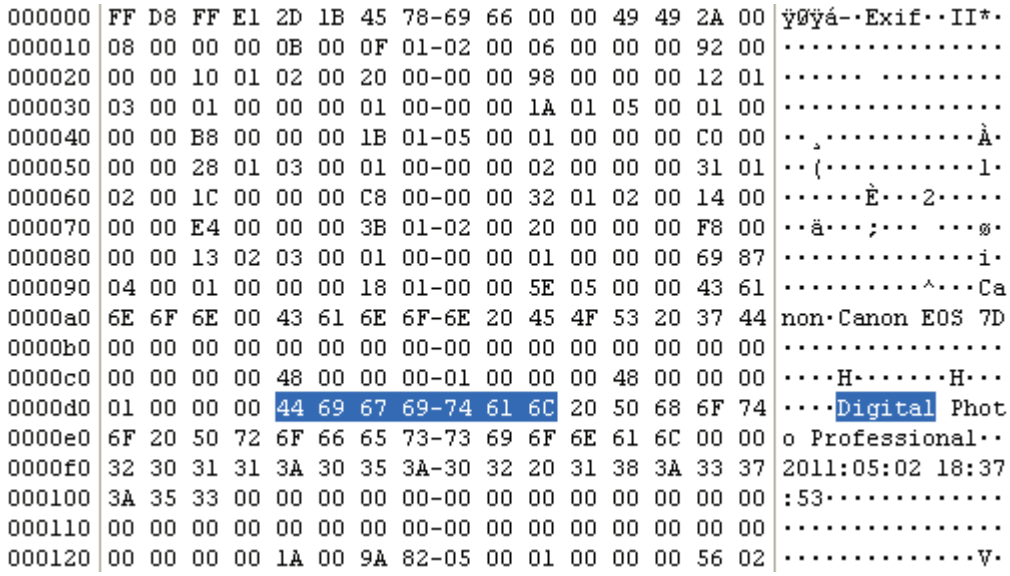


Figure 7 Hex Search

The following text string was found using a keyword search of "Digital." The image software *Digital Photo Professional* was used to simply open and resave the image. While no alteration was done to the image content, this trace was still embedded in the hex data.

2.1.3 EXIF Data

For digital photography, the Exchangeable Image File Format (EXIF) is a standard for storing information about a digital image. It is used in almost all modern cameras to record equipment model, date and time the image was taken, f-stop, ISO speed, resolution, metering, GPS coordinates, and other information relevant at the time of the image acquisition. The EXIF data is embedded with the image in the header of the digital file. In addition to digital cameras, cell phones, scanners, and software programs also use EXIF data. For forensic image analysis, the EXIF is an important part of the file structure to inspect because information in the EXIF can be used to validate information about the acquisition of the digital image [30].

The EXIF tag was adopted from the TIFF image format. Even though newer and more efficient ways of storing this data exist, EXIF continues to be utilized because it has been widely adopted by the user and implemented in

almost all digital cameras and software applications. Information embedded in the EXIF includes a number of standard fields including Filename, FileModDate, Make, Model, DateTime, and others [Figure 8].

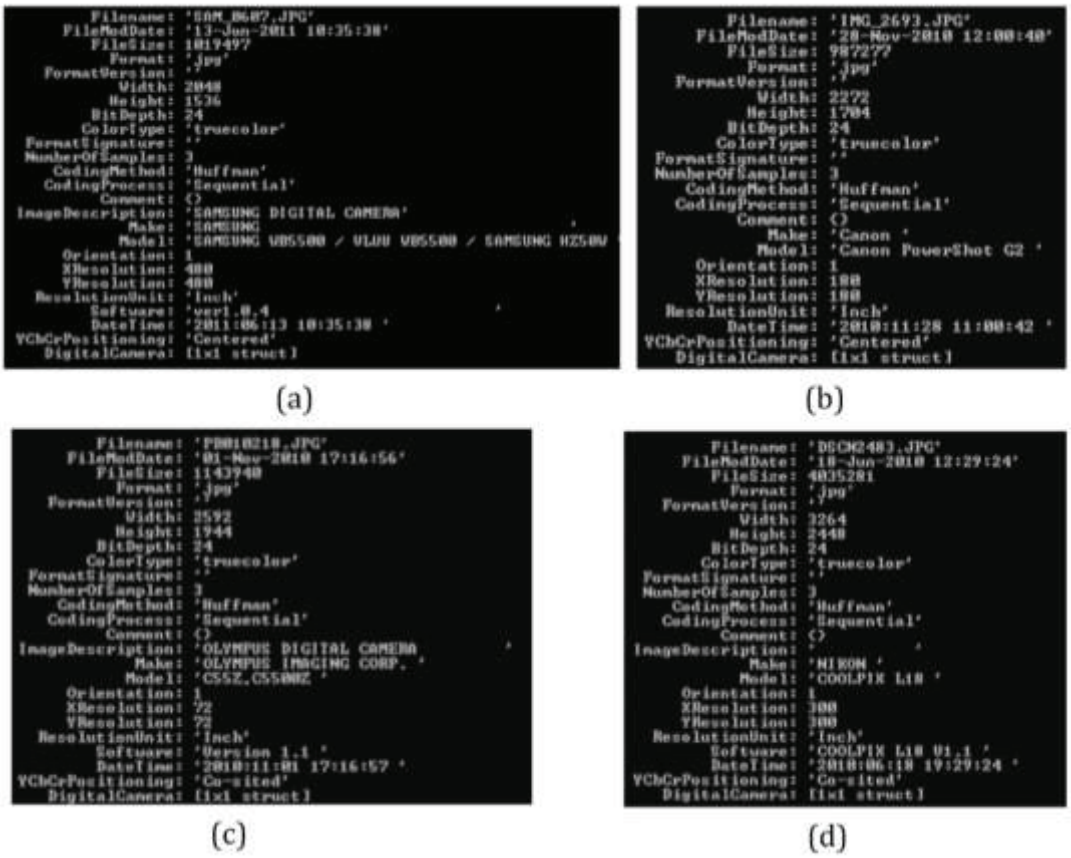
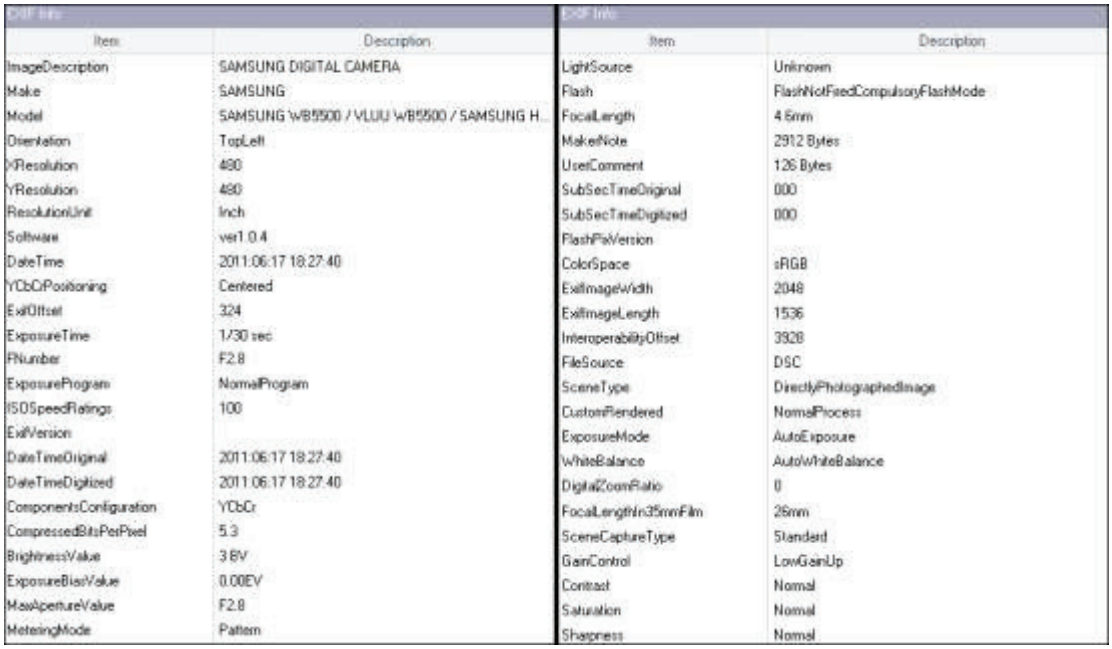


Figure 8 Examples of EXIFs From Digital Cameras
Examples of EXIF information embedded in the header by the digital camera at the time of acquisition. Samsung HZ50W (a), Canon PowerShot G2 (b), Olympus C5500Z (c), and Nikon Coolpix L18 (d).

However, developers have failed to adopt a standardized format for how to handle and encoded the EXIF information. Although the examples in Figure 8 have similar fields, *ImageDescription* and *Software* are not included in the EXIF created by the Canon PowerShot G2 [Figure 8 (b)]. While many cameras

populate the EXIF with similar information, the way in which they represent that information can vary amongst different manufacturers. For example, the *Make* field for the Samsung HZ50W contains a large gap of empty characters between 'SAMSUNG' and the last apostrophe [Figure 8 (a)]. The gap is not present in the *Make* field of the other three cameras. Closer inspection of the examples in Figure 8 reveals more dissimilarity between the EXIFs of the different cameras.

In addition, information in the EXIF can be tailored, with many camera manufacturers populating the EXIF with custom fields such as serial number, processing settings, and many others. Some values, such as artist and copyright, can be defined by the user. The placement of these fields can be difficult to retrieve and may even be encrypted so only a manufacturer's proprietary software viewer can properly decode the EXIF information [Figure 9].



Item	Description
ImageDescription	SAMSUNG DIGITAL CAMERA
Make	SAMSUNG
Model	SAMSUNG WB5500 / VLUU WB5500 / SAMSUNG H...
Orientation	TopLeft
XResolution	480
YResolution	480
ResolutionUnit	Inch
Software	ver1.0.4
DateTime	2011:06:17 18:27:40
YCbCrPositioning	Centered
ExifOffset	324
ExposureTime	1/30 sec
FNumber	F2.8
ExposureProgram	NormalProgram
ISOSpeedRatings	100
ExifVersion	
DateTimeOriginal	2011:06:17 18:27:40
DateTimeDigitized	2011:06:17 18:27:40
ComponentsConfiguration	YCbCr
CompressedBitsPerPixel	5.3
BrightnessValue	3 BV
ExposureBiasValue	0.00EV
MaxApertureValue	F2.8
MeteringMode	Pattern

Item	Description
LightSource	Unknown
Flash	FlashNotFeedCompulsoryFlashMode
FocalLength	4.6mm
MakerNote	2912 Bytes
UserComment	126 Bytes
SubSecTimeOriginal	000
SubSecTimeDigitized	000
FlashPixVersion	
ColorSpace	sRGB
ExifImageWidth	2048
ExifImageLength	1536
InteroperabilityOffset	3928
FileSource	DSC
SceneType	DirectlyPhotographedImage
CustomRendered	NormalProcess
ExposureMode	AutoExposure
WhiteBalance	AutoWhiteBalance
DigitalZoomRatio	0
FocalLengthIn35mmFilm	26mm
SceneCaptureType	Standard
GainControl	LowGainUp
Contrast	Normal
Saturation	Normal
Sharpness	Normal

Figure 9 EXIF View Using Proprietary Software
This EXIF information was retrieved using the Samsung Intelli-Studio Viewer software from a JPEG image created by the Samsung HZ50W digital camera .

The EXIF data is a fragile part of the file structure that can easily be corrupted by a device or software that is incompatible with a manufacturer's

specifications. The location of the EXIF inside a file format is determined by offset pointers, which identify the address of the information packets within an image file. Because the EXIF data can reside anywhere within the file structure, software unable to decode or encode the EXIF information properly may damage, alter or remove the EXIF when it is resaved. Figure 10(a) shows the EXIF of an image that was resaved using an image processing software program. When compared to the standard EXIFs in Figure 8, nine fields after *Comment*, including *Make*, *Model*, and *DateTime*, have simply been deleted. Other indications of altered EXIF information can be seen by experimenting with how other software programs alter the EXIF data. When an image taken by a Canon 7D, was open and resaved using Canon's digital image viewer, Digital Photo Professional, the *Software* and *DateTime* fields were updated to reflect the program used to create the file, along with the date and time the file was resaved [Figure 10(b)]. In primary images, the *DateTime* field would mirror the same information found in the *FileModDate*. One aspect to note in this example is that the program left the other fields populated with the same information generated by the Canon digital camera. This is probably due to the compatibility of how the EXIF information is used in both Canon products. In the normal operation of most cameras, the *Software* field would indicate the firmware version installed in the camera at the time the photo was taken. Figure 10 (d) is the EXIF of an image that was altered with a hex editor to remove the values in the *Software* and *DateTime* fields.

Analysis of the EXIF can reveal important information about the device that generated the digital file. Of course, the EXIF should be analyzed in conjunction with an unaltered EXIF of an image from the same make and model camera. This way, any particularities in the suspect image EXIF can be compared to that of the exemplar to determine if similarities exist between the two.

However, there are some limitations of the EXIF that should be considered when basing decisions about the integrity of an image. The EXIF information is embedded in the header of the digital files, and as such, can be altered very easily with anti-forensic software and hex editors. There exist software programs that can replace the EXIF with user-defined data. One in particular called EXIFER has the ability to replace the EXIF with information from another image [31].

Another issue concerning the EXIF data is that some fields can be defined by the user in the settings of the camera. One such area is the date and time settings. On some digital cameras, the user can specify this setting. Even if the

time was set correctly, most cameras will not account for the change for daylight savings time. The date and time offset should be determined for the device, if available, and considered in the analysis. It should be understood that the date and time could be modified at the discretion of the photographer.

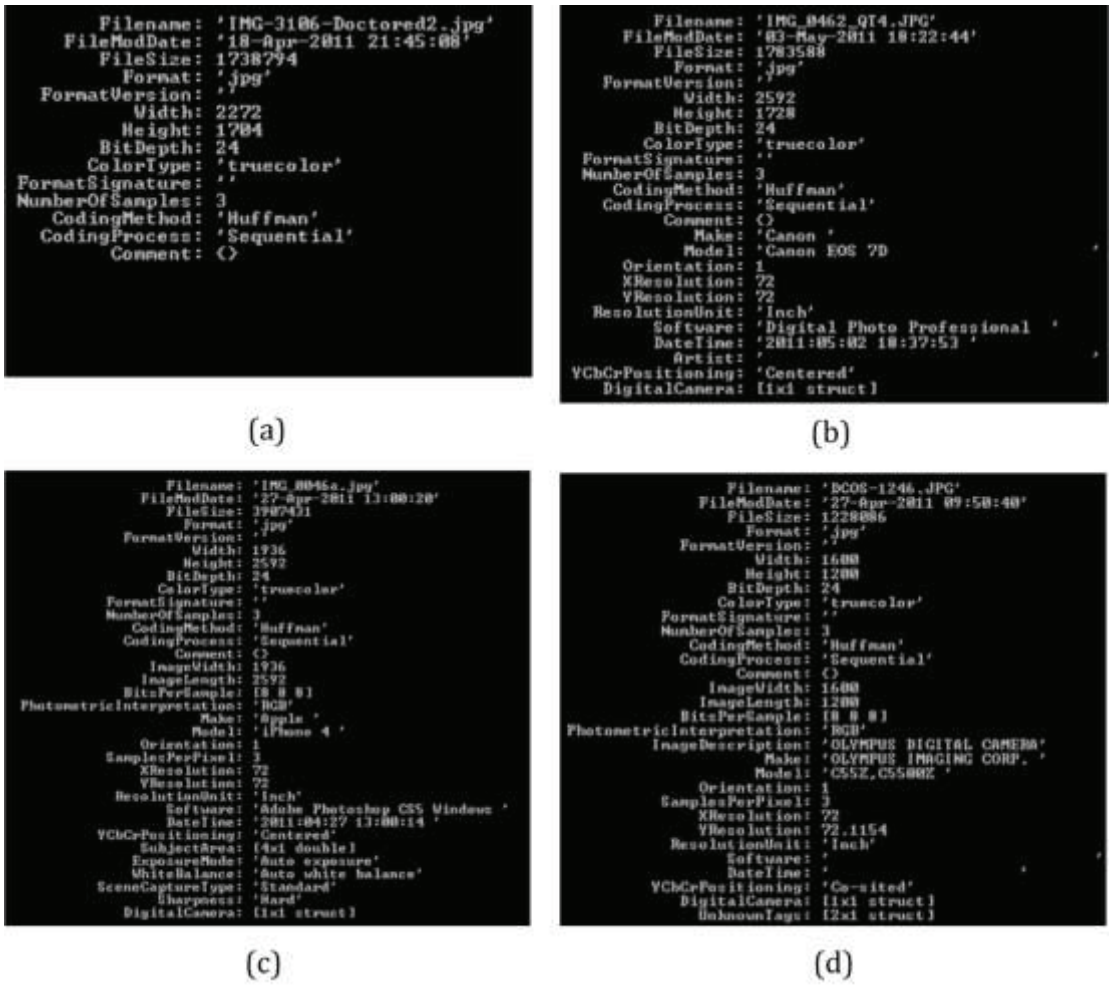


Figure 10 Examples of Manipulated EXIFs

Examples of EXIF information that have been altered by software programs. EXIF data removed by unknown program (a), Digital Photo Professional (b), Adobe Photoshop (c) and (d).

2.1.4 MAC Stamps

While not embedded in the digital image file, MAC times are part of a computer's file system metadata that record the date and time when certain events pertaining to a file occurred. MAC is an acronym for Modified, Access, and Created times. The *Modified* time stamp is updated when the contents of a file have changed. This would update when portions of a file have been overwritten by a save command. *Access* time is updated when a file has been interacted with in some way. Opening and closing a file may change the accessed time, or it may change if a virus checker scanned the file. The *Created* time stamp can be a little confusing because it does not always indicate when the file was actually created. It refers to the time the file was first created by the computer operating system. For example, the created time of a file copied from a CD-ROM, would be the date and time the file was copied onto the computer, not necessarily the date the file was created.

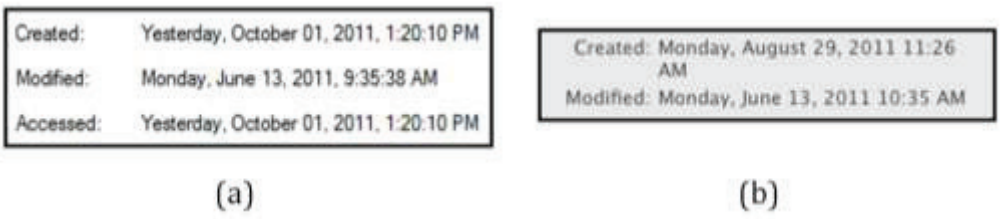


Figure 11 MAC Times
MAC time examples of the same file as represented by Windows Vista (a) and Apple OSX 10.5 (b).

The MAC time stamps in Figure 11, from the same unmodified image, are represented differently by two different operating systems. Windows [Figure 11(a)] shows all three MAC times, while Apple OSX [Figure 11(b)] does not show the *Accessed* time. The file used in the example was an image created on June 13 at 10:35 AM. Close inspection of the MAC times from both operating systems show inconsistencies with this fact. Both operating systems show that the file was modified before it was even created. Because the image had been unmodified since its creation, this modified date just happens to be the date it was created by the camera. Whereas the *Created* stamp indicates when the file was copied onto each machine. But close inspection of the *Modified* date show different time stamps. While the date is correct, June 13, 2011, the times are

different by one hour. The OSX time stamp is correct but the Windows one is not. This is caused by incorrect time settings of the Windows operating system. In this example, the time of the Windows machine was incorrectly set back one hour to show the effect on the MAC stamps of the image file.

While MAC time stamps can help in identifying dates and times, they must be regarded with caution as many events can alter these times and give a false impression of the timeline. There are also software programs that can easily manipulate the MAC times to a user defined time and date. In addition, the MAC times depend on the time settings of the computer operating system, which can be inaccurate or changed by the user.

2.2 Global Image Structure Analyses

The format of a digital image file can be considered the container, within which resides the information that consists of the media information. While the previous section's analyses investigated the structure of the container, this section considers the information of the data that represent the actual image content. Analyses of the image data as a whole help determine if the overall structure of the image deviates from the normal operations of the acquisition device. In order for a digital image file to be manipulated, it must be opened, processed and resaved. The resaved image then exhibits small deviations from an original, or 1st generation image. Modifications can alter the random distribution of numerical values in original images and introduce relationships not found in original, un-doctored images. These relationships can be uncovered using mathematics and then compared to camera exemplars.

The term primary image is used to represent an image created with an acquisition device that was only processed by the normal functions of that device [5]. In other words, how the camera treats binary representations of light from the CFA, through in camera processing such as white balance and gamma correction, to JPEG compression (if compressed), until it is finally saved as an image file onto a storage medium.

When authenticating an image, it is important to first determine if the suspected camera supports the file type and resolution combination. If it does, exemplars should be taken with the suspect camera. The analysis results of the suspect image should be compared to exemplars to determine if the results are consistent with a primary image. While the techniques in this paper provide general guidelines for interpretation, some cameras operate differently and

produce results outside of the general norm. Therefore, exemplars will help determine a baseline for comparison. Since many of the techniques reviewed in this section concern aspects of a JPEG images, a thorough understanding of the JPEG compression method is imperative. After describing the JPEG compression standard, various *Global Structure* image analyses will be discussed.

2.2.1 JPEG Compression

The JPEG compression standard was created in 1991 by the Joint Photographic Experts Group to meet the growing demand for a universal standard that could support a wide range of applications and equipment from different manufacturers [32]. It was also created to help address the fact that the storage space required for an uncompressed image can be quite large. In 1991, the average hard drive capacity was about 100 MB. Therefore, a need to reduce a file's size was required so more images could be retained on storage media. The JPEG compression algorithm can be adjusted to provide a trade-off between file size and image quality. The JPEG compression scheme is one of the most widely used standards for lossy compression. Because the JPEG compression standard is so commonly utilized in nearly every modern camera, a brief discussion of the steps involved is crucial for a proper understanding of the analyses presented in this section.

The JPEG image standard supports bit depths of 8 or 12 bits per color channels. However, the 8-bit model has become widely adopted since the standard was introduced. Some manufacturers claim to save JPEG files in 24 bits, but this is slightly misleading as each of the three color layers are only 8-bit, 8 x 3 = 24. For each 8-bit layer, there are only 256 discrete values used to express light intensity levels at each pixel location. The efficiency of the JPEG compression is achieved by using the Discrete Cosine Transform (DCT) [33]. The DCT introduces no loss to the quality of an image, instead it converts it from the spatial domain into the frequency domain, where it can be more efficiently encoded.

The steps for encoding a three-channel color image using the JPEG compression standard are as follows [32]:

1) The image is converted from the RGB color space to the Luminance/Chrominance (YCbCr) color space.
2) The image is separated into non-overlapping 8 x 8 pixel blocks.
3) Values in the block are converted from unsigned integers (0 to 255) to signed integers (-128 to 127).
4) Each block is converted from the spatial domain to the frequency domain by DCT processing.
5) The resultant values are quantized.
6) The DCT coefficients are then losslessly encoded.
7) A header is attached to the resultant data stream.

Step 1, the conversion of the color space into the (YCbCr) color space is the first step in the JPEG compression process. This separates the image into one luminance component and two chrominance components. This is desirable because the human eye is more sensitive to brightness information, luminance, and has less spatial sensitivity to color changes, chrominance [34]. Because of this particularity in human vision, information for the chrominance channels can be greatly reduced with no noticeable loss of image quality. The chrominance channels (CbCr) are generally reduced by a factor of two relative to the luminance channel [35].

Step 2, the image is then separated into non-overlapping 8 x 8 pixel blocks for processing [Figure 12 (a)]. Step 3, the resultant values are shifted from signed integers [0, 255] to unsigned integers [-128, 127]. Step 4, each block is then individually processed by the DCT and converted from the spatial domain into the frequency domain, which comprise the "frequency spectrum" of the input signal [Figure 12 (b)]. Areas of slow change, like those in a uniform sky are comprised of low frequency information [Figure 13 (b)]. Areas of rapid change, such as individual blades of grass, comprise high frequency information [Figure 13 (c)]. The concept of frequency is important because JPEG compression has a more pronounced effect on higher frequency information in the form of detail loss. The resultant values from the DCT transformation are a set of 64 signal-based amplitudes referred to as the DCT coefficients. In a color image, each color layer is processed separately as an independent image.

139	144	149	153	155	155	155	155
144	151	153	156	159	156	156	156
150	155	160	163	158	156	156	156
159	161	162	160	160	159	159	159
159	160	161	162	162	155	155	155
161	161	161	161	160	157	157	157
162	162	161	163	162	157	157	157
162	162	161	161	163	158	158	158

(a) Source Image Samples

236	-1	-12	-5.2	2.1	-1.7	-2.7	1.3
-23	-18	-6.2	-3.2	-2.9	-0.1	0.4	-1.2
-11	-9.3	-1.6	1.5	0.2	-0.9	-0.6	-0.1
-7.1	-1.9	0.2	1.5	0.9	-0.1	0	0.3
-0.6	-0.8	1.5	1.6	-0.1	-0.7	0.6	1.3
1.8	-0.2	1.6	-0.3	-0.8	1.5	1	-1
-1.3	-0.4	-0.3	-1.5	-0.5	1,7	1.1	-0.8
-2.6	1.6	-3.8	-1.8	1.9	1.2	-0.6	-0.4

(b) Forward DCT Coefficients

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

(c) Quantization table

15	0	-1	0	0	0	0	0
-2	-1	0	0	0	0	0	0
-1	-1	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

(d) Normalized Quantized Coefficients

240	0	-10	0	0	0	0	0
-24	-12	0	0	0	0	0	0
-14	-13	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

(e) De-normalized Quantized Coefficients

144	146	149	152	154	156	156	156
148	150	152	154	156	156	156	156
155	156	157	158	158	157	156	155
160	161	161	162	161	159	157	155
163	163	164	163	162	160	158	156
163	164	164	164	162	160	158	157
160	161	162	162	162	161	159	158
158	159	161	161	162	161	159	158

(f) Reconstructed Image Samples

Figure 12 DCT and Quantization Example

Source image samples from an 8 x 8 pixel block (a) are converted to the spatial domain by DCT (b). The DCT coefficients are divided by the quantization table (c) producing the normalized quantized coefficients (d). On recompression, (d) is multiplied by the quantization table (c), producing the de-normalized quantization coefficients (e). Inverse DCT is applied to (e), reconstructing the image sample (f). Note the slight differences between (a) and (f).

The DCT coefficients are separated into two types of signals, DC and AC components. The "DC" coefficient refers to the mean value of the waveform and represents the average of the input samples. The DC components typically contain a significant portion of the total energy for the image. The remaining 63 coefficients are referred to as the "AC" coefficients [Figure 14].

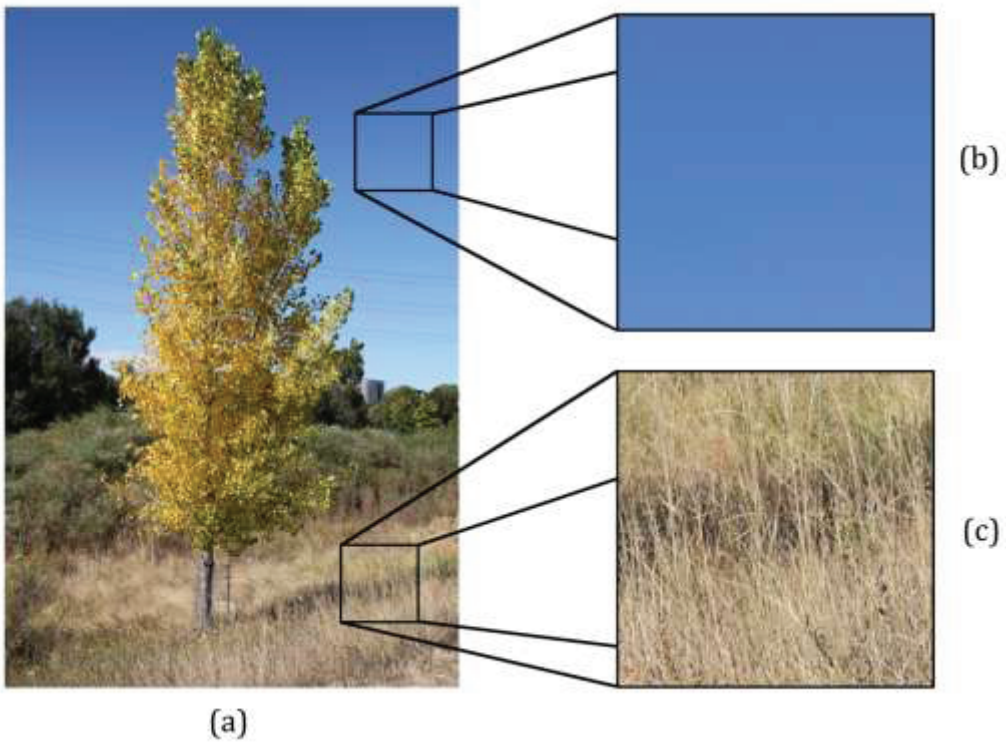


Figure 13 Image Frequency Examples
Low frequency information is an area of slow or uniform texture changes like that of a clear sky (b). High frequency information includes areas of rapid texture change, like those in the blades of grass (c).

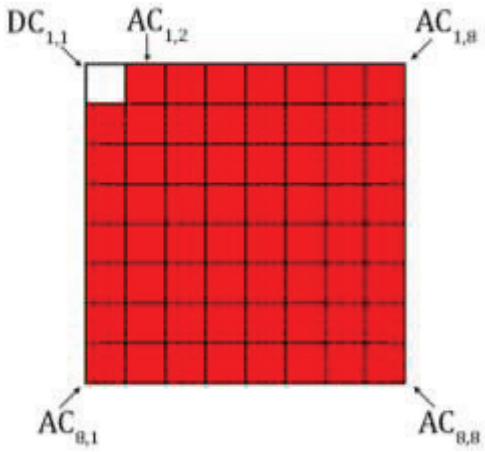


Figure 14 DC and AC Components of the DCT

Step 5, after output from the DCT, each coefficient value from the luminance and chrominance components are then divided using a 64-element quantization table. The quantization table used for the chrominance channels and luminance channel are different. In addition, all pixel blocks for each channel are processed with the same quantization table [Figure 12 (c)]. Lower values in the table indicate higher image quality, while higher values indicate lower image quality. Each element in the quantization table can be an integer value from 1 to 255. After quantization, the resultant values are then rounded to the nearest quantize step size, or nearest whole number [Figure 12 (d)]. This is the main cause of information loss in DCT-based encoders and is referred to as quantization error. This error is responsible for a small change in the value for each pixel between the original image and when it is compressed.

Step 6, after quantization, lossless entropy encoding is used to further compress the image by encoding the coefficients based on their statistical characteristics. As can be seen in Figure 12 (d), the quantization process produces many zero values. These are encoded as strings of zero-runs as opposed to the hex value of '0' repeated numerous times. These coefficients are ordered in a zigzag sequence by spatial frequency from low to high when encoded [Figure 15]. Step 7, a JPEG header is combined with the entropy encoded data stream to create a JPEG file.

The decoding process uses the same steps, but in reverse. The data is recovered from the entropy encoding [Figure 12 (d)]. The quantized DCT-coefficients are then multiplied by the same quantization table used for quantization to recover the de-quantized DCT coefficients [Figure 12 (e)]. Further errors are generated in this step due to rounding and truncation of the resulting frequency values. The de-quantized coefficient values are converted from the frequency domain into the spatial domain using the inverse DCT (iDCT) [Figure 12 (f)]. Finally, the image is converted from the YCbCr color space back to the RGB color space and the image is reconstructed. Errors of the quantization can be seen as slight variations in the pixel values between the reconstructed image samples, [Figure 12 (f)], and the original image, [Figure 12 (a)].

JPEG compression, on average, can effectively reduce file size to 1/10 of its original size with very little perceptual loss in image quality [Figure 5 (b)]. While size can be reduced even further, image degradation, in the form of JPEG artifacts will occur [Figure 5 (d)]. Such artifacts include blockiness, abrupt color transitions, and fringing around edge detail. JPEG compression is a lossy compression algorithm, meaning that information discarded in the compression process cannot be recovered.

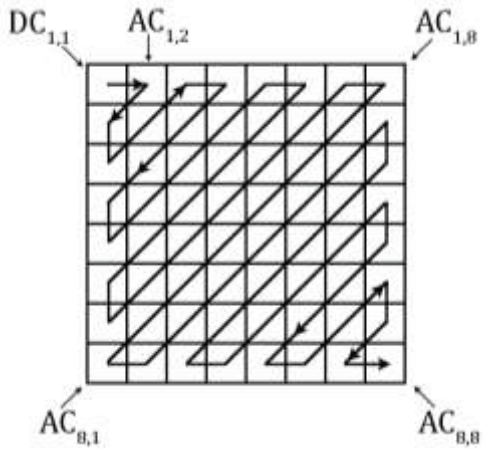


Figure 15 DCT Coefficient Entropy Encoding
Arrangement of the DCT coefficients for entropy encoding.

2.2.2 Interpolation Analysis

Interpolation is the process of approximating a value, or function, between two defined discrete points. When a digital image is rescaled, the process is either the addition of new pixels, or the removal of existing ones. Either way, new values for an image must be estimated from the old ones. There is a wide variety of interpolation algorithms used for images, however, discussing them is outside of the scope of this paper. Instead, the concept will be explained using bilinear interpolation. Bilinear interpolation works in two directions to achieve a best approximation for an unknown pixel value, based on the values of the surrounding pixels.

Figure 16 shows a simple 2-D lattice that has been up-sampled by a factor of 2. The known values of the original sample are separated by the newly created pixels [Figure 16 (b & e)]. Unknown pixel values are approximated using the known values of the original samples. In Figure 16 (c & f), the original values are located at the corner of the image so that $y_1=x_1$, $y_3=x_2$, $y_7=x_3$, and $y_9=x_4$. Using bilinear interpolation, the missing values on the edges are approximated from two known values such that:

$$\begin{aligned} y_2 &= 0.5y_1 + 0.5y_3 \\ y_4 &= 0.5y_1 + 0.5y_7 \\ y_6 &= 0.5y_3 + 0.5y_9 \\ y_8 &= 0.5y_7 + 0.5y_9 \end{aligned} \tag{1}$$

While, the center value is approximated from all four known values by:

$$y_5 = .25y_1 + .25y_3 + .25y_7 + .25y_9 \tag{2}$$

The process of interpolation, being a mathematical procedure, causes the random distribution of values in an original image to become ordered. For example, in Figure 16 note that the values of the pixels in the odd rows and even columns have the same linear combination of their horizontal neighbors, as defined by Eq. (1). Likewise, the pixel values of the even rows and odd columns will have the same linear combination of their vertical neighbors. This ordered relationship is identified by reoccurring patterns, or periodicity, found in the pixel values of the image matrix.

While Gallagher [36] uses second derivatives to identify periodicity in rescaled images, this method can be further applied to JPEG compression analysis to identify whether a JPEG image has been compressed once, or more than once. Traces of periodicity, in the form of repeating patterns in the second derivative, indicate that the image is most likely not a first generation image.

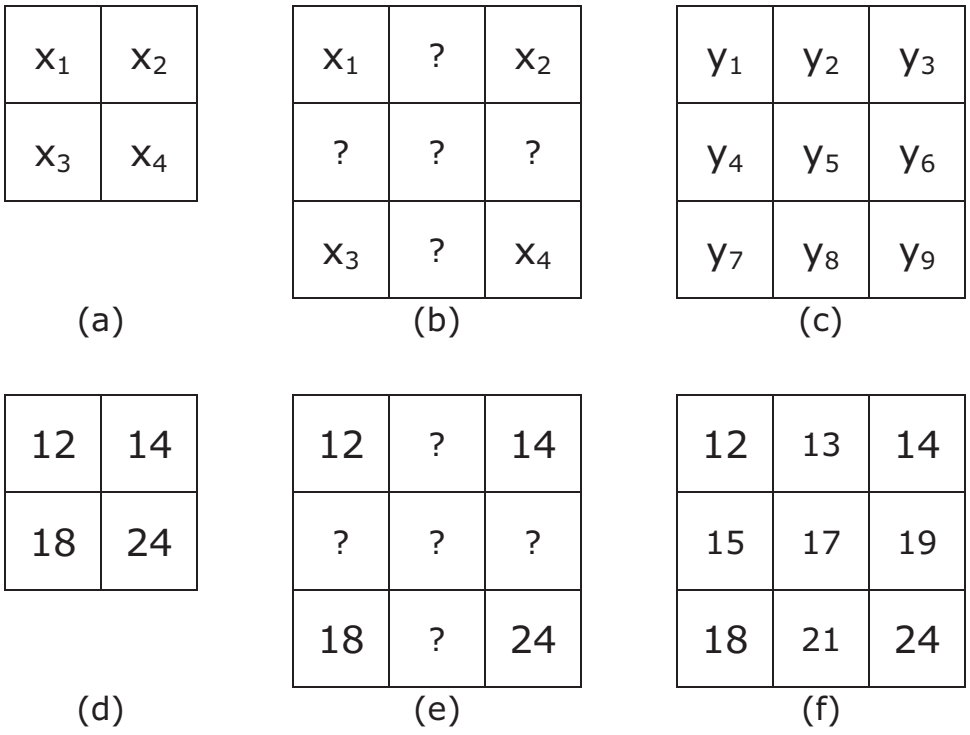


Figure 16 Bilinear Interpolation
Sample of a 2-D lattice taken from an image (a). Lattice up-sampled by a factor of 2 (b) and missing values computed using bilinear interpolation (c). (d-f) is an example using real numbers.

This analysis can detect traces of interpolation in images by computing a second derivative signal to identify signs of periodicity. The interpolation detection algorithm works by first computing the second derivative of each row of the image matrix [Figure 17]. Gallagher points out that columns may also be used because interpolation algorithms usually use values in both directions, row wise and column wise, to achieve the most accurate interpolated value. The absolute values are then averaged over all rows to calculate a mean and the Discrete Fourier Transform (DFT) is computed to identify frequency peaks of the second derivative signal.

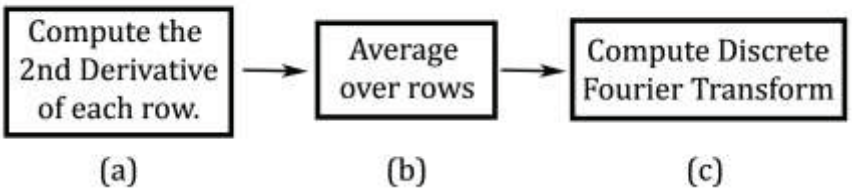


Figure 17 Block Diagram of the Interpolation Detection Algorithm

The algorithm proposed by Gallagher works well on uncompressed images that have been resaved as JPEG images. Looking at Figure 18, the interpolation artifacts of the camera are easily discernable as sharp peaks in the second derivative signal at the same points in the graphs [Figure 18 (a, c, e)]. Notice that the center spike is much larger than other two spikes. Furthermore, the signal is relatively smooth in the logarithmic scale. When the image has been recompressed, multiple sharps peaks appear and the strength of the center spike decreases [Figure 18 (b, d, f)]. In addition, the signal of the logarithmic scale has a jagged characteristic. This method tends to break down after an image has been recompressed twice, and even more if the image is recompressed with a higher quality compression.

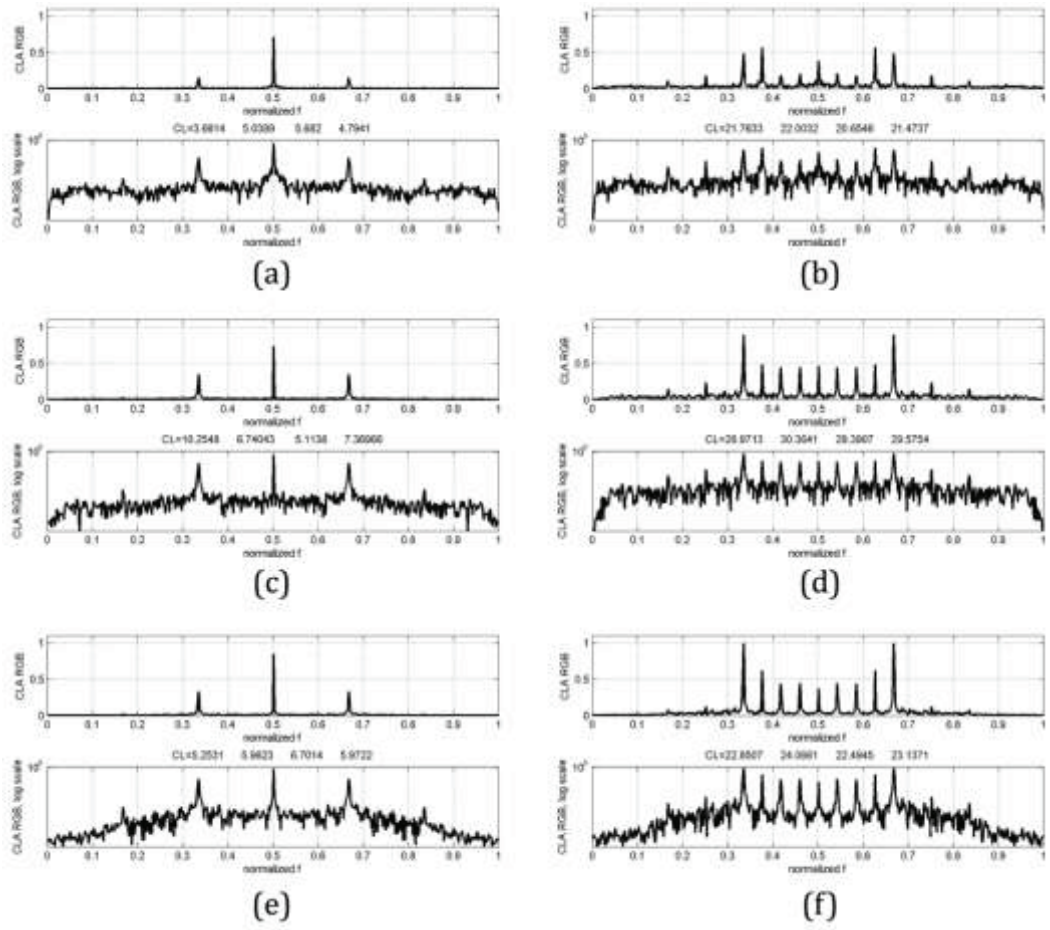


Figure 18 Interpolation Analysis 1st Generation Images
This figure shows graphs of interpolation artifacts using the algorithm described by Gallagher. Uncompressed images are on the left (a, c, e) and the same images recompressed by Photoshop (compression setting 5) are on the right (b, d, f).

Interpolation of pixels can also occur when an image is resized or rotated. Popescu and Farid propose a technique to identify interpolated areas by using the expectation/maximization algorithm [37]. Essentially the algorithm uses an iterative process to estimate an unknown parameter, in this particular case the interpolation method used to scale or rotate an image. The algorithm is used to determine if neighboring pixel values are correlated to each other by a set of

periodic samples. The results of the EM step produce a probability map that can be used to determine if an image has been resized or rotated. The Fourier transform is computed on the probability map and any patterns that emerge in the transform is an indication of interpolation. When uncompressed images were used, the false-positive rate was less than 1% for up sampling and rotations greater than 1 degree. The accuracy of the algorithm drops from 99% as down-sampling passes 20%. The algorithm also does well in the presence of gamma correction and in the presence of low signal to noise ratios. This technique also works well with JPEG compressed images with minor drawbacks. The JPEG block size interferes with the algorithm when down sampling is 20% or when the up-sampling rate is 60%. However, these artifacts do not affect the detection of rotation. This technique can also be used to determine if small composited portions within an image have been resized and rotated to get the piece to fit into the new image [37].

2.2.3 Color Filter Array

The pixels in almost all CMOS and CCD sensors are only sensitive to light intensity values and do not distinguish between the different wavelengths of the spectrum. Currently, the only exception is the Foveon X3 sensor, which has the ability to record color information at each pixel location [38]. To overcome this limitation of the CMOS and CCD sensors, an array of color filters is placed over the sensor to filter light by color. Therefore, each pixel only records light intensity information for a wavelength range of light dependant on the color. This filter is referred to as the Color Filter Array (CFA). While there are many types of CFAs, the most common type is the Bayer CFA, which consists of a multitude of 2 x 2 mosaic sub-blocks, each with two green, one red, and one blue filter [Figure 19]. Since the human eye is more sensitive to green light, the Bayer CFA filter usually contains more green filters than red and blue. The raw sensor information is a collection of green, red, and blue intensity values spread across a single matrix [Figure 20 (b)].

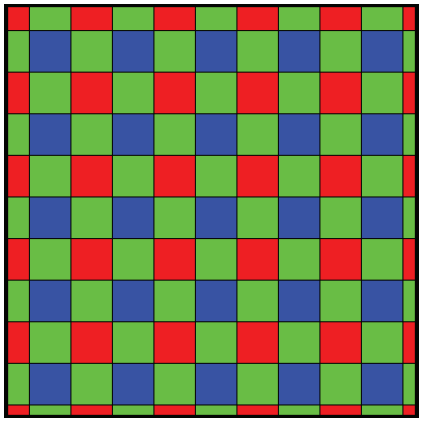


Figure 19 Bayer RGB Color Filter Array
Illustration of the Bayer CFA, which contains a mosaic of red, green, and blue filters mounted onto the sensor matrix. The CFA filters light by color so intensity values for only one color range are recorded by each pixel.

However, a color image consists of three color layers: red, green and blue. The raw data is separated by color onto its respective layer. As can be seen from Figure 20 (c), 50% of the green information is missing, while 75% of the red and blue information are missing. A process known as interpolation, or demosaicing approximates these missing values.

There are a number of different demosaicing algorithms used in the industry today and each handles the task differently. Demosaicing algorithms can be grouped into two classes. The first treats each layer as a separate image, interpolating missing values from those contained only in that layer. These simple algorithms include bilinear, nearest neighbor and bi-cubic interpolation. While these interpolation algorithms work well in smooth areas of an image, they tend to produce artifacts along edges and in sections of high detail. The second class of demosaicing algorithms utilizes the fact that all three color channels are highly correlated to each other. The mathematics used by these algorithms are considerably more complicated, and they produce better quality images near the edges and in areas of fine detail.

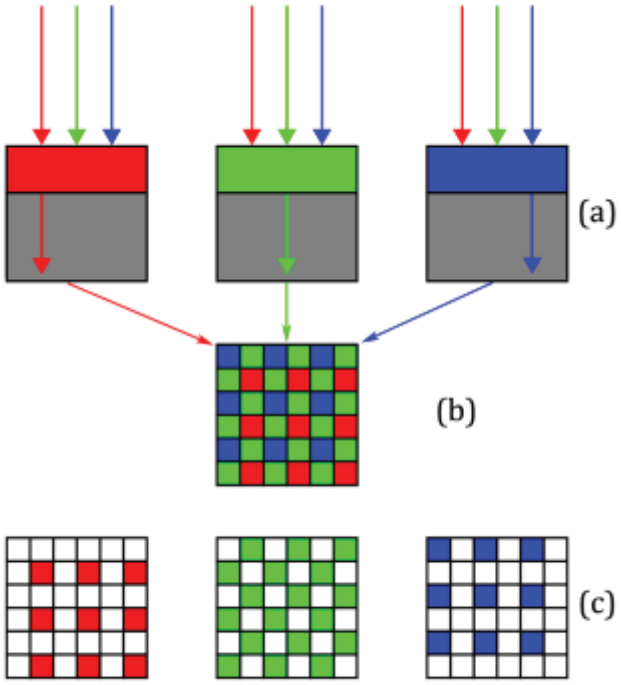


Figure 20 CFA to Three Channel Layer
Light entering the CFA is separated by color frequency (a). Intensity values for one color range are recorded by the sensor at each pixel location in a mosaic pattern (b). RGB color images require three layers, red, green and blue, to create a full color image (c). Note the missing values in each color layer that will need to be interpolated.

Whatever the process, the task of demosaicing is to determine the missing values for each color layer. Since these algorithms are used to estimate missing pixel values, this process introduces correlations between neighboring pixels. Correspondingly, because the CFA pattern is periodic, we can expect the correlations to be periodic as well. Interpolation algorithms vary from manufacturer to manufacturer and even between different models made by the same manufacturer. Because the demosaicing algorithms are different, we can exploit the variations of the color interpolation to determine if images were produced using a specific camera model, or if the image had been altered.

Analysis of the CFA has been useful in detecting the presence of tampering by using the expectation-maximization (EM) algorithm [39]. This technique presented in this paper produced 0% false positives, with close to 100% accuracy for the interpolation algorithms tested on both un-tampered images and images with non-linear point-wise gamma correction. However, the accuracy drops slightly when an image is processed with Gaussian noise, and drops considerably in the presence of increasing JPEG compression. In addition, there was vulnerability in the algorithm when it was tested against the JPEG2000 compression scheme. They explain that quantization of wavelet coefficients introduces artifacts that are indistinguishable from the artifacts caused by the CFA interpolation.

Some have even used the EM algorithm on individual color channels for source camera model identification [40, 41]. While only mildly successful, the test results are not too reliable since the test was performed using images from only 3 cameras. Because similar demosaicing algorithms are probably used in many camera models made by the same manufacturer, we can expect many to share similar characteristics.

2.2.4 Quantization Tables

This section focuses on the analysis of the quantization table (QT) for JPEG compressed images. The quantization table controls how much compression is applied to an image when being saved using JPEG compression. After an 8 x 8 pixel block is converted to the frequency domain via the discrete cosine transform (DCT), a quantization table is applied to the resultant 64 DCT coefficients. This step in the process is called quantization and it is where visually insignificant information is discarded to reduce the amount of information required to represent the image, i.e. reduce the digital file size. This step is one of the main sources of information loss in a JPEG image. The unimportant information that is discarded is determined based on the psycho-visual characteristics of the human eye, which is good at detecting slight differences in brightness over large areas, but not proficient at distinguishing slight color variations. The information that is removed is information of high frequency, or areas of high detail [Figure 13 (c)].

A quantization table is an 8 x 8 matrix that consists of 64 elements that are applied uniformly to the 64 DCT coefficients. Values within the quantization table can range anywhere from 1 to 255, and are considered the scaling factor [Figure 21]. A low value of 1 indicates very little compression is being applied to the image. This in turn retains larger, higher quality images. Whereas a high value of 100 indicates that, a large amount of information will be discarded, resulting in a small, low quality image. The process is simply accomplished by dividing each of the DCT coefficients by its corresponding quantizer step size. After this division, the remaining value is then rounded to the nearest whole integer [Figure 12 (d)]. Each JPEG image consists of 2 separate quantization tables, one used for the luminance channel, and one used for both chrominance channels.

1	1	1	1	2	3	4	5
1	1	1	2	2	5	5	4
1	1	1	2	3	5	6	4
1	1	2	2	4	7	6	5
1	2	3	4	5	9	8	6
2	3	4	5	6	8	9	7
4	5	6	7	8	10	10	8
6	7	8	8	9	8	8	8

(a)

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

(b)

Figure 21 Sample Quantization Tables

Lower values in the quantization table mean less compression and a higher quality image output (a). Higher values in the quantization table will produce a lower quality image (b).

JPEG compression can be achieved through in camera processing or by computer software. Some cameras utilize the standard quantization table published in the International JPEG Group standard [Figure 22]. This table can be scaled by a factor of Q using the following formulas:

$$S = \begin{cases} for\,(Q \le 50) & \dfrac{5000}{Q} \\ for\,(Q > 50) & 200 - 2Q \end{cases} \tag{3}$$

$$T_s[i] = \left\lfloor \frac{S * T_b[i] + 50}{100} \right\rfloor \tag{4}$$

where Q is the quality factor, S is the scaling factor, T_b is the base table and T_s is the scaled table. The quality factor, Q, can be between 0-100 and is used to create the scaling factor S, shown in Eq. (3). Each element *i* in the scaled table is computed from the base table by Eq. (4). A higher value of Q will result in a less compressed, higher quality image.

While the user has some choice as to the quality of compression, the manufacturer of the device or software predefines the exact values within the quantization table. In fact, quantization tables can be different between camera models of the same manufacturer. Hany Farid extracted the quantization tables from the images created by 204 digital cameras [42]. On average, each camera's quantization table matched 1.43 other cameras, noting that only 62 out of the 204 cameras had unique quantization tables. Furthermore, the quantization tables used by Adobe Photoshop do not match the quantization tables of the other cameras. A more vigorous testing was done in a followed up study, which included the different quantization tables for the different compression settings of each camera [43]. Out of 10,153 different cameras, it was found that 517 entries (5.1%) had a unique quantization table, 843 (8.3%) had at most two matches, and 1,056 (10.4%) had at most three matches to other cameras. When combined with resolution size, the number of unique quantization table/resolution combinations rose to 2,704 (26.6%) for a unique pairing, 3,777 (37.2%) for at most two matches, and 4,477 (44.1%) for at least three matches.

Quantization tables are classified into four different categories [44]. *Standard Tables* are defined as scaled versions of the quantization tables

published in the International JPEG Group standard [44]. Images with standard tables have two sets of quantization tables, one for luminance and one for the chrominance channels. The standard quantization tables have scaled values of Q between 1 and 99 [Figure 22]. The author notes that many cameras and software programs use these standard tables. *Extended Standard Tables* are similar to the standard tables with the exception that three quantization tables are specified within the image file. While a standard table shows the same quantization table used on both chrominance channels, the extended table actually shows a third table, which is a duplicate of the second.

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

6	4	4	6	10	16	20	24
5	5	6	8	10	23	24	22
6	5	6	10	16	23	28	22
6	7	9	12	20	35	32	25
7	9	15	22	27	44	41	31
10	14	22	26	32	42	45	37
20	26	31	35	41	48	48	40
29	37	38	39	45	40	41	40

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

7	7	10	19	40	40	40	40
7	8	10	26	40	40	40	40
10	10	22	40	40	40	40	40
19	26	40	40	40	40	40	40
40	40	40	40	40	40	40	40
40	40	40	40	40	40	40	40
40	40	40	40	40	40	40	40
40	40	40	40	40	40	40	40

(a) (b)

Figure 22 Standard Quantization Tables
A standard quantization table (a) and the standard quantization table scaled by a value of Q=80. The top tables are used on the luminance channel, while the bottom tables are used on both chrominance channels.

Custom Fixed Tables are pre-defined by the manufacturer as a proprietary table for their camera or software. These tables are built around a limited number of compression settings available to the user. For example, Adobe

Photoshop has 13 quality settings (0 - 12), each with a unique quantization table. Furthermore, the quantization tables for the 13 settings have remained the same from Photoshop 3, to the current version of CS5.1 [43].

The fourth category, *Custom Adaptive Tables*, does not conform to the JPEG standard and in fact, change between images of the same camera on the same setting. Kornblum did notice that while the quantization values changed, some values were consistent between images taken at the same camera setting [44]. These variable tables, implemented in newer cameras, change the quantization table based on scene content and resolution. The camera noted in the study was a Fuji Finepix A200.

While this analysis is not effective at distinguishing between JPEG images taken with different camera makes and models, it can be helpful in excluding devices from the pool of possible sources. Caution should be taken when dealing with cameras that use custom adaptive tables. Identifying all possible QTs associated with these cameras can be a difficult process. If the analyst is not careful, cameras could be inadvertently excluded from the pool of possible sources because all QTs were not accounted for. In addition to identifying camera models, looking at the quantization table may be useful in determining if the file had been saved using image-processing software, as is the case with Adobe Photoshop [43].

2.2.5 DCT Coefficient Analysis

Analysis of the DCT coefficients can yield indications of image tampering in the form of double JPEG compression. Typically, when an image is manipulated, it must first be loaded into a photo editing software. When the alterations have been made, the image is then resaved. In the case of a JPEG image, this process groups the DCT coefficient values into multiples of the quantization step size [45].

DCT coefficient analysis is only available for JPEG compressed images. The Discrete Cosine Transform converts the image from the spatial domain into the frequency domain. This conversion can be explained by thinking of the forward DCT as frequency analyzer and the inverse DCT as a frequency synthesizer. Each 8 x 8 block is essentially a 64-point discrete signal, which is a function of the two dimensions, width and height. The forward DCT decomposes this block into 64 orthogonal basis signals, each one corresponding to the spatial

frequencies of the inputs spectrum [Figure 23]. These signals are referred to as the DCT coefficients.

The DCT coefficients are separated into two types of signals, DC and AC components [Figure 14]. The *DC* coefficient refers to the mean value of the waveform and represents the average of the input samples. The DC component typically contains a significant portion of the total energy for each JPEG block. For each pixel block there is only one DC component. The remaining 63 coefficients are referred to as the *AC* coefficients.

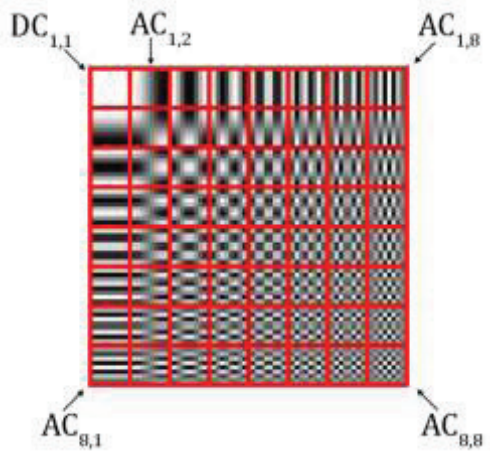


Figure 23 DCT Coefficient Ordering

The DC coefficient resides in matrix location (1,1). The remaining AC coefficients are ordered from lowest frequency (top left) to highest frequency (bottom right.) This figure represents the linear combinations of the input values.

Before being processed by the quantization table, the DCT coefficient values exhibit a Laplacian distribution, characterized by a curve with a pronounced sharp peak centered on a mean value [Figure 24]. However, this distribution is disrupted when the DCT coefficients are divided by the quantization table and the resulting values are rounded to the nearest integer [Figure 25]. The DCT histogram of each AC coefficient for a first generation JPEG image shows periodic spikes and valleys at multiples of the quantization step size. It is important to note that the coefficient values are periodic for a first generation JPEG image. However, the periodicity of a first generation JPEG image is still evenly distributed by the Laplacian model [Figure 25 (b)].

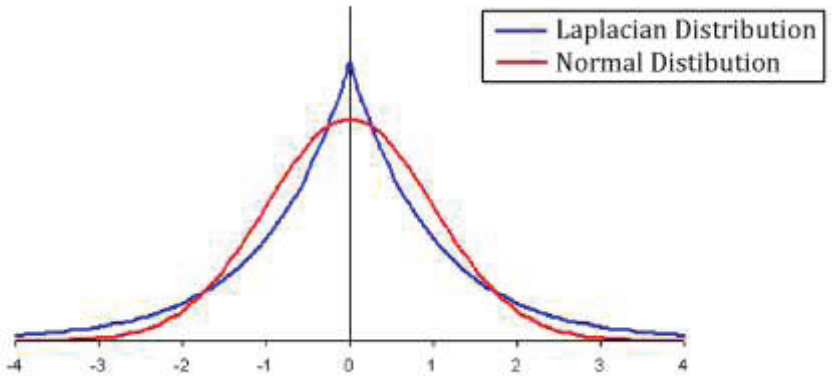


Figure 24 Laplacian Distribution
The Laplacian distribution, characterized by its sharp peak, is modeled in blue. The normalized, or Gaussian distribution, characterized by a smooth peak, is modeled in red. (Image courtesy of Vose Software [46].)

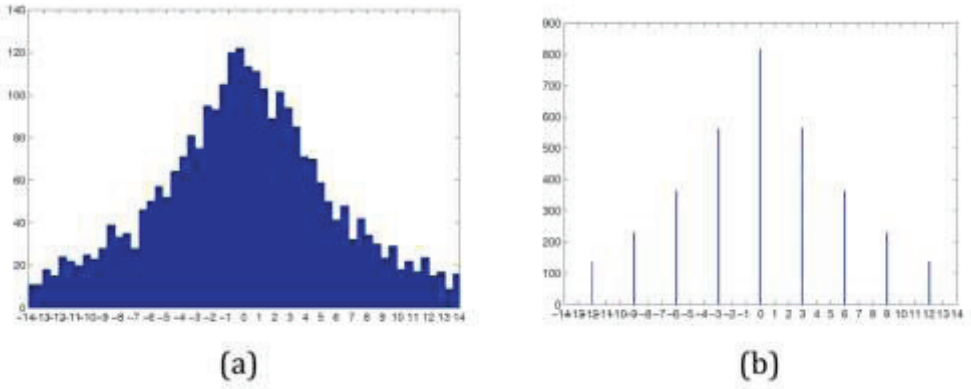


Figure 25 DCT Coefficient Distribution
Distribution of AC coefficients at (1,2) for all blocks before quantization (a). Distribution of AC coefficients at (1,2) after quantization with step size = 3 (b). (Figure originally appeared in [45], courtesy of Lou, W., et al.)

When a JPEG image is saved a second time with JPEG compression, this image is referred to as a 2nd generation image. When the JPEG compression process is repeated, the DCT coefficients undergo further transformation and

exhibit characteristics of double quantization, also known as the double quantization effect. With slight compression, the quantization step will have a smaller effect on an image. In contrast, increased compression will have a more noticeable effect on the final image. It should be noted that the quantization tables for the luminance channel and the chrominance channels are different.

There are three possible scenarios with second-generation JPEG images. The first is that the secondary quantization of the recompressed image, denoted as Q2, is smaller than the primary quantization Q1 of the original compressed image, or Q2 < Q1. The second, is the secondary quantization step size is more than the primary quantization quality, or Q2 > Q1. The third is that Q2 = Q1. The values of the quantization matrixes can have differing impacts on the distribution of the DCT coefficients.

If the secondary quantization step is less than the primary value, the histogram of the double compressed image will exhibit peaks with periodic missing values [Figure 26 (a)]. If the secondary quantization step is more than the primary value, the DCT histogram can exhibit a periodic pattern of peaks and valleys [Figure 26 (b)]. If the two compression settings are the same, then the DCT histogram will not exhibit such a clear indication because the values of the coefficients will not be redistributed based on a secondary quantization value.

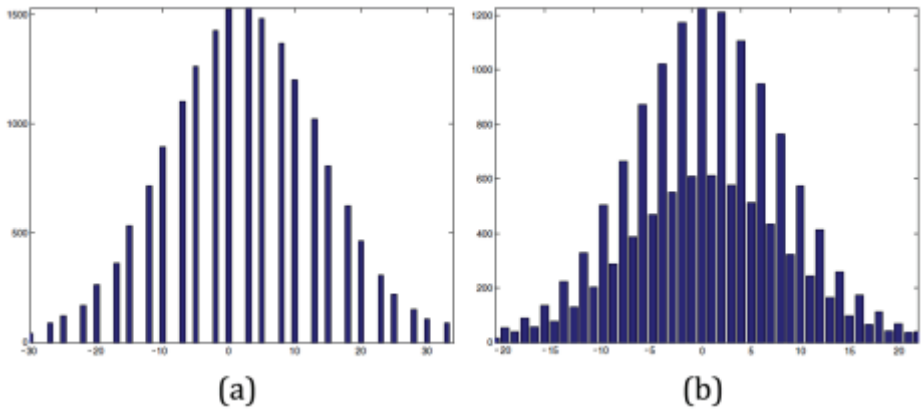


Figure 26 Double Quantization Effect
These two figures represent the histograms of a double quantized signal. The first panel is a signal quantized by a step size of 5, followed by 2 (a), and the second is a signal quantized by a step size of 2, followed by 3 (b). (Figure originally appeared in [47], courtesy of He, J., et al.)

While the previous examples represent simplistic views of the quantization and double-quantization effects, the reality is that the DCT coefficients in actual images are far more complex. Because quantization step sizes can be anywhere between 0 and 255, signs of double compression may not be so readily apparent [Figure 27]. The artifacts of double compression are apparent in the repeating pattern of 2 taller spikes, followed by a slightly smaller one, which is most noticeable on the left side of the histogram [Figure 27 (b)]. These artifacts are not present in an original image coming from the camera. The quantize step for the DC component was 9 for the original image, and 12 for the recompressed image.

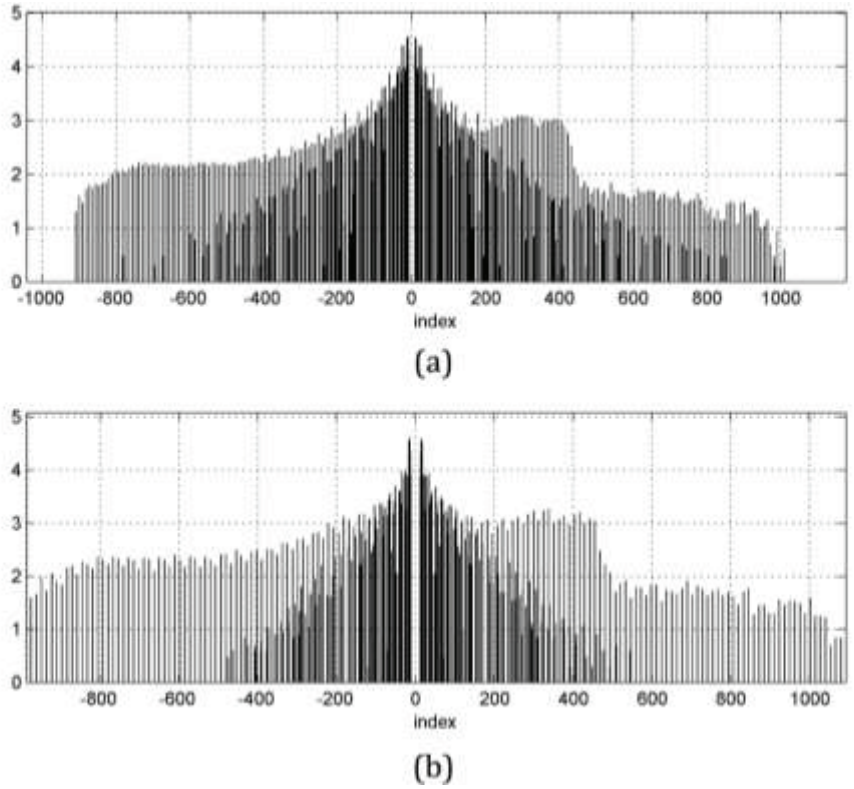


Figure 27 DCT Histogram of DC Component
This histogram represents the DC component of an original JPEG image (a) and a double compressed image saved with Photoshop with a compression of 5(b).

In addition to the histograms, the periodicity of the DCT coefficients can also be viewed by computing the Fourier transform on the DCT histograms [48]. Artifacts are noticeable as sharp peaks in the Fourier domain in the higher frequencies. In Figure 28 (a), a histogram of the AC coefficient values for 2,2 is shown for a double compressed JPEG image. The Laplacian distribution for the AC coefficient values is clearly disturbed. In addition, an FFT of the AC coefficient values histogram shows periodicity. In Figure 28 (b), two smaller peaks (marked with red arrows) separate the larger sharp peaks (circled in red).

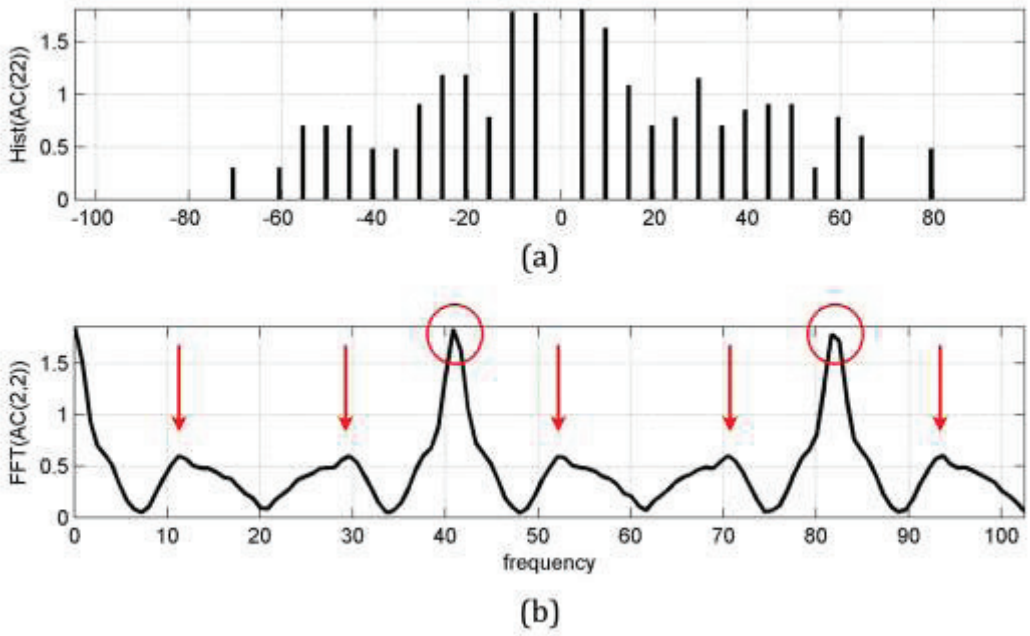


Figure 28 FFT of DCT Coefficients
The top panel (a) shows the AC (2,2) coefficient values for a double compressed image. The bottom panel (b) shows the FFT of the histogram. Periodicity in the histogram is show by peaks in the FFT.

Stamm et al. discovered a weakness in the DCT analysis when they manipulated the DCT coefficients of a JPEG image during the decompression of the image file [49]. A small amount of additive noise is added to the normalized DCT coefficients concealing the effects of quantization error. The result is that the distribution of DCT coefficients in the manipulated image resembles those of an uncompressed image. Too counter this anti-forensic attack, investigation of the high frequency sub-bands of an image can determine if the attack referenced above was used on the image [50].

2.3 Local Image Structure Analyses

While analysis of the global image structure can help determine whether an image has been altered on a grand scale, it does not help determine where the image has been altered. While indications may exist that an image may not be a first generation image, global image analysis cannot be used to determine if malicious alteration of scene content occurred. For example, pictures sent to family and friends may be resized and compressed for easy distribution over the Internet. These types of alterations are not considered a malicious attack on scene content.

Figure 29 Non-Malicious Alteration
An example of a non-malicious alteration by making a *levels* adjustment using Adobe Photoshop. The adjustment helps the lettering on the object to become more legible.

The analyses described in this section focuses on determining what in the image has been altered. While simply adjusting the brightness and contrast of an image to improve appearance is changing pixel values of the original image, it does not inherently alter the perception of events in the image. Consider an image taken in a low light situation. The image is considered underexposed because much of the scene detail is lost in the darker parts of the image. To correct this, a simple brightness adjustment is made to raise the overall exposure level, which in turn reveals details previously hidden. This type of processing does not alter the content of the image [Figure 29].

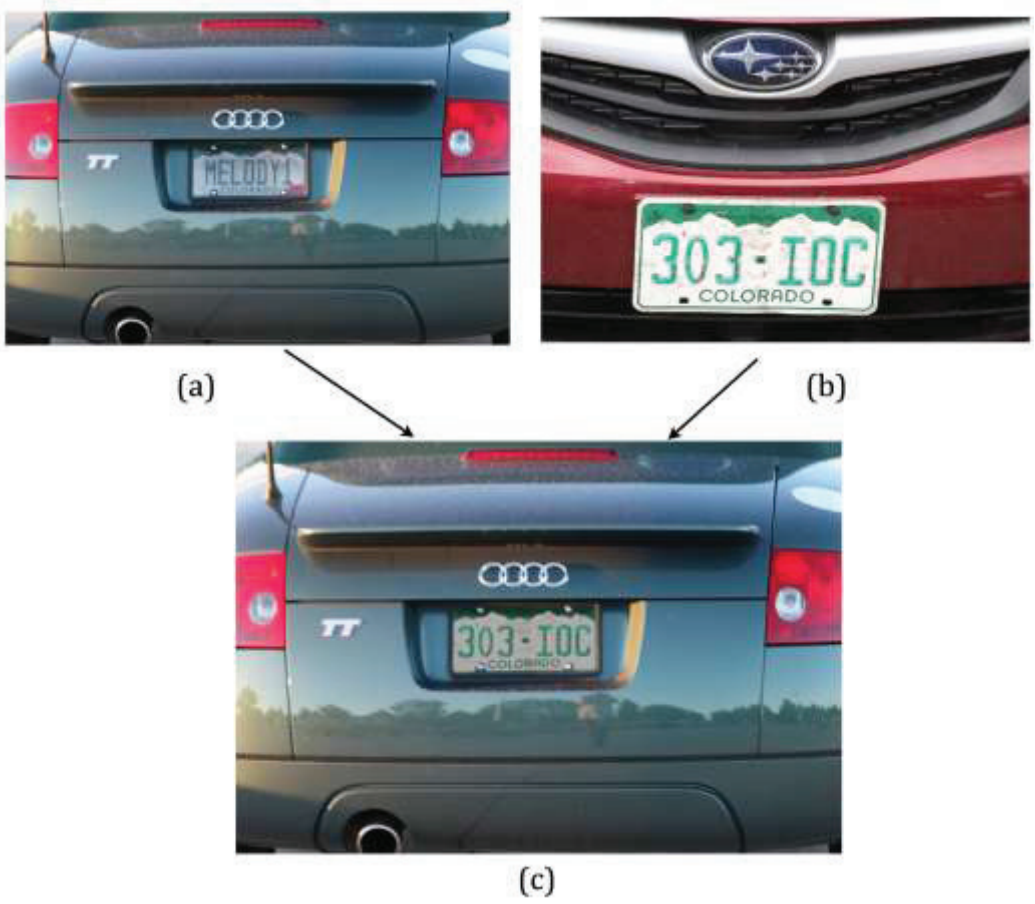


Figure 30 Malicious Alteration
Maliciously altered image (c), created by compositing two images from different years and locations (a, b).

Malicious manipulations can be defined as the application of techniques in an attempt to create an illusion, or deception, of the events in an image [Figure 30]. For example, in 2004 an image was widely circulated of presidential candidate John Kerry sharing the stage with anti-war activist Jane Fonda [19]. Although the two never shared a stage, the image was meant to draw attention to his anti-war activities during the Vietnam War and combine them with the charged nature of Ms. Fonda's actions during this time. This image is considered by some to have damaged his presidential hopes even after it had been shown to be a composite of two separate images.

The analyses presented in the following section help identify local areas of alteration on the pixel level, which is helpful in identifying where in the image content has been changed or altered.

2.3.1 Copy and Paste Detection

Identification of malicious alteration in image content is a large part of image authentication. One of the most common techniques is the use of copy and paste, which takes information from within an image, or separate image, and copies it over existing content. This type of technique can be used in two ways. The first is to replace content that existed in the scene at the time the image was taken. The second is to add content into a scene that was not present in the original image. Thus creating a relationship that did not exist at the time the picture was taken. These types of techniques can add or remove image content and leave no visually apparent traces of alteration.

These types of alterations are easily accomplished using most image processing software programs. Cloning is a specific technique that alters image content by using "with-in image" information to cover up other areas. For example, if a gun lay in a grassy field, the manipulator could remove the gun by using grass from other areas of the image and placing it over the gun. Depending on the skill level of the user, these types of alterations can be very hard to detect. The size of the manipulated area depends on the size of the object the manipulator is trying to hide. If the area were large enough, a visual inspection of the image would reveal two objects, or patterns, that are repeated in two different parts of an image. The problem however, is that if the area is small enough, or from another image, the forgery may not be detectable by the human eye.

If the image includes "within-image" alteration, the detection of cloning is a relatively simple process. The algorithm looks for exact pixel matches in a cluster of pixels to determine if two parts of an image are the same. This approach works on the assumption that the copy and paste techniques will use large contiguous areas of an image to alter content, rather than a multitude of individual pixels. This exhaustive approach compares a predefined pixel cluster, to every pixel block of the same size to determine a match. However, this type of analysis is very computationally intensive, and makes it impractical in all but small images. In addition, the results are dependent on the amount of retouching done to the cloned regions and the amount of compression applied to the image after alteration. To overcome this, the use of a robust matching technique that utilizes lexicographically sorted DCT coefficients, and sorting them by similar spatial offsets was developed [51]. A similar approach was used by applying the principle component analysis onto each image block [52]. These techniques were found to be slightly more robust when retouching was used to image content in the form of additive noise and lossy compression.

In addition, lateral color aberrations can be used to determine if portions of an image were altered [53]. Almost all optical lenses contribute to color aberrations due to failures in the optics to uniformly focus the different wavelengths of light onto the sensor. The degree of spatial shifting increases the further away the light is from the optical center of the lens. When an image is altered these aberrations fail to be consistent across the image matrix. Color aberrations are apparent at the edges of objects or in high contrast areas and are visible as green or magenta halos that radiate outward from the optical center. When a copy-paste alteration occurs, the direction of the aberration may be inconsistent with the surrounding material. While JPEG compression did affect the accuracy of this technique, detection of color aberration was still found to be a useful tool in detecting the tampered region.

2.3.2 PRNU Comparison

Malicious alteration can also be determined by looking at the photo response non-uniformity (PRNU) of an image (explained in chapter 4.2.1). When an image has been geometrically altered by a copy paste technique, or is a composite of two different images, the PRNU is altered as well. By comparing the PRNU of the suspect image to that of a reference image, inconsistencies in the PRNU can be an indication that a change has been made.

Mo Chen et. al., describe a process in which the PRNU of an imaging device is compared block by block to the PRNU of a suspect photo [54]. The basic principle is that any altered regions in the suspect image will not contain the same PRNU signature as an image from the same camera. This method assumes that the examiner has access to the suspect camera, or has access to un-manipulated images from that camera. A PRNU template can be estimated from the suspect camera by taking multiple, out-of-focus images of an evenly lit scene, like a clear sky. A strong PRNU template can be created from 8 images produced in this manner [55]. If the suspect camera is not accessible, then it is recommended that 50 images be used to produce a strong PRNU signature [56]. In [54] a wavelet-based filter was used to mitigate scene content and the resultant images were averaged to remove any remaining scene content and random noise. Similar steps were performed on the suspect image.

The images were compared using the normalized correlation coefficient algorithm with a sliding window of block size 128 x 128. Any correlation values that exceeded a pre-defined threshold were flagged as deviating from the reference PRNU pattern. The test was run on 345 forged images and saved as JPEG compressed images at quality (Q) equal to 90 and 75. For JPEG quality 90 images, the test correctly identified 2/3 of the forged regions in 85% of the forgeries, while incorrectly identifying 20% of the pixels as forged, in relation to the size of the forged areas, in 23% of the manipulated images. For the JPEG quality 70 images, the test correctly identified as least 2/3 of the forged region in 73% of the forgeries, while incorrectly identifying 20% of the pixels as forged in 21% of the manipulated images.

They note that the falsely identified pixels were generally located around the area of the forgery and were probably marked as such due to the 128 x 128 block size of the sliding window. Pixels were also falsely identified in the test photos at regions of high frequency information. It was also determined that missed identification of forged areas was due to large dark regions. PRNU is dependent on the amount of light hitting a sensor and is naturally suppressed in such regions.

2.3.3 JPEG Error Analysis

Alterations in JPEG images can be identified using the quantization and rounding errors inherent in the JPEG compression process. As described in section 2.2.1, an image is converted from the spatial domain into the frequency

domain using the Discrete Cosine Transform (DCT). A quantization table then quantizes the resulting DCT frequency coefficients. The main loss of information is due to the quantization error, or rounding of decimal values to the nearest integer. JPEG error level analysis focuses on the quantization and rounding error caused by the quantization and de-quantization process in the DCT coefficients of a JPEG compressed file.

In the majority of natural images, AC coefficient distribution can be modeled by a Laplacian curve [57]. After conversion into the frequency domain, AC coefficients cluster around a single mean value. But unlike a Gaussian, or normalized distribution, which has a smooth "bell" shaped maxima, the AC coefficients exhibit a pronounced sharp peak at the mean value [Figure 24]. When these values are quantized using the quantization table, the AC coefficients cease to be smoothly distributed and become grouped into multiples of the quantization step size 'q'. When the compressed image is again converted into the frequency domain, the DCT coefficients are no longer evenly distributed around a single value, but are spread with a Laplacian distribution around the multiples of the original quantization step size [Figure 25 (b)].

Weiqi Luo et al. proposed a technique that can detect traces of JPEG compression in an uncompressed image [58]. Their proposed method works by converting a suspect image into the frequency domain and analyzing the resulting AC coefficients. If the image had been previously compressed as a JPEG image, the distribution of the AC coefficients will exhibit periodicity as explained previously. The proposed technique claims accuracy of 98.6% on 256 x 256 pixel blocks, down to 95.08% on 8 x 8 pixel blocks. While the technique presented in this paper was only applied to the global image, this method could possibly be used to identify if any part of an uncompressed image came from a previously JPEG compressed photo.

Hany Farid proposed a method to identify manipulated areas of an image by detecting the presence of 'JPEG ghosts' [59]. Consider an image compressed by quality factor q1 to create image c1. If the image is resaved at quality factor q2, image c2 will be created. If we subtract the values of the DCT coefficients of c1 from c2 we will have the difference between the two compressed images. If the sum of the squared differences between the DCT coefficients of c1 and c2 are graphed onto a chart, the differences will increase as the compression quality q2 increases [Figure 31 (a)]. It is worth noting that the graph will reach a minimal difference when q1 = q2, indicating the original compression setting of c1. Note in Figure 31 (a), the minimal difference reached at the original compression quality = 17.

In another scenario, an image is compressed by quality factor q1, and subsequently compressed by quality factor q2 < q1, creating image c2. If c2 is compressed again with compression quality q3, c3 will be created. Again, graphing the sum of the squared differences of the DCT coefficients for an increasing q3 will show a minimal difference at quality level q2. In addition, a second minimum will be revealed indicating the level of the first quantization level q1 [Figure 31 (b)]. This second minimum is what is referred to as a 'JPEG ghost'. Because numerous minima are expected when quantization tables share integer multiple values, comparing the differences directly from the image's pixel values, as opposed to the DCT coefficients directly can mitigate this.

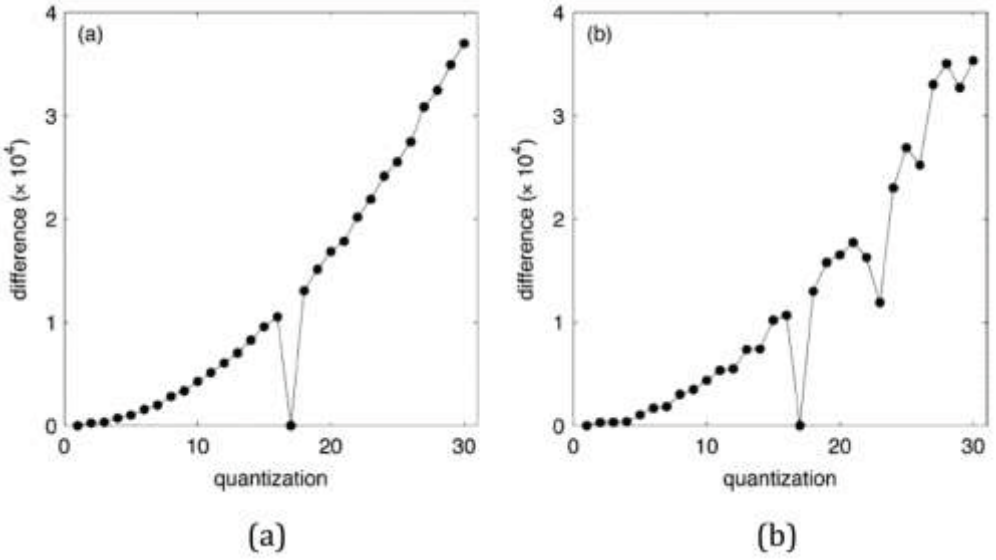


Figure 31 Graph Showing the Sum of the Squared Difference
Sum of the squared difference of a JPEG image originally compressed by quality factor 17 followed by recompression at increasing quality factors from 1 to 30 (a). Figure (b) shows the squared differences of an original image compressed at quality 23, then recompressed at quality 17, then recompressed at increasing quality from 1 to 30, and subtracting each from the quality 17 image. Note primary minimum at quality = 17, but a secondary minimum at 23 exists, indicating the original compression quality. (Figure originally appeared in [59], courtesy of Farid, H.)

Forged areas are simply discovered by taking a suspected image, recompressing it at sequentially different JPEG quality settings and subtracting each recompressed image from the original suspect image. Any areas that have been previously altered will result in a JPEG ghost appearing in the image around a specific JPEG quality value [Figure 32]. JPEG ghosting is highly prominent and easily visible in most instances.

Figure 32 JPEG Ghosting
An example of JPEG ghosting made by recompressing a suspect image (a) with consecutively higher compression levels, subtracting each from the original, and squaring the difference. The manipulated area is easily seen in (b) and is most prominent when the suspect image was recompressed with Q=75 and subtracted from the original.

The proposed technique, however, is only useful if the manipulated region was taken from an image compressed at a lower quality factor than the image in question. Furthermore, any misalignment in the 8 x 8 JPEG lattice structure will prevent the JPEG ghost from appearing. This problem can be overcome by shifting the image horizontally and vertically onto each of the 64 possible alignments before recompressing the image at the different quality settings.

In addition to the above technique, errors in the DCT can also be determined by mapping certain components of the DCT coefficients. Indications of alteration can be determined from the DC components, an average of the AC

components, or values of a specific AC component for each JPEG block [70]. Referred to as DCT mapping, the results using this technique are also highly prominent [Figure 33]. While the limitations are not yet explored, they are suspected to be the same as presented by Farid in [59].

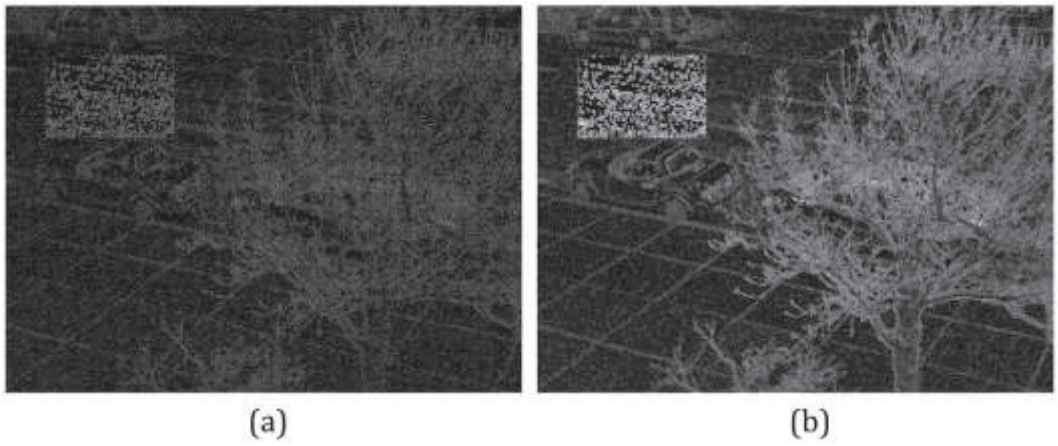


Figure 33 DCT Map
DCT map of the same manipulated image used in Figure 32. These images were made by mapping the DC components (a) and an average of all AC components (b) for each 8 x 8 pixel block.

2.4 Source Image Identification

Source image identification is the process of determining the origin of an image for identifying the device that created the picture. This process is valuable in determining if two sets of images share similar characteristics. Identification is accomplished by uncovering characteristics unique to an individual camera, scanner or other device, which are present in the image. The most robust of these techniques utilize imperfections that are introduced by a digital camera's sensor. This section discusses the source of the imperfections and how they can be used to provide strong evidence to determine if an image, or two sets of images, came from the same source.

2.4.1 Sensor Imperfections and Noise

The camera sensor introduces defects that are not a part of the real-world scene. The inherent limitations in the design and manufacturing of the sensor mean these defects will be introduced as imperfections and noise into the image output. Imperfections are artifacts that remain constant from image to image, while noise is considered a random artifact, much like the static on a television set. Sensor noise will not survive frame averaging, while sensor imperfections will.

Image acquisition for any given imaging device is complex and varies depending on the equipment and manufacturer. However, there are similar types of noises that are inherent in each device, both random and systematic. Shot and quantization noises are erratic and do not have consistent or predictable patterns. Shot noise is a result of the non-continuous flow of electrical current and is the sum of discrete pulses in time for each pixel. This means that the longer a sensor is active, i.e. longer shutter speeds, or the more sensitive the sensor is, i.e. low light conditions, a higher number of random electron noise will be recorded by the image sensor and recorded along with the scene. This type of noise is temperature dependent, meaning that higher temperature conditions will cause higher electron movement in the circuitry than lower temperatures. Quantization noise is caused by the process of converting light from an infinite amount of intensity values into a digital medium that has a finite amount of intensity levels. While this process introduces small distortions into the image, finer detail with larger bit depth can minimize this error.

Pattern noise is a systematic distortion inherent in the operation of a particular electronic sensor. The pattern of this noise is consistent from one image to the next and consists of dark current noise and photo-response non-uniformity (PRNU). Dark current noise refers to the charge generated in each pixel on its own by the electronic components associated to each individual pixel. While dark current is a fixed pattern noise, the intensity of the dark current is dependent on temperature. It can only be extracted from an image when the sensor is not exposed to light.

PRNU is the dominant part of the pattern noise and is caused by imperfections in the manufacturing process, sensor components, as well as the non-homogeneity of the silicon wafer that is used in the sensor. These slight imperfections affect a pixels ability to convert photons to electrons, causing minor variations to exist between pixels, so that some pixels are more sensitive

to light while others are less sensitive. This imprints a fingerprint, so to speak, onto each image produced by the sensor. PRNU is light dependant and the strength of the fingerprint amplifies as the intensity of light hitting the sensor increases.

Another characteristic of the sensor that can be used for identification purposes is dead or defective pixels that exist in the sensor matrix. These improperly functioning pixels can be mapped and appear at a consistent address in the sensor matrix. Since there may be millions of pixels in some sensor chips, the probability of two sensors having exact matches is highly improbable. Finding these imperfections, however, is dependent on scene content and temperature.

The following sections discuss the analyses of PRNU and defective pixels for source image identification. These analyses can also be used for source matching to identify if two groups of images where created by the same camera or sensor.

2.4.2 Photo Response Non-Uniformity

Characteristics exhibited by the Photo-Response Non-Uniformity (PRNU) make this component of the digital image a unique and helpful tool in identifying the "fingerprint" of digital sensors [56][60-65]. The PRNU is inherent in all imaging sensors, which makes it a universal identifier. PRNU contains a large amount of information that makes it a unique component specific to an individual sensor. The signal is present in all images and is independent of camera settings, scene content, and camera optics. It has been shown to remain stable over time and under a wide range of environmental conditions [64]. It has also been shown to survive JPEG lossy compression and image processing alterations such as brightness, color and gamma adjustments, to a certain extent [60].

While PRNU is an individual component of all digital image sensors, the ability to extract the signal is affected by the quality of the sensor, the amount of light interacting with the sensor, and scene content. Camera sensors of inferior quality, such as those in cell phones and lower-priced cameras, are more susceptible to the defects of the sensor components, than higher end model cameras. In addition, the strength of the PRNU signal is relative to the amount of light from the real world scene. Therefore, PRNU will be hard to extract from low light conditions and will be absent in completely dark images, such as when

the lens cap is on. Another hurdle in extracting the PRNU component is scene content. Areas of high frequency information (i.e. detailed areas of an image) will mask the PRNU, while areas of low frequency content, like those in a uniformly lit sky, will make the PRNU easier to extract [Figure 13].

The PRNU pattern is a relatively weak signal when compared to the strength of the scene content. Thus, depending on the scene, it can be extremely difficult to extract the PRNU from a single image. To properly remove the scene content and unwanted random noise, it is necessary to average multiple images together. With strong scene content it was found that 50 images averaged together was sufficient to extract a strong PRNU estimation [56]. If the scene is evenly illuminated, like those of a clear sky, the number drops considerably to about eight [55]. However, further study is being conducted to decrease the number of images required when scene content is present in suspect images [54][55][60]. The resultant averaged images should have a smooth and homogeneous texture without major identifiable features and splotches. This helps reduce bias in the following methodology and creates a more robust procedure.

Figure 34 contains examples created by averaging multiple images from the same camera. A wide variety of images having different compositions and taken in different lighting environments are critical for proper PRNU extraction. Since the fingerprint is a relatively weak signal, extraction is best accomplished by removing as much scene content as possible. Using too few images for averaging will leave scene content highly prominent, resulting in a poor PRNU fingerprint [Figure 34 (a)]. While using more images will mitigate scene content, images taken in similar lighting conditions and scene composition will result in average images with severe color splotches and boundaries, which will also affect the quality of the PRNU fingerprint [Figure 34 (b)]. Excellent images for PRNU extraction will be those that represent a wide range of lighting conditions or evenly illuminated scenes. Out of focus images or images of a clear sky will help facilitate a proper PRNU extraction [Figure 34 (c)].

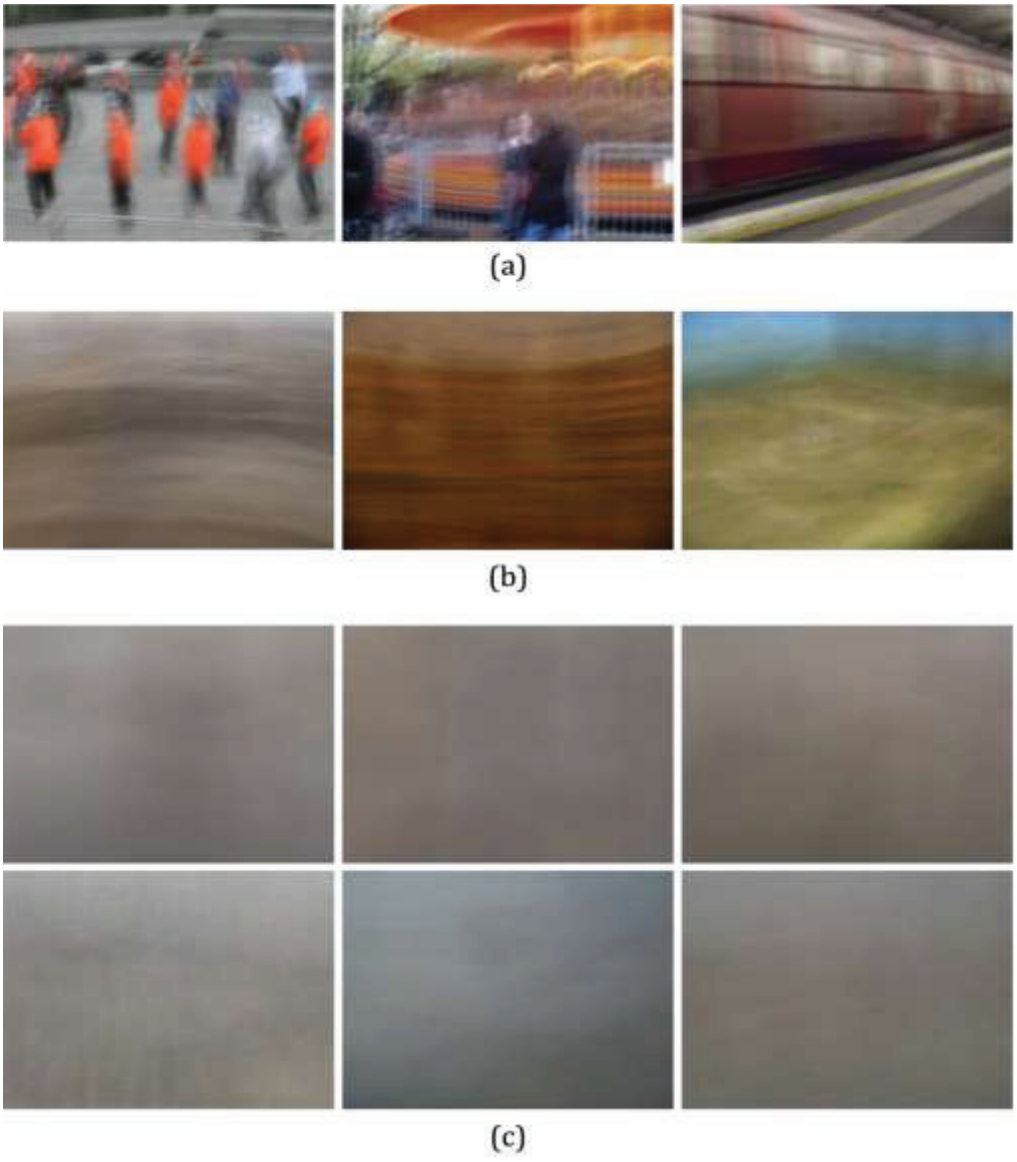


Figure 34 Averaged Frames for PRNU Extraction

PRNU is a relatively weak signal and is heavily influenced by scene content. PRNU extracted from (a) will produce poor PRNU signatures. While better, the pictures in (b) will still affect the PRNU extraction. Homogeneous images like those in (c) produce the best PRNU signatures.

There are two main methods proposed to mitigate scene content after averaging. The first uses a fourth-level wavelet decomposition to suppress the remaining image content [56][61-64]. This technique, however, is computationally intensive and takes a considerable amount of time to perform. The second method, which is computationally simpler, is to apply a simple Gaussian blur to a the averaged image [60][65]. While the wavelet process is technically more accurate, the results from testing have shown the differences to be small. After the scene content has been suppressed by either method above, the processed image is then subtracted from the original image. The difference between the two images is the PRNU signature.

The noise removal and subsequent averaging is used for a single grayscale image. Color images consist of 3 color layers: red, green and blue. Each layer should be processed separately producing 3 PRNU estimations. Since the color layers are highly correlated due to demosaicing, they should be combined into one grayscale image (*K*) using the following formula [55]:

$$K = .3Kr + .6Kg + .1Kb \tag{5}$$

where *Kr*, *Kg* and *Kb* are the PRNU estimates obtained from each color layer: red green and blue respectively.

Once the averaged PRNU image is acquired, it is necessary to compare this image to a database of PRNU images from multiple cameras. For the PRNU database to be as comprehensive as possible, it should contain images from as many different cameras as possible taken at different resolutions and quality settings.

The images are compared for similarities using the standard formula for correlation coefficient modified for:

$$corr(X,Y) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}(X * Y)}{\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}(X * X) * \sum_{i=1}^{n}\sum_{j=1}^{m}(Y * Y)}} \tag{6}$$

where *X* and *Y* are the PRNU estimation images, and *m* and *n* are the height and width of the image resolution respectively. This equation can only be used on PRNU signatures of the same resolution.

If the purpose of the analysis is for device linking or print matching, i.e. connecting two sets of images to the same source, both sets of images should be

prepared in the same manner. The two sets of images are compared to each other and the PRNU image database. For identifying whether a set of images came from a source camera, it is necessary to have access to this camera. It has been determined that out of focus images of an evenly lit scene can reduce the number of images required for proper PRNU extraction down to as little as eight [55]. Once the suspect camera PRNU frame is extracted, the frame is then compared to all images in the database, including the suspect images using the correlation formula. The author conducted a small-scale identification test using PRNU and the results are explained in Appendix B.

A large-scale PRNU test using over a million images resulted in a false rejection rate to be less than .0238 with a false acceptance rate below 2.4 x 10-5 [63]. Meaning that out of 1,000,000 images, 23,800 were excluded as not being from the camera they were actually taken from, and only 24 images were falsely identified as coming from the wrong camera. This signature has also been shown to be very robust against light forms of image processing such as gamma correction, brightness adjustments, re-sampling and JPEG compression [55][56][64][66]. While the robustness of proper PRNU identification is expected to decrease as stronger image processing is applied (i.e. heavy JPEG compression, heavy re-sampling), image identification was still very accurate. A study noted that even though the correlation decreased on heavily processed images, the correlation to PRNU reference patterns of differing cameras also decreased [56].

2.4.3 Defective Pixels

An additional tool for image source identification is defective pixels. Due to the large number of pixels in an image sensor, the likelihood that different cameras share defective pixels at the exact same location is very low, and only decreases as the number of defective pixels increases. There are five different kinds of pixel defects [67]. One of the most identifiable defects is the *hot point* defect, which is defined as a pixel that has a very high output voltage. These pixel defects result in bright spots in the output image [Figure 35]. Due to the CFA, these can show up as bright areas of red, green, blue, or white spots. The other identifiable defect is the *dead pixel*, which is defined as a pixel that no longer functions or has poor sensitivity to light. This type of defect is seen as a black spot in the image. A *point defect* is a pixel that deviates more than 6% when the sensor is illuminated to 70%. A *pixel trap* results in a partial or

completely bad column. Finally, a *cluster defect* is defined as a cluster of point defects.

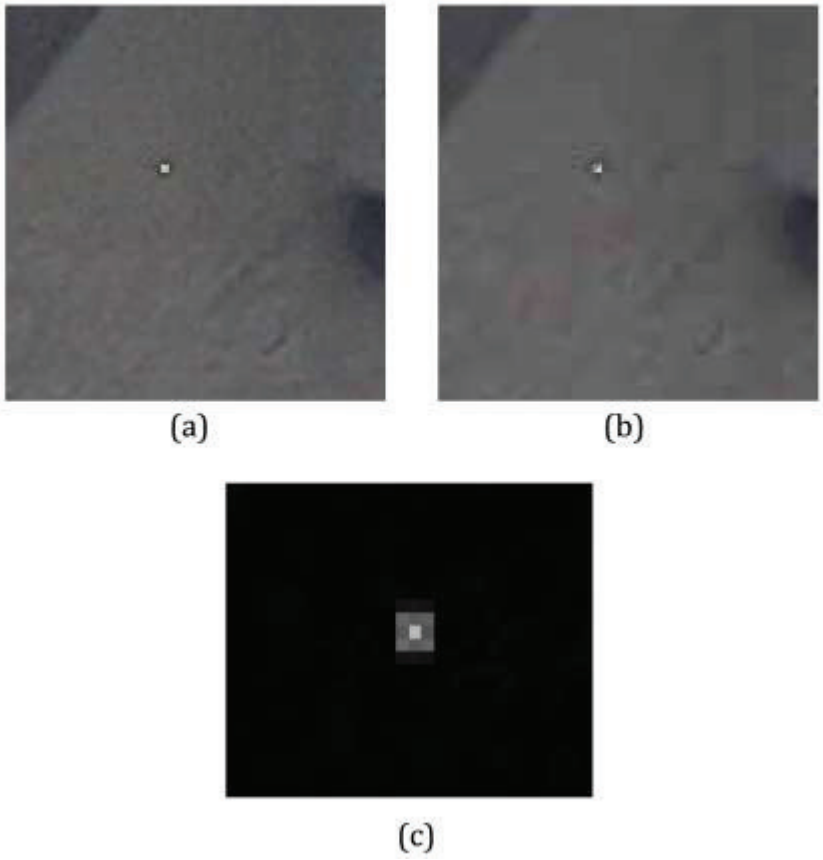


Figure 35 Hot Pixel

The hot pixel produces a high output regardless of scene content, however it may look different depending on how the camera processes the information. A hot pixel from an uncompressed image (a), JPEG image (b), and after color filter interpolation from in-camera processing(c).

Geradts et al., explain that three conditions must be met to most accurately identify if two images came from the same source [67]. First, the error due to defective pixels is random and is not caused by defects in the manufacturing process. Second, the temperature of the sensor at the time the images were taken was the comparable. Third, the defective pixel is independent of the other pixels. A fourth, not presented in [67], is that the operating conditions of the camera (i.e. settings, ISO speed) are comparable to the settings of the suspect image (Vorder Bruegge, R. W., personal communication, November 7, 2011). Vorder Bruegge observed that pixel defects, specifically 'hot' pixels, became more apparent with longer exposure times. While the exact cause is unknown, it has been attributed to charge buildup in the sensor at the pixel location. Since digital camera sensors can contain millions of pixels, the likelihood that two cameras share defective pixels at the same address start to decrease with a higher number of identifiable defects. As images start to have more identifiable pixel defects, the stronger the proof becomes that two images came from the same source.

While this method seems like an ideal identifier for source image identification, it does have some limits. The number of defects is dependent on the quality of the sensor. Cheaper sensors tend to have a larger number of bad and inferior components, which can wear out more quickly than the sensors in more expensive cameras. However, modern camera models may have internal processing that detect and compensate for these defective pixels. In addition, expensive cameras have been found contain fewer defective pixels than older model cameras. Therefore, some cameras may not exhibit traces of defective pixels at all.

Another drawback of this approach is that the amount of visible pixels is dependent on scene content, which can hide or mask the presence of the defective pixels. The best way to find hot point defects is to take multiple images without removing the lens cap, or in low light conditions, and then averaging the images together. Any random noise will be removed and the hot point defects will remain. However, the image must still be searched manually to find the location of the defects. Once the locations of the defects are known, they can be easily spotted in other images from the same camera.

A test was performed to establish the consistency of defective pixels due to temperature [67]. They tested the cameras in temperatures ranging from 0 degrees Celsius to 40 degrees Celsius. It was found that as temperature decreased it was harder to locate the defective pixels. However, they do note

that when the location of a defective pixel was known, it could be easily identified in other images from the same camera.

Another test was performed to determine the influence that JPEG compression had on the visibility and location of the defective pixels [67]. They note that visibility and location was not affected significantly until the image had been compressed to about 50%. Due to the operation of the JPEG function, the pixels spots started to shift and spread out to the neighboring pixels depending on the position of the DCT matrix.

3. Authentication Framework Proposal

Since more digital images are being used in legal proceedings, we can expect the authenticity of digital images to be challenged more often. Currently, image authentication can be determined by witness testimony claiming that the photo is a fair and accurate portrayal of the scene [15]. This "witness authentication" scenario can be satisfied by the testimony of the person who took the image, or one who witnessed the scene that was depicted. In the absence of such a witness, an expert can be called upon to evaluate the provenance of the image by analyzing the image content and the details surrounding its acquisition. This chapter focuses on a framework to help analysts in a cognitive interpretation of the analyses discussed in chapter 2. The task of analytically authenticating a digital image is not an easy one. The nature of computer files is mathematical and not physical. A basic understanding of the image creation process is necessary to determine what artifacts and features are relevant to the investigation. Plus, the analyst must know the significance of these features and how they relate to the principle of individualization. While a statistical model would be preferable in these cases, they are sometimes not practical, or even possible for some aspects of the digital image authentication process. Therefore, analysts combine previous experience, known circumstances, and training in a cognitive evaluation to determine the significance of their findings.

The authentication framework is divided into four main areas of investigation; the *File Structure*, *Global Structure*, *Local Structure* analyses, and Source Image Identification [Figure 36]. File structure analyses will focus on the bytes of information that make up the digital image file. In addition, information about the image file, the EXIF data, will be used to determine if inconsistencies exist between an exemplar image and the suspect image. These types of analyses are usually black and white, either there are characteristics inconsistent with exemplar images or there are not. However, inconsistencies in this area do not necessarily mean that the content of the image has been altered, it is an indication that another program interacted with the file at some point.

File Structure	File Format
	Hex Data
	EXIF Data
Global Structure Analysis	Compression Level Analysis
	Color Filter Array
	Quantization Tables
	DCT Coefficients
Local Structure Analysis	Copy and Paste Detection
	Error Level Analysis
	DCT Map
Source Camera Identification	Defective Pixels
	PRNU

Figure 36 Image Authentication Framework

Next, the global structure of the image will be analyzed in reference to the image file format. For non-compressed images, artifacts consistent with the re-interpolation in the image and color filter array will be investigated. In addition, JPEG compression, while lossy, has unique characteristics that can be exploited to help identify artifacts consistent with alterations. These are most noticeable in the quantization tables and the DCT coefficients. The file and global structure analyses are based on a comparative analysis model. Exemplars should be used whenever possible to compare the file structure and global image structure of a suspect file to those of exemplar images. Characteristics of the suspect image are compared against exemplars using the same process (in these cases to a still camera), to determine its degree of similarity or dissimilarity. These indications

do not indicate that the original meaning of the content has been altered. For example, resizing an image and saving it as a JPEG does not mean that malicious alteration has occurred.

Next, local image analyses will be performed on the pixel level to determine if areas of alteration exist in the digital image structure. Utilizing errors caused by JPEG compression and sensor noise data, areas of alteration will disrupt the natural, mathematical relationship of values in the pixel data. Once altered, new relationships are formed between the pixels. Techniques that concern the local image structure are used to identify areas where this relationship changes. These techniques tend to be more statistical, and require the use of specially crafted algorithms to determine alteration. A positive indication of alteration using these types of analyses is very strong proof that alteration has occurred. Source image identification techniques are then used, if possible, to identify characteristics in the image that could provide information about the acquisition device.

One important aspect to note is that these analytical techniques can only provide positive proof of image tampering. The simple fact is that is extremely difficult (if not impossible [28]) to prove a negative, i.e. prove that an image is free of modification. In order to provide proof that an image has not been manipulated, one would need to apply all known techniques that uncover all known forms of manipulation. Even in doing so, the possibility may exist that an unknown manipulation technique was applied to the image that left no trace. The best that an analyst can do in these types of scenarios is to search for indications that support the proposition that the image was generated without modification.

While new techniques are being developed, the art of hiding manipulations may only become easier. Depending on the skill level of the manipulator, traces of alterations will be harder to detect. While certain manipulation techniques can elude one or more analyses, it may be difficult or even impossible to elude them all. It is up to the analyst to look at the image from every conceivable angle to find these traces. The simple fact is, however, that it may not be feasible, or even possible, to perform every technique on every image due to limited manpower, time, or financial resources. Therefore, it is important to apply what resources an analyst has in the most efficient way possible. In order to do this effectively an analyst must be resourceful with the tools that are available to them.

There are many commercial programs available that aid in the identification of alterations in digital images. Rigour, Tungstene, and the

Forensic Image Analysis System (FIAS) are software packages that perform numerous image authentication techniques on multiple facets of a digital image file [68-70]. In addition, image-processing software such as Adobe Photoshop, Corel Paintshop Pro, and GIMP, provide basic image processing tools that can be used to enhance image files or perform basic mathematical functions [71-73]. Furthermore, JPEGsnoop is a free software program designed to extract information from JPEG compressed image files [74]. Depending on the ability of the analyst, programmable software programs, such as MATLAB can be used to create custom algorithms that incorporate peer-reviewed authentication techniques [75].

For the following case studies the author only had access to the following image authenticating tools: FIAS, Photoshop, and JPEGsnoop. While these tools do not allow for image analysis using every possible authentication technique, this chapter shows how these tools can be used within the proposed analytical framework to provide as complete an examination as is possible. For the case studies presented in the next section, an image authentication table is used to record the results for each technique [Figure 36]. This table divides the analyses into file, global, and local image structure analyses. While the quality of image analysis techniques is constantly evolving, each area of the authentication table should be populated with analyses relevant to the structure of the file format. The results of each analysis will be marked by a green ✓ if the image shows no indications of alteration. The checklist will be marked with a red ✗ if the analysis shows any indication of alteration. A grey ○ will be used to signify if the analysis is inconclusive. Findings are then combined to determine how much evidence supports or weakens the authenticity of a digital image.

If an analyst uses every technique described in the proposed analytical framework and finds no indications of alteration, the author does not guarantee that the image will be authentic. The techniques presented in this paper only discuss those techniques that investigate the digital information that constitute the digital image file. Many other techniques are available to determine authenticity by examining the physics based characteristics of the scene content such as lighting inconsistencies, historical inaccuracies, anachronisms, color aberration and geometry. Whenever possible, these techniques should be used in conjunction with the following framework.

3.1 Case Study 1

In this particular case study, an image is presented for authentication [Figure 37]. The object in question is a hammer that is present on the table. Investigators would like to know if someone manipulated the image by inserting the hammer onto the table after the initial image was taken. The camera stated to have taken the picture, a Samsung HZ50W, was available for the analysis.

Figure 37 Case 1 - Suspect Image
Image suspected of manipulation. Object in question is the hammer on the table.

The first step is to verify that the suspect camera is capable of taking the image. If so, the best way to verify the analyses is to make a comparison of the suspect image to exemplars taken with the suspected camera. A review of the file structure reveals that the photo is a JPEG image with a resolution of 2048 x 1536. The information in the EXIF reveals that three different camera models could have taken the image: the Samsung WB5500, VLUU WB5500, and the Samsung H50W [Figure 38]. This would indicate that it is possible that the suspect camera, Samsung HZ50W, could have taken the image. The user's manual of the HZ50W digital camera confirms that a JPEG image at the specified

resolution is available with the camera set to the '3M' resolution setting. It is worth noting that the resolution setting of '3M' only outputs a JPEG file at a resolution of 2048 x 1536. If any manipulations were to be made to an original JPEG image from this camera, it would have to be resaved as a JPEG. Thus, there may be indications of recompression if this occurred, and the best way to determine this is to compare the suspect image to an exemplar taken from the camera. However, there are three different JPEG compression settings for this particular camera, and it must be determined which one was used during the image acquisition.

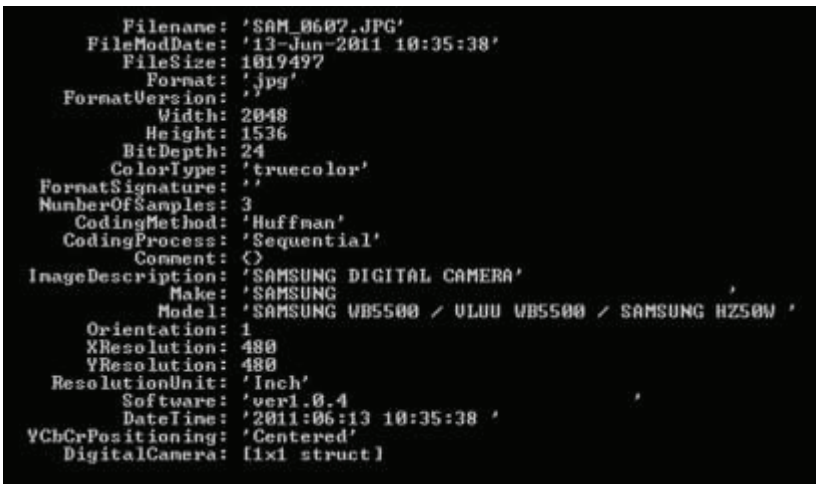


Figure 38 Case 1 - Suspect Image EXIF

Analysis of the EXIF data using the Samsung Intelli-Studio image viewer showed that the suspect image contained indications consistent with a picture taken using the camera's 'SmartAuto' shooting mode [Figure 39]. This was confirmed by taking exemplars from the Samsung HZ50W at each compression setting and comparing the *ExposureProgram* description to that of the suspect image. The 'SmartAuto' is the only compression setting that returns a value of 'NormalProgram.'

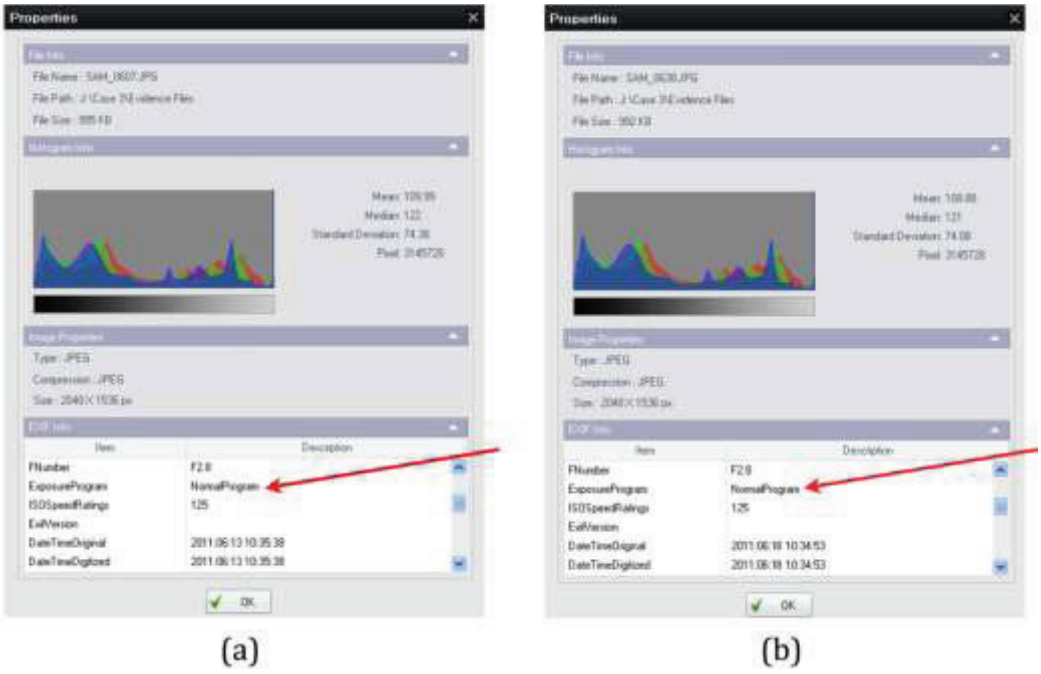
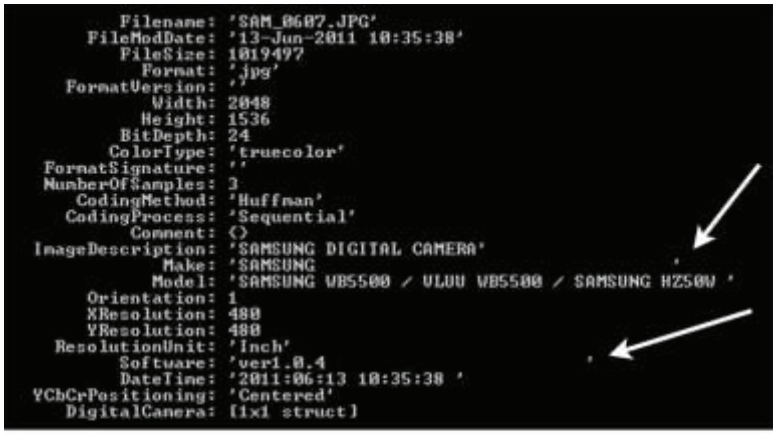
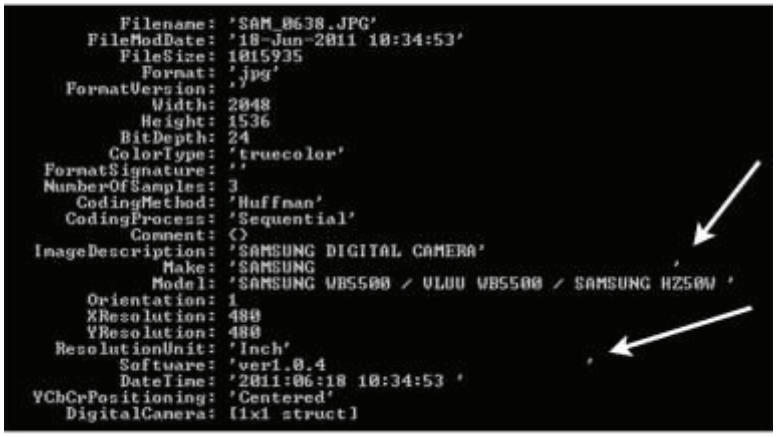


Figure 39 Case 1 - EXIF View Using Samsung Intelli-Studio
Intelli-Studio screen shot of the EXIF for SAM_0607 (a) and exemplar taken with the SmartAuto compression setting (b) to determine the exposure setting. 'NormalProgram' in the EXIF description indicates the 'SmartAuto' exposure setting for the camera.

The EXIF data was examined with both FIAS and a visual inspection of the file using a hex viewer. The information and formatting of the suspect image is consistent with those of the exemplars taken at the 3M resolution setting on SmartAuto mode with the Samsung HZ50W camera. Even the odd placements of the apostrophes in the *Make* and *Software* fields are consistent [Figure 40]. A search of the hex data for image manipulating software signatures (See Appendix A) returned no results.



(a)



(b)

Figure 40 Case 1 - EXIF Comparison
EXIF of the questioned image SAM_0608.JPG (a) and an exemplar (b). There are no inconsistencies between the two. Even the oddly placed apostrophes (marked with arrows) are in the same place.

Analysis of the quantization tables was accomplished by comparing multiple exemplars taken with the camera, with the suspected resolution and compression setting, in wide range of lighting conditions. Extracting the quantization tables from the exemplars reveal that a wide range of tables are

used by the camera on the 'SmartAuto' setting. The variability of the quantization tables for the camera is quite large and is represented in the bottom two panels of Figure 41. Luckily, images taken in similar lighting conditions produced a quantization table similar to the suspect image [Figure 41 (b)]. This particular camera uses a custom adaptive quantization table, which changes depending on the scene content. Different quantization tables will have an effect on analyses that investigate the interpolation of the image, like compression level analysis and color filter array, because of the JPEG compression artifacts. For the remainder of this authentication, the suspect image will be compared to exemplar images with the same quantization table.

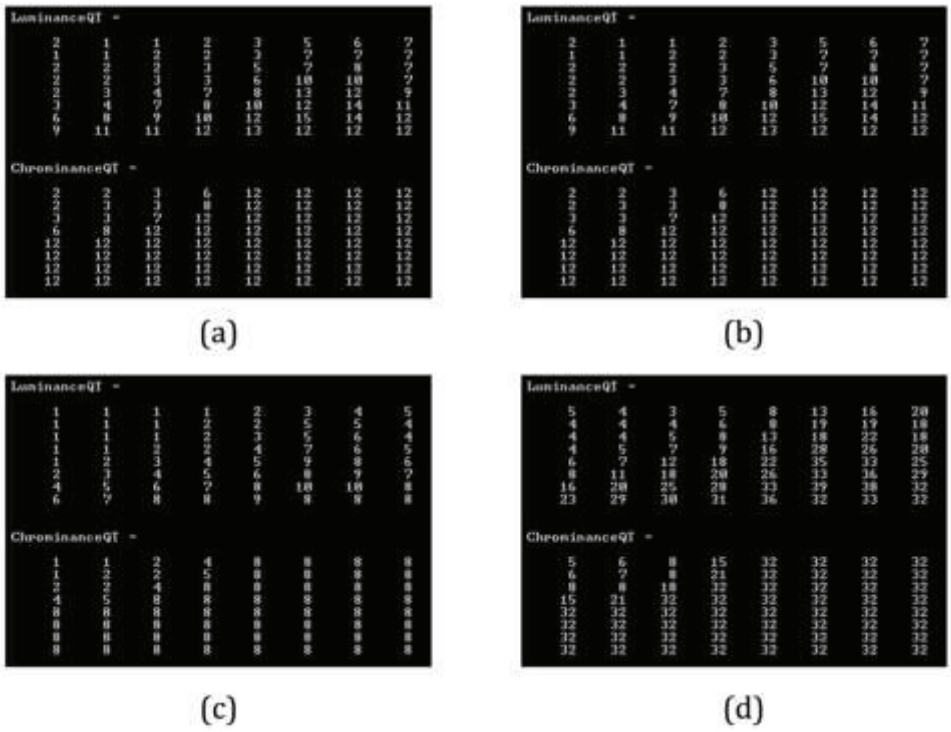


Figure 41 Case 1 - Quantization Tables

Quantization tables of questioned images SAM_0607.JPG (a) and three exemplars taken from the suspect camera with the same compression settings (b - d). Note the wide range of quantizer steps between c and d indicative of a camera able to produce custom adaptive tables.

Analysis of the color filter array produced no identifiable artifacts indicative of image manipulation or recompression [Figure 42]. Contour comparisons for each of the color layers in the suspected image are made to the exemplar and are similar. In addition, there are no traces of recompression of the suspect image, which show up as pronounced spikes, deep valleys, and/or different shapes in the graph contours.

A comparison of the compression level analysis graphs shows close similarities. Each graph shows low level noise and a large central spike, which is usually an indication of a first generation JPEG image [Figure 43]. Because the CLA graph of the suspect image matches that of the exemplar, it is consistent with an image that was not recompressed or resized.

DCT coefficient analysis of the questioned image against the exemplar show very similar shapes and contours [Figure 44]. The values of the DC coefficient have the same range from -1000 to 1000 [Figure 44 (a, c)]. If these images had been manipulated, it may be possible for these values to shift and change. In addition, a close up view of the DC components show spikes at the same integer values, indicative that the coefficients are grouped into evenly spaced multiples of the quantizer step size. Furthermore, there is no indication of repeating patterns deviating from a first generation JPEG image. There is also no indication that the suspect image was recompressed, which would be expected if an external program had been used to manipulate it and then resaved it as a JPEG image.

Detection of copy and pasted pixels revealed no results. A review of the DCT map shows no particularities of interest. Indications of manipulation would show up as areas of differing texture that are inconsistent with other objects in the surrounding image [Figure 45]. The suspect image has no such identifiable features. While the hammer could arguably contain such a feature, its intensity is consistent to the other objects just above the hammer on the table.

Error level analysis revealed no indications of JPEG ghosting [Figure 46]. This means that at least, the image was not manipulated by placing a previously, lower recompressed portion of an image, into the suspect one.

To search for defective pixels, 10 images were taken with the lens cap on using the camera settings indicated by the EXIF. The images were then averaged together to remove shot noise. There were no defective pixels found in the test images using the camera settings as indicated by the EXIF. A PRNU test was not performed because the strength of the PRNU signature using one image is not reliable with the tools available to the author.

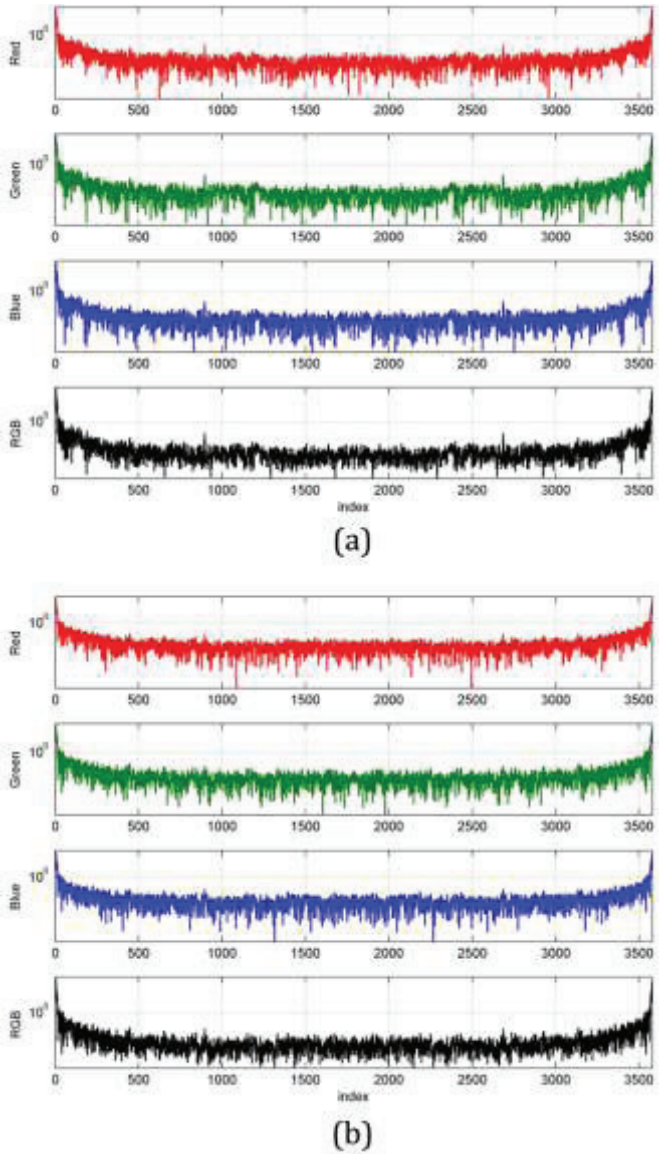


Figure 42 Case 1 - Color Filter Array Analysis
Color filter array graphs of the questioned image (a), and an exemplar (b).

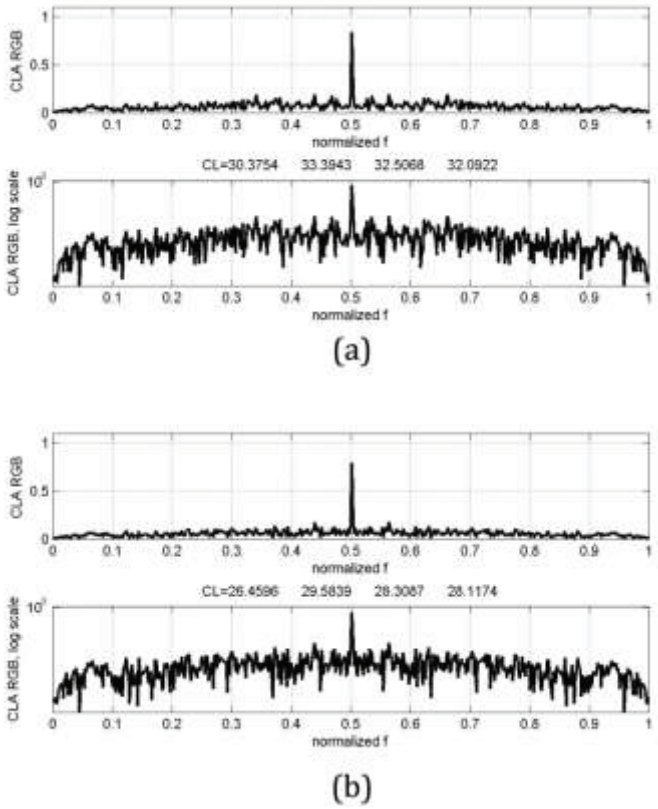


Figure 43 Case 1 - Compression Level Analysis
Compression level analysis of the questioned image (a) and the exemplar (b). For each panel, the top graph represents the plot using a linear scale, while the lower one uses a logarithmic scale.

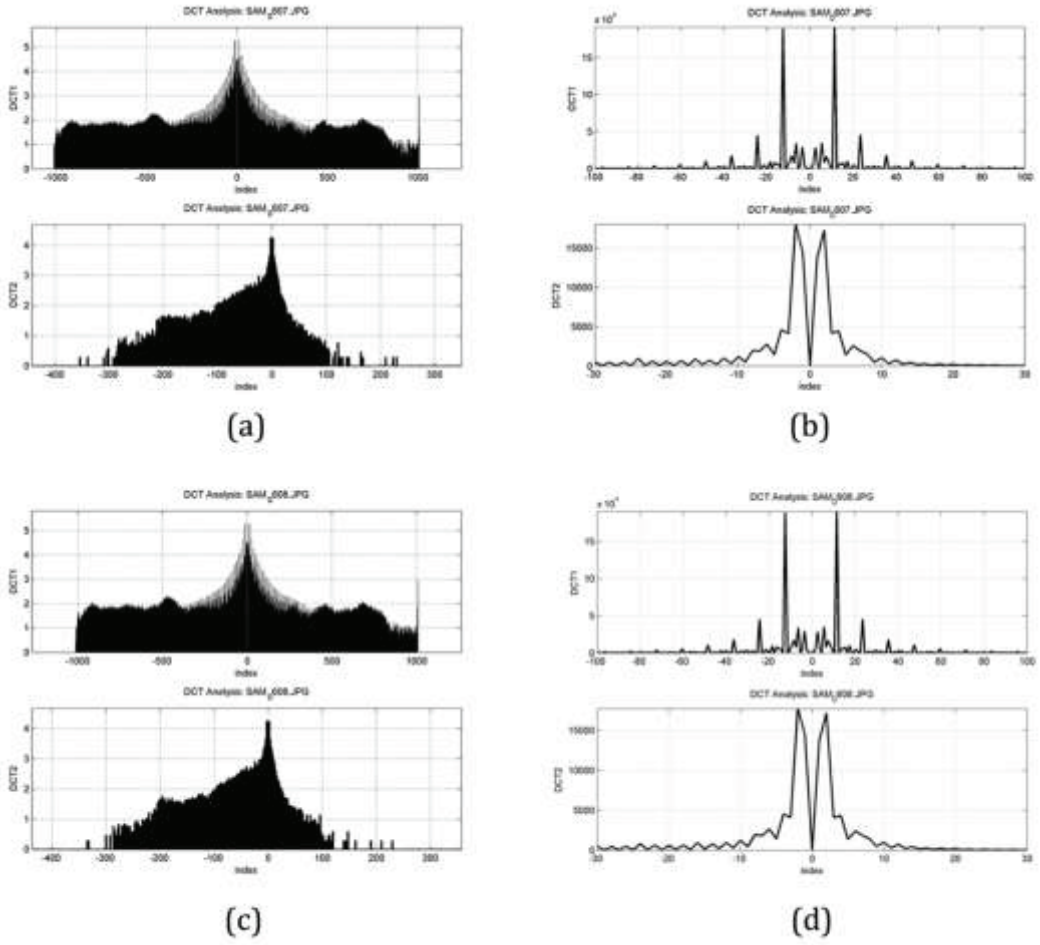


Figure 44 Case 1 - DCT Coefficients

Histogram of the DCT coefficients of the questioned image (a, b) and the exemplar (c, d). The left column graphs show the DCT coefficients for the DC component (top) and the AC coefficient of [1, 2] (bottom). The right column is a magnification of the central spike for each respectively.

Figure 45 Case 1 - DCT Map
Shown here is the AC component map, which is made from an average of all AC components in each JPEG block. There are also no apparent indications of manipulation in the DC component map (not shown). The image has been cropped to show more detail in the area of interest.

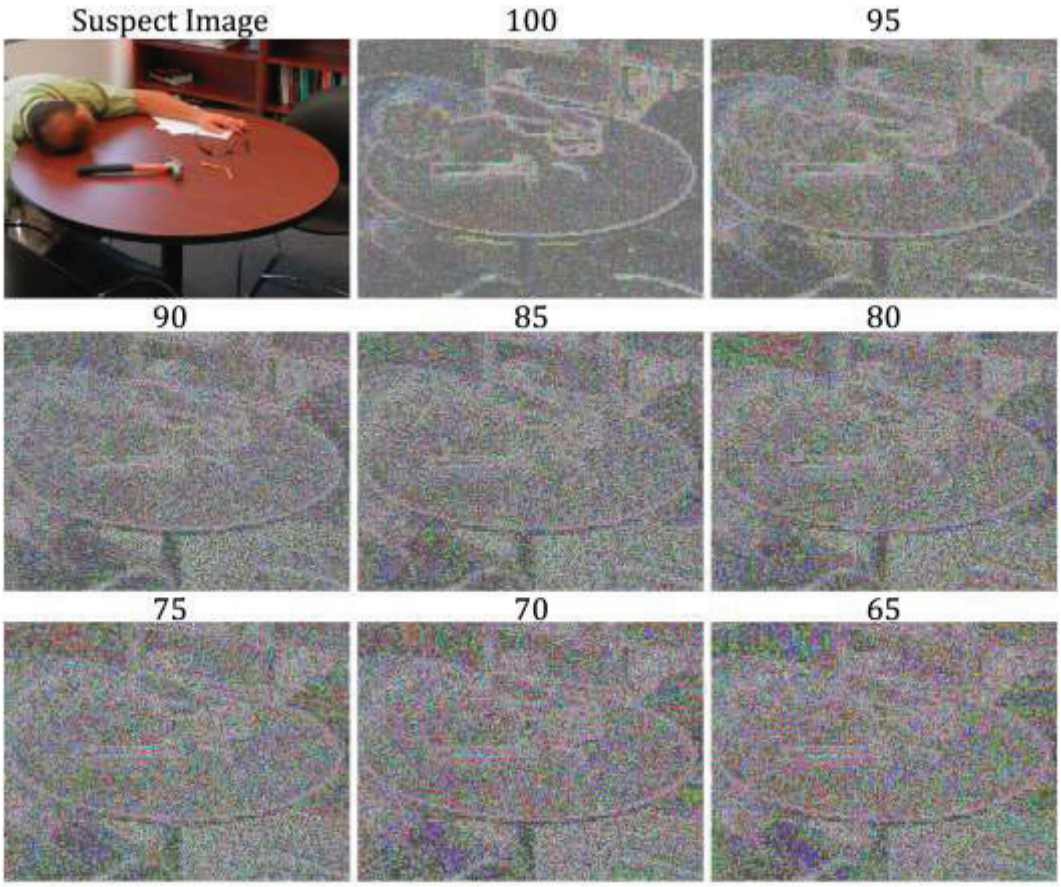


Figure 46 Case 1 - Error Level Analysis
Shown in the upper left corner is a cropped portion of the suspect image. The subsequent panels show the difference between the suspect image and recompressed versions at varying compression levels [100-65]. There are no indications of JPEG ghosts.

There are no clear indications of manipulation in the suspect image around the hammer or otherwise [Figure 47]. In this particular case, comparisons could be made to an exemplar image taken from the same camera, which resulted in no inconsistencies between the two images. There are no indications of recompression in the suspect image that would be expected from a JPEG image that had been manipulated and resaved as a JPEG file. The hammer, and the area around the hammer, shows no artifacts consistent with being added or manipulated with an image-editing program[2].

File Structure	File Format	✓
	Hex Data	✓
	EXIF Data	✓
Global Structure Analysis	Compression Level Analysis	✓
	Color Filter Array	✓
	Quantization Tables	✓
	DCT Coefficients	✓
Local Structure Analysis	Copy and Paste Detection	✓
	Error Level Analysis	✓
	DCT Map	✓
Source Camera Identification	Defective Pixels	N/A
	PRNU	N/A

Figure 47 Case 1 - Authentication Table Results

[2] The image used in this case study was not altered in any way.

3.2 Case Study 2

In this case study, an image is submitted for authenticity [Figure 48]. The camera stated to have taken the image, an Olympus C5500Z, is available for the analysis.

Figure 48 Case 2 - Suspect Image

The JPEG image, size 1.2 MB, has a resolution of 1600 x 1200 and is named DCOS-1246.jpg. The EXIF indicates that either an Olympus C55Z or a C5500Z took the image, which is consistent with the model claimed to have taken the image [Figure 49]. The next step is to compare the suspect image to exemplars taken from the camera using the same settings. The only setting of the camera capable of taking an image at the specified resolution is the 'SQ1' setting. There are two different compression schemes for this setting; Normal and High. In order to determine what compression level was used, exemplars are taken from the camera at each compression setting and the information in the EXIF is compared to determine what field indicates compression setting. In this instance, a value of '2/1' in the 'Image compression' indicates the compression setting of the camera was set to 'Normal' compression [Figure 50].

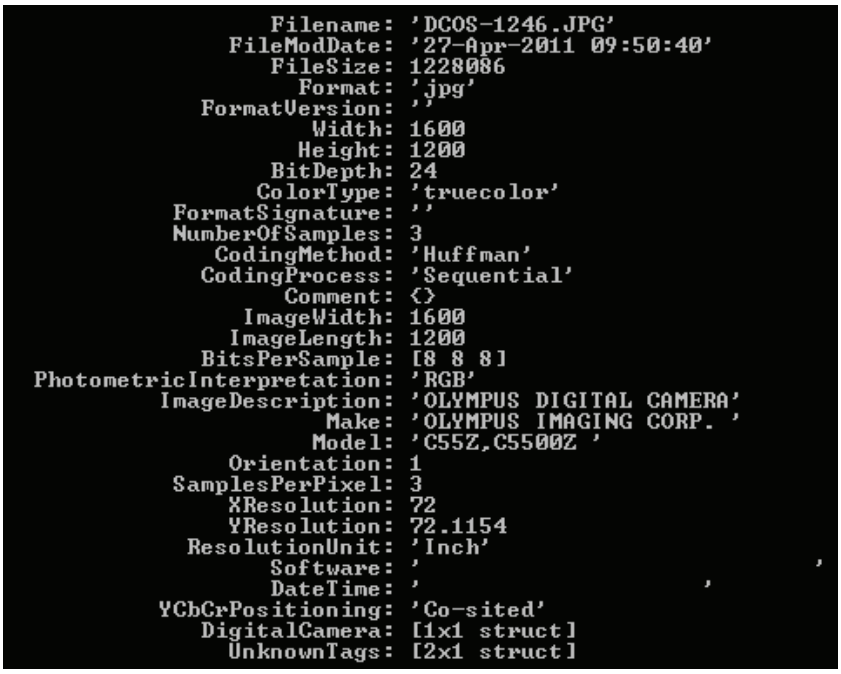


Figure 49 Case 2 - Suspect Image EXIF

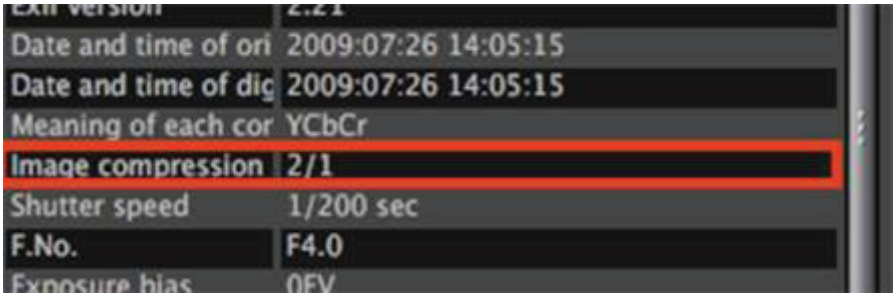


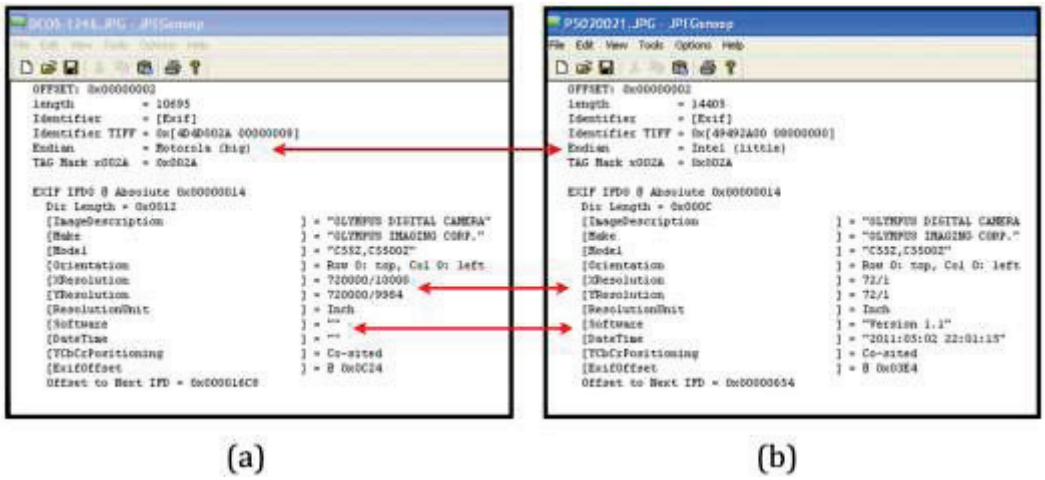
Figure 50 Case 2 - EXIF Using the Olympus Viewer 2

This figure is a view of the EXIF using the Olympus software, Olympus Viewer 2, ver. 1.21. Compression level settings can be determined by the value in the 'Image compression' field. A value of '2/1' indicates the camera setting was set to 'Normal,' while a value of 5/1 indicates 'High.'

After determining the camera settings, exemplars are taken with the suspected camera on the 'SQ1' resolution and 'Normal' compression setting. The first inconsistency is the naming convention of the camera. Exemplars taken with the camera had a file name of "P50xxxxx.jpeg." The suspect image started with "DCOS." In addition, the average size of the exemplar files were around 316 KB, almost a fourth of the size of the suspect image file's 1.2 MB.

The EXIF of the suspect and the exemplar images are compared. There is information that is inconsistent in the EXIF of the suspect image when compared to the exemplar [Figure 51]. The first item that is not similar to the exemplar is the ordering of the subcomponents in the digital file. The *Endian* field, which indicates the byte ordering, specifies that the ordering is 'Motorola (big)' in the suspect image, while it is shown be 'Intel (little)' in the exemplars. In addition, there is something odd in the representation of the 'XResolution' and 'YResolution' fields. The exemplar values indicate '72/1' resolution, but the suspect image has rather larger numbers in both the numerator and denominator of the fraction. In the case of the 'YResolution,' this number does not equal the value of 72/1, which can also be seen as '72.1154' in Figure 49. Likewise, the information for both *Software* and *DateTime* are missing in the suspect image. The camera populates these fields with the firmware version of the camera and the time the image file was created. Either the suspected camera did not create this image or a software program had altered it.

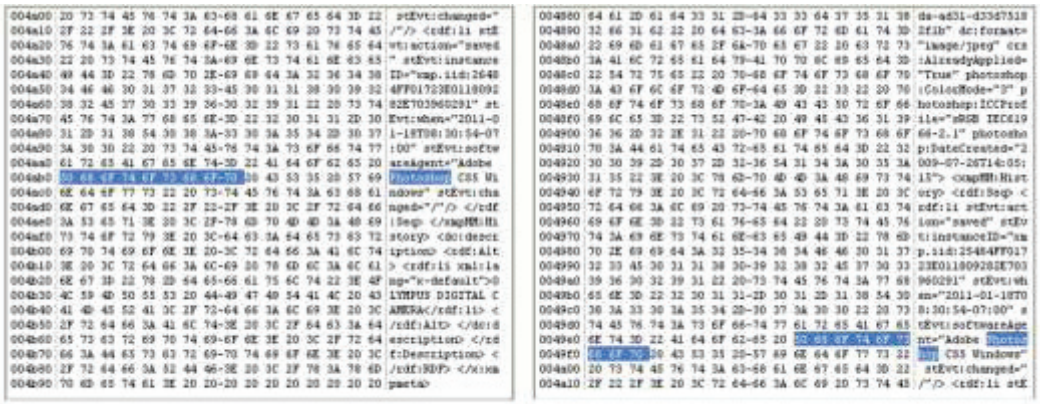
The next step in the process is to search the hex data for traces of image manipulating left by image processing software. The search terms used for this case are presented in Appendix A. The term "Photoshop" was found in the hex data multiple times throughout the digital image file [Figure 52]. This is an indication that Adobe Photoshop interacted with the image at some point. According to the hex data, the version of the Photoshop software used was version CS5 for Windows.



(a) (b)

Figure 51 Case 2 - EXIF Information Using JPEGsnoop

Comparison of the EXIF information present in the suspect image (a) and an exemplar (b). Items of inconsistent information are indicated with arrows.



(a) (b)

Figure 52 Case 2 - Hex Search

Indications that Photoshop was used were found in a search of the hex data. There are more but only two are shown here.

Moving on to the Global structure analysis, the compression level of the suspect image is checked against the exemplars. As can be seen, the exemplar has interpolation artifacts, which are not present in the suspect image [Figure 53]. While the exact reasons these spikes are present in the exemplar are unknown, there are two possible contributing factors. First, the native resolution size of the C5500Z camera is 2592 x 1944. When the resolution of the image is changed to record pictures at 1600 x 1400, the resolution setting of the camera at SQ1, some information from the sensor has to be either discarded, or condensed into the smaller resolution size. The other possible cause of the artifacts is the compression algorithm used by the camera. Whatever the reason, these artifacts are a product of the camera's normal operation, and are not present in the suspect image.

Analysis of the color filter array shows similar inconsistencies between the suspect images and the exemplar [Figure 54]. The contour of the CFA for the suspect image does not match those of the exemplar. In addition, there are very pronounced valleys in the CFA that are not present in the suspect image. Again, this is most likely due to the re-interpolation of the image from the sensor size, 2592 x 1944, down to 1600 x 1400, in the exemplar images.

Analysis of the quantization table of the suspect image reveals it to be quite different from that of the exemplar [Figure 55]. Furthermore, the quantization tables of the suspect image matches those used by Photoshop, specifically when the image is compressed at a setting of '12' [43].

Analysis of the DCT coefficients reveals inconsistencies between those of the suspect image and the exemplar [Figure 56]. The periodicity caused by the quantization table has grouped many DCT coefficient values into fewer bins in the exemplar. This is most easily seen in the AC coefficient panel of Figure 56 (c). This pattern is not present in the DCT coefficients of the suspect image. Furthermore, the range of values of the DCT varies greatly, most notably in the AC coefficients [Figure 56]. The suspect image range of values for the AC components is between [-600, 150], while those of the exemplar are [-170 20]. However, this could be a product of the dynamic range difference in lighting between the two images. More than that however, the DCT coefficients in the exemplar are grouped into very identifiable bins. The DCT coefficients in the suspect image, which are smoothly distributed across a wider range of values, do not exhibit the same characteristics.

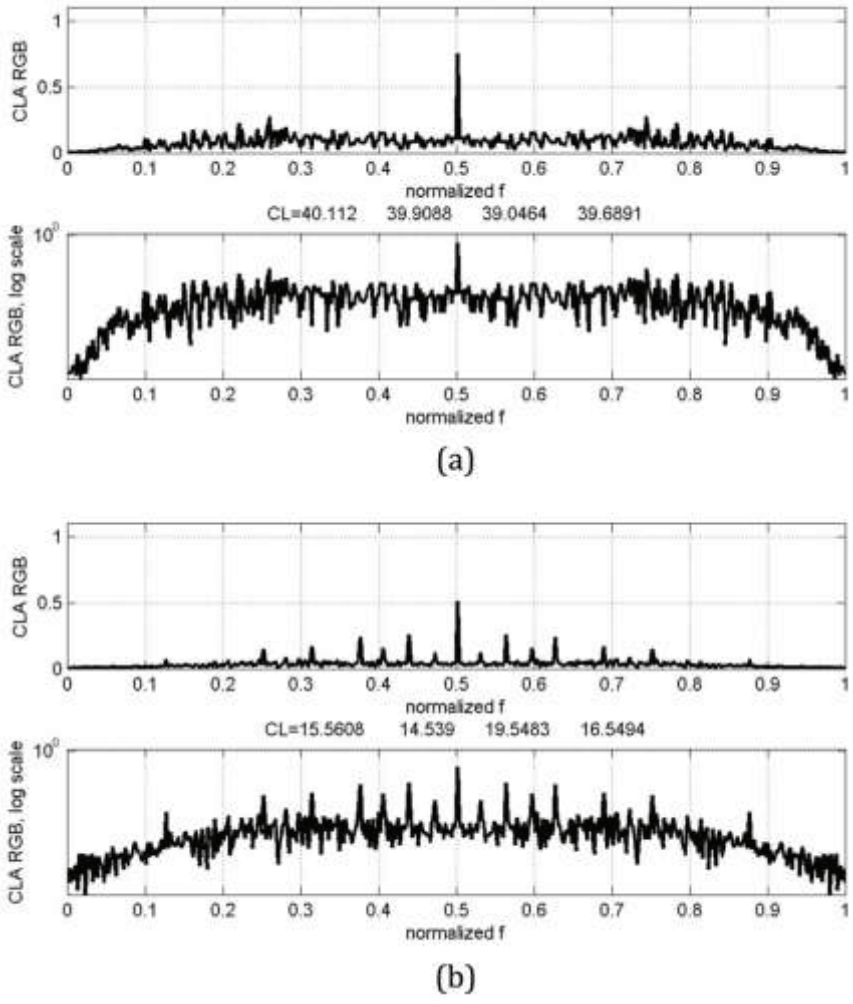


Figure 53 Case 2 - Compression Level Analysis

Compression level analysis of the suspect image (a) and the exemplar (b). For each panel, the top graph represents the plot using a linear scale, while the lower one uses a logarithmic scale.

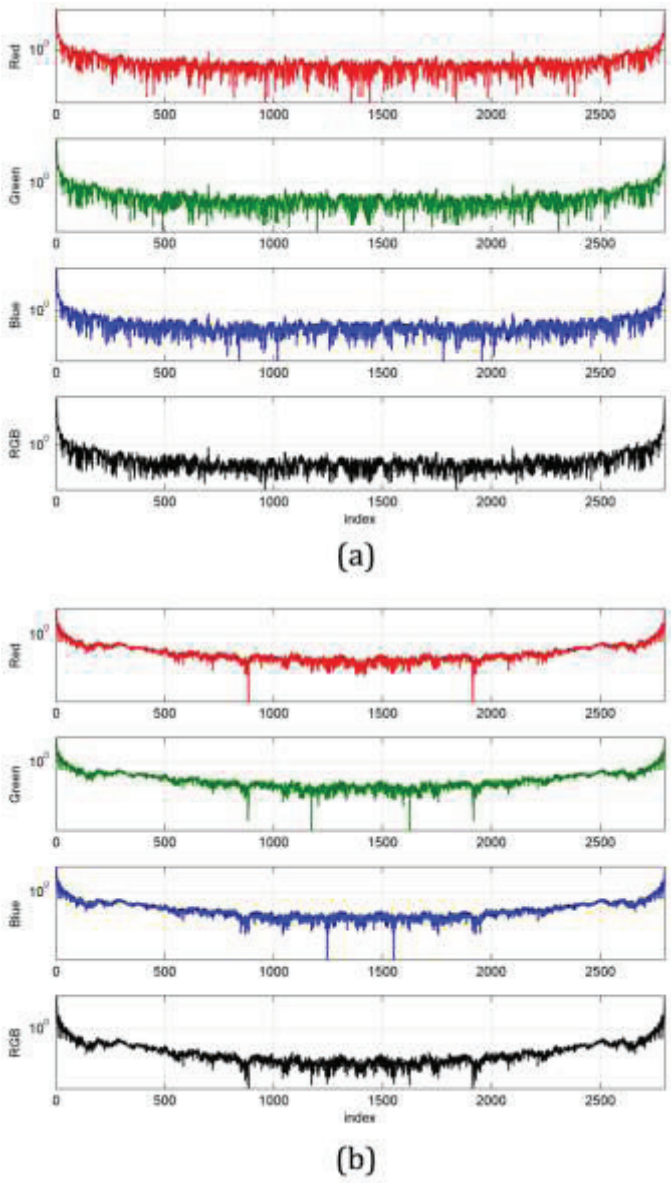


Figure 54 Case 2 - Color Filter Array Analysis
Color filter array graphs of the questioned image (a), and the exemplar (b).

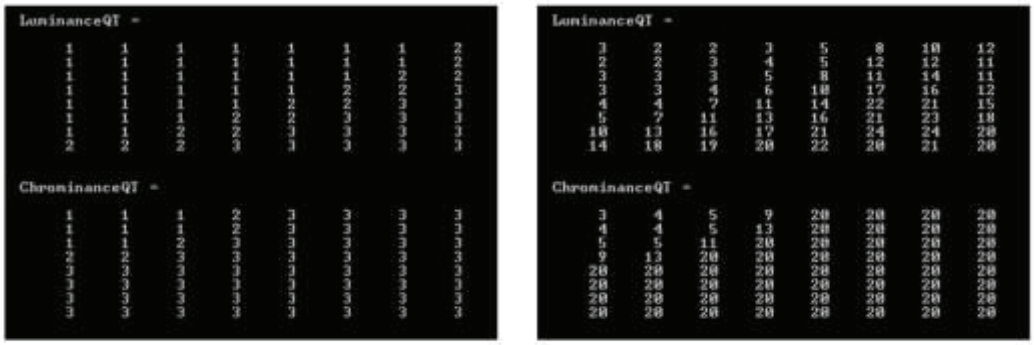


Figure 55 Case 2 - Quantization Tables
Quantization tables from the suspect image (a) and the exemplar (b).

Analysis of the DCT map, shows that there is an inconsistency in the center of the image that is an indication of manipulation [Figure 57]. The area of manipulation has a different texture than the rest of the image. In this particular case, the area of interest is considerably lighter than the surrounding area. This mass has no explainable cause by any object in the suspect image. In fact, grass and flowers similar to the surrounding area cover the suspect region, yet do not exhibit the same characteristics. In addition, analysis of the image for JPEG ghosting reveals indications as well [Figure 58]. The JPEG ghost is most prominent when Q = 80 and 90. A strange anomaly occurs at Q = 85 when the mass intensity decreases suddenly. The cause of this is unknown.

A copy-and-paste analysis was performed on the suspect image to identify if any "within image" content was cloned onto the affected area. There were no indications of copy-and-paste until the search block size was reduced to 2 x 2. Even with this small block size, areas were not well defined [Figure 59]. However, a closer visual inspection of the indicated area revealed the copy-and-pasted portions were significantly larger than indentified, along with other anomalies [Figure 60]. A visual inspection of the indicated areas show abrupt gradient changes and blur along the edges of the identified mass. In addition, a closer look at the area within the mass revealed that several elements from the image were copy and pasted inside the identified area. The reason larger areas of copy-and-paste were not identified was probably due to noise introduced at the time of manipulation and/or obfuscated by JPEG compression.

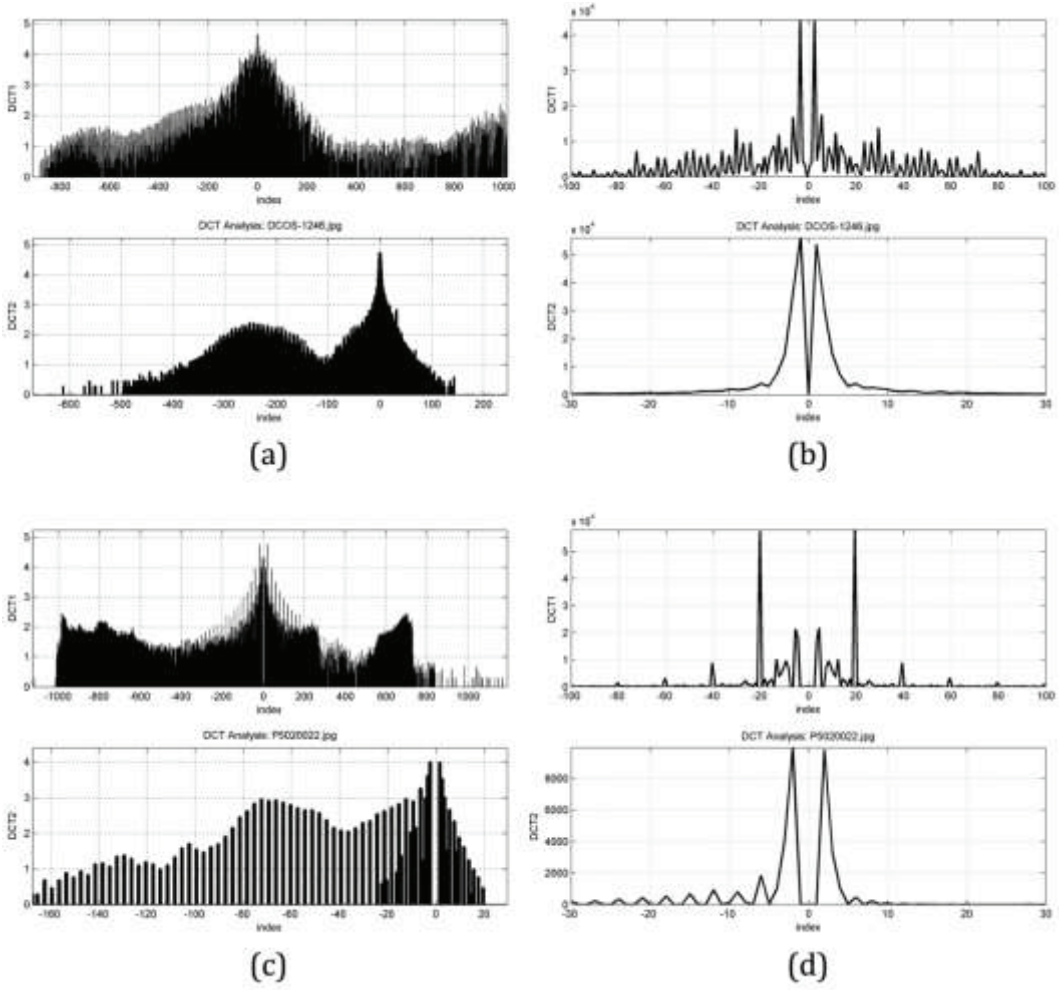


Figure 56 Case 2 - DCT Coefficients

Histogram of the DCT coefficients of the questioned image (a, b) and the exemplar (c, d). The left column graphs show the DCT coefficients for the DC component (top) and the AC coefficient of [1, 2] (bottom). The right column is a magnification of the central spike for each respectively.

To search for defective pixels, 10 images were taken with the lens cap on with the camera settings set to the specifications indicated by the EXIF. The images were then averaged together to remove shot noise. There were approximately 12 hot pixels found on the test images. The suspect image had no defects at the locations specified by the test images.

A PRNU test was not performed because the strength of the PRNU signature using one image is not reliable with the tools available to the author.

Figure 57 Case 2 - DCT Map

Shown here is the AC component map, which is made from the average of all the non-zero AC components in each JPEG block. Similar indications of manipulation appear in the DC component map as well (not shown).

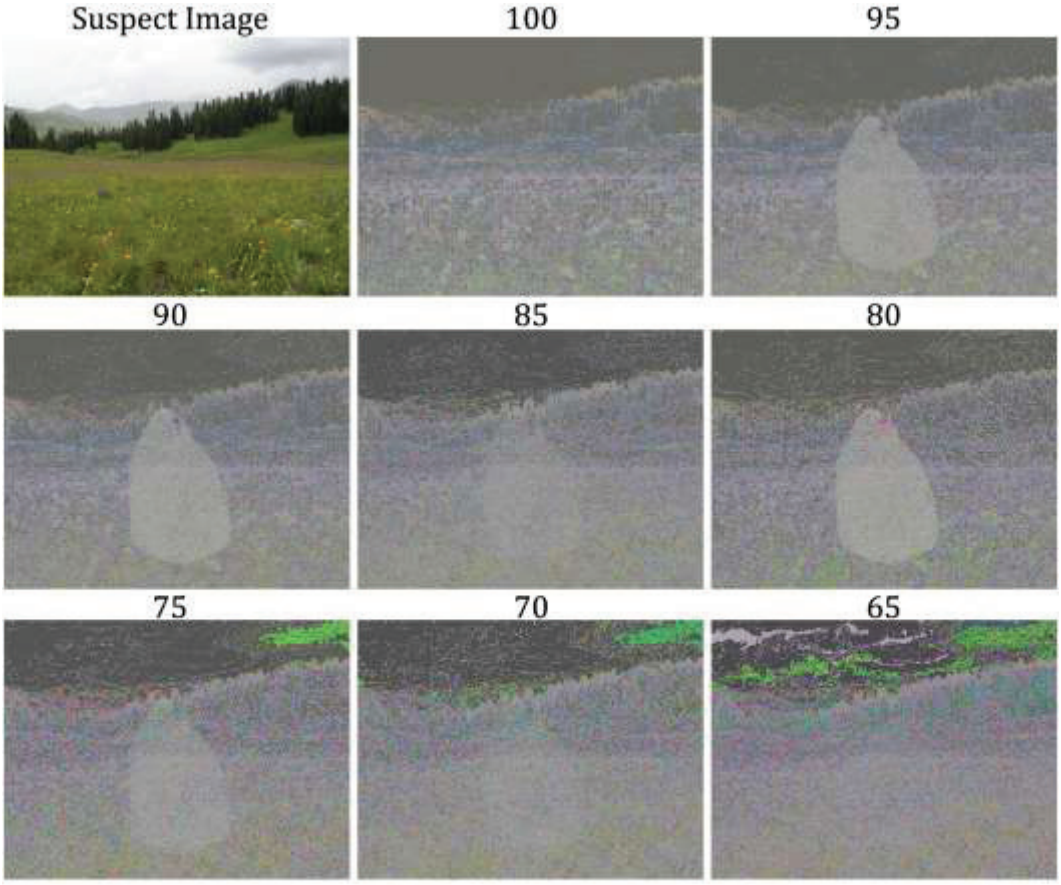


Figure 58 Case 2 - Error Level Analysis
Shown in the upper left corner is a cropped portion of the suspect image. The subsequent panels show the difference between the suspect image and recompressed versions at varying compression levels [100-65]. The area of interest is easily seen.

Figure 59 Case 2 - Copy-and-Paste Analysis
A copy and paste analysis was run against a cropped portion of the suspect image. No results were indicated until the search block size was reduced to 2 x 2.

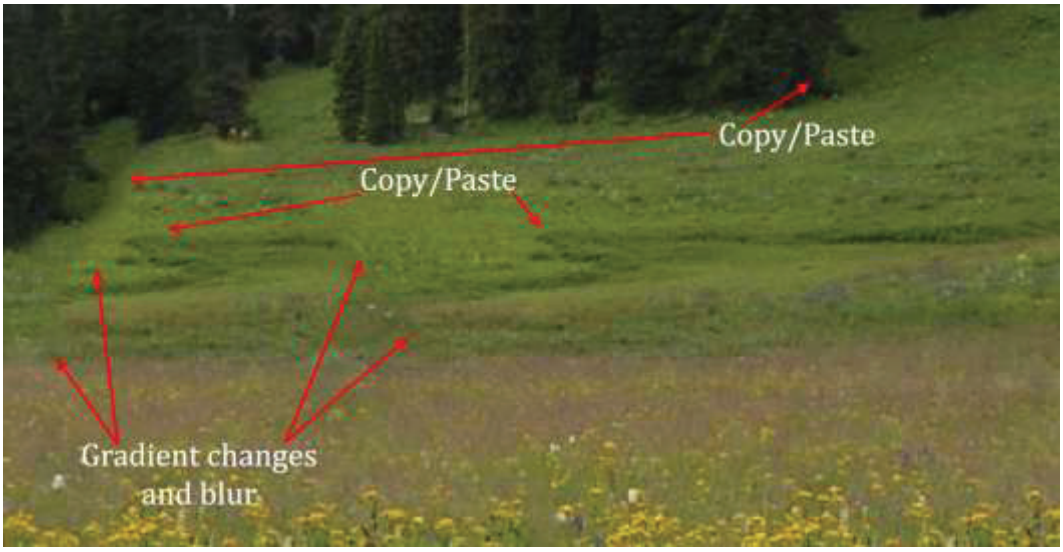


Figure 60 Case 2 - Visual Analysis
Visual verification of manipulation in a cropped portion of the suspect image. Areas of abrupt gradient changes and blur are noted, as well as two areas that have been copied and pasted.

Figure 61 shows the results of the analyses. The suspect image file contains clear indications of alteration as shown by the analysis of the header information, compression analysis, quantization table, DCT coefficients, DCT map, correlation map, and lack of defective pixels. In addition, there are very clear visual signs of manipulation in the image with artifacts consistent with a copy/paste and blur tool[3]. Furthermore, the suspect image contains indications that it may have not been a product of the suspect camera as indicated by the lack of defective pixels in the image and lack of consistency in the EXIF.

File Structure	File Format	✗
	Hex Data	✗
	EXIF Data	✗
Global Structure Analysis	Compression Level Analysis	✗
	Color Filter Array	✗
	Quantization Tables	✗
	DCT Coefficients	✗
Local Structure Analysis	Copy and Paste Detection	✗
	Error Level Analysis	✗
	DCT Map	✗
Source Camera Identification	Defective Pixels	✗
	PRNU	N/A

Figure 61 Case 2 - Authentication Table Results

[3] The original image and how it was manipulated is explained in Appendix C

3.3 Case Study 3

In this case study, an image is submitted for authenticity [Figure 62]. The objects in question are the hand bells appearing multiple times in the image. The camera stated to have taken the image, a Canon 7D, is available for the analysis.

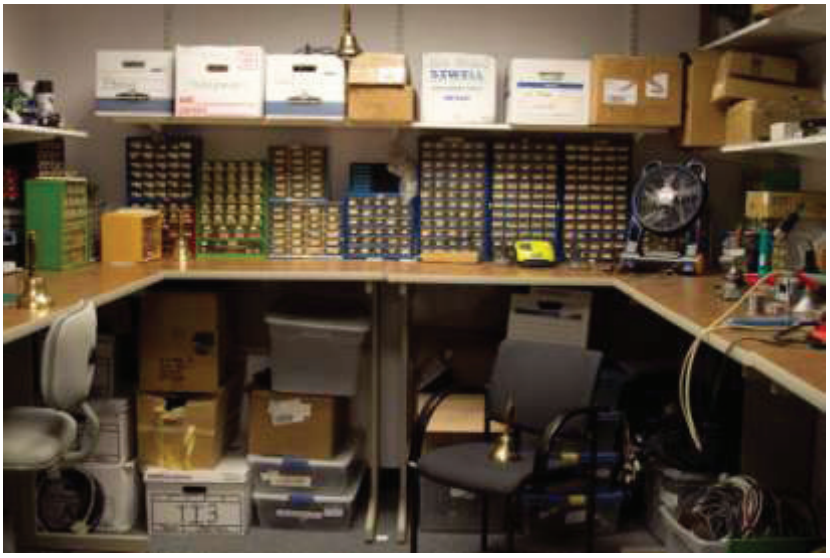


Figure 62 Case 3 - Suspect Image

The JPEG image, size 1.6 MB, has a resolution of 2592 x 1728 and is named IMG_0799.jpg. The EXIF indicates that the image was taken with a Canon 7D, which is consistent with the model claimed to have taken the image [Figure 63]. The resolution of the image, 2592 x 1728, is available on this camera for the following settings: S-RAW and JPEG Small. Because this image is a JPEG image, focus will be concentrated on the JPEG settings. There are two compression settings, normal and fine, which will need to be determined in order to compare the suspect image to exemplars taken using the same settings. Using the Canon image viewing software, Digital Photo Professional, it can be determined from the *Image Quality* field what compression setting was used at the time of image capture. A value of 'Normal' indicates the Normal compression setting and 'Fine' indicates the fine compression setting. The suspect image EXIF indicates it was captured using the normal compression setting [Figure 64]. It was noted at this

time that the thumbnail of the JPEG image was not showing using the Canon Digital Photo Professional. In addition, the software would not open using Photoshop because of an "unknown or invalid JPEG marker type".

A search of the hex data for the image processing software search terms in Appendix A returned no matches. In addition, there are no forensically relevant differences between the EXIFs of the suspect image and those of exemplars taken from the camera Figure 65.

It was also determined that the quantization table of the 'Normal" JPEG compression of the Canon 7D remained constant under a wide variety of lighting conditions and scene content. This characteristic of this quantization table most closely matches those of a *Standard Quantization* table [44]. In addition, the QT of the suspect image matched those of the exemplars taken using the 'Normal' compression setting [Figure 66].

A comparison of the color filter array graphs of the suspect and exemplar images show close similarities in spikes and contours [Figure 67]. However, there appeared some sharp dips in the graph that were present in many of the exemplar images, but not in the suspect [Figure 67 (b)]. This type of indication is not very strong due to the effect that scene content has on the color filter array analysis. Variety of the CFA dips was observed in many of the exemplar images, some exhibiting stronger dips than others in all color layers. The result of this analysis in inconclusive due to differences observed, but the exact reason for the difference is unknown.

A comparison of the compression level analyses graphs returned some inconsistencies. A graph from the suspect image shows a low noise level and a single central spike [Figure 68 (a)]. However, CLA of exemplar images indicate more spikes should be present during the creation of images from the Canon 7D. Figure 68 (b) is the graph from a sample exemplar image. The spikes circled in red were consistent in all of the exemplar images taken in a wide range of lighting conditions and scene content. Although the strengths of the spikes varied from image to image, they were constant in all images. The additional spikes in Figure 68 (b), indicated by the red arrows, were not always present in all images. However, if they were present their location was consistent with those of the other exemplar images. Indications of the additional spikes are not present in the suspect image.

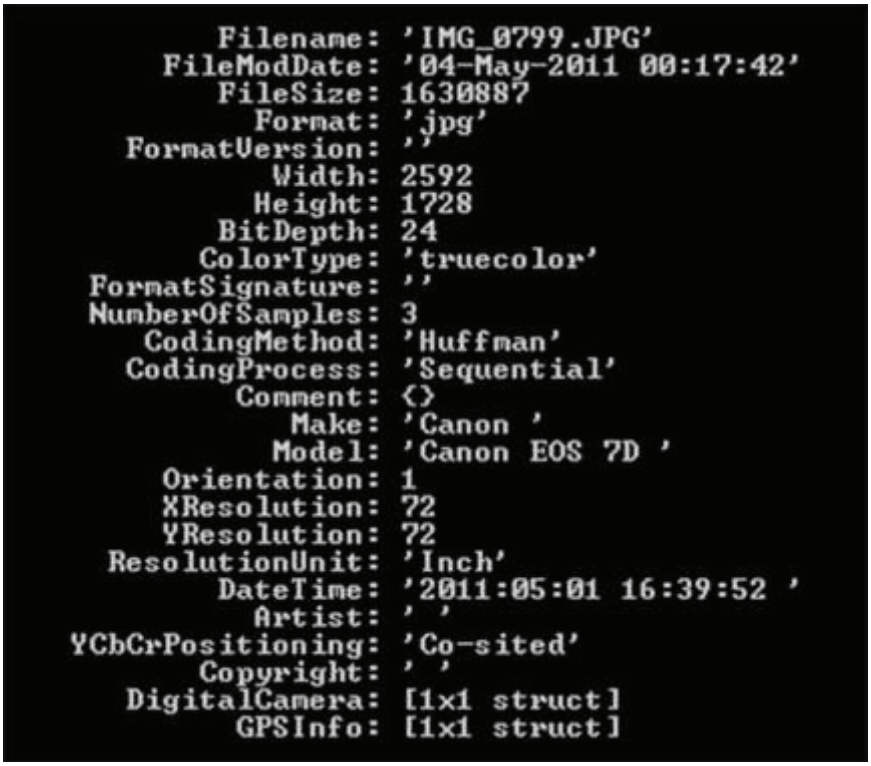
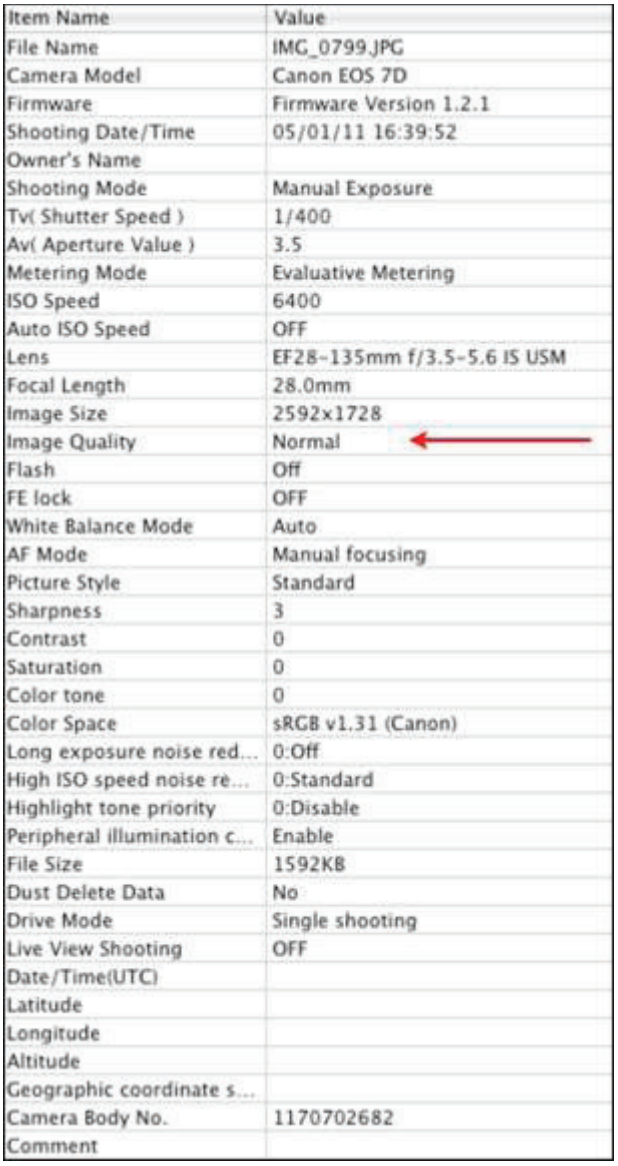


Figure 63 Case 3 - Suspect EXIF



Item Name	Value
File Name	IMG_0799.JPG
Camera Model	Canon EOS 7D
Firmware	Firmware Version 1.2.1
Shooting Date/Time	05/01/11 16:39:52
Owner's Name	
Shooting Mode	Manual Exposure
Tv(Shutter Speed)	1/400
Av(Aperture Value)	3.5
Metering Mode	Evaluative Metering
ISO Speed	6400
Auto ISO Speed	OFF
Lens	EF28-135mm f/3.5-5.6 IS USM
Focal Length	28.0mm
Image Size	2592x1728
Image Quality	Normal
Flash	Off
FE lock	OFF
White Balance Mode	Auto
AF Mode	Manual focusing
Picture Style	Standard
Sharpness	3
Contrast	0
Saturation	0
Color tone	0
Color Space	sRGB v1.31 (Canon)
Long exposure noise red...	0:Off
High ISO speed noise re...	0:Standard
Highlight tone priority	0:Disable
Peripheral illumination c...	Enable
File Size	1592KB
Dust Delete Data	No
Drive Mode	Single shooting
Live View Shooting	OFF
Date/Time(UTC)	
Latitude	
Longitude	
Altitude	
Geographic coordinate s...	
Camera Body No.	1170702682
Comment	

Figure 64 Case 3 - EXIF View Using Digital Photo Professional
EXIF of IMG_0799 using the Digital Photo Professional made by Canon. 'Normal' in the *Image Quality* field indicates that the image was taken with the JPEG compression setting of the camera set to Normal.

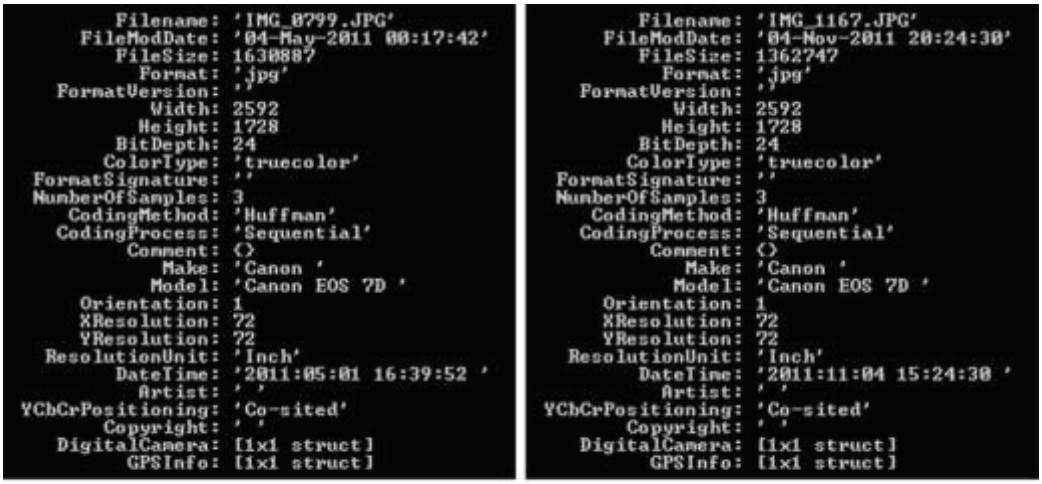


Figure 65 Case 3 - EXIF Comparison
EXIF of the questioned image IMG_0799.JPG (a) and an exemplar (b). There are no forensically relevant inconsistencies between the two.

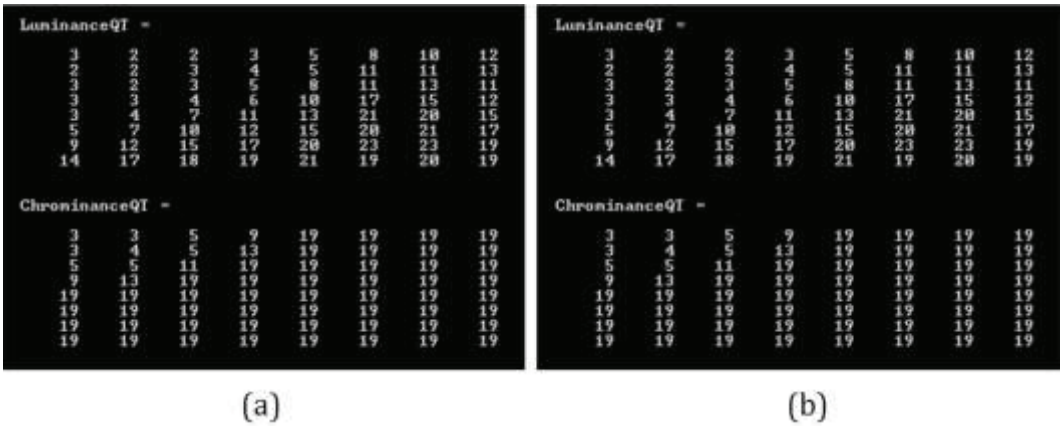


Figure 66 Case 3 - Quantization Table
Quantization tables of the questioned image IMG_0799.JPG (a) and an exemplar (b). There are no differences.

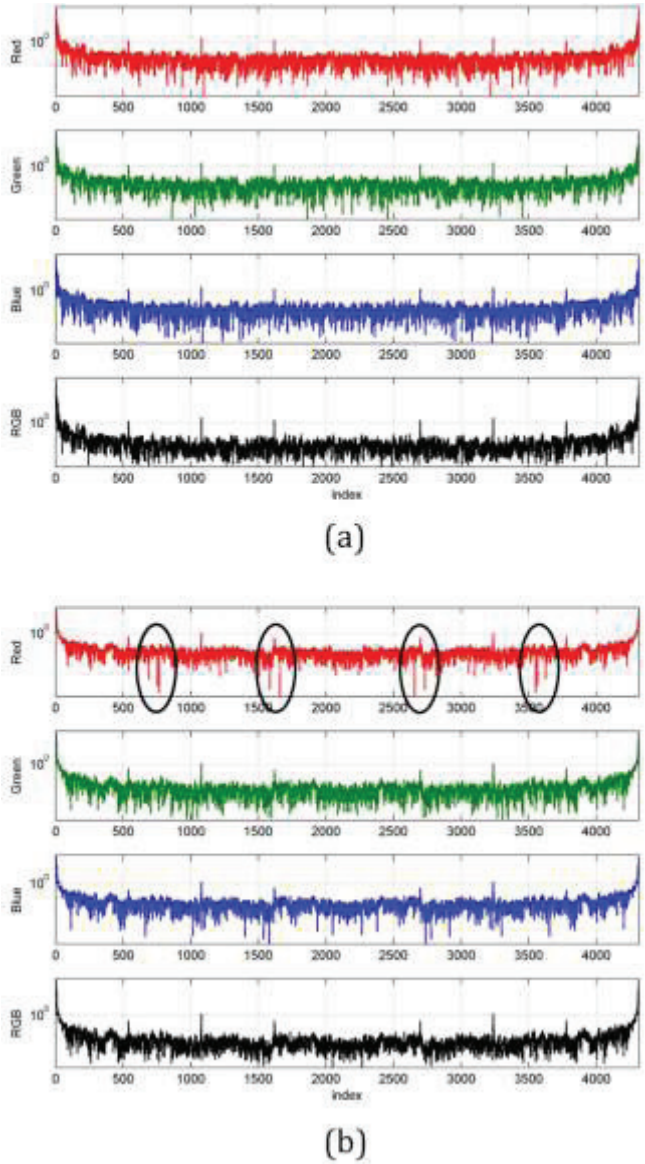


Figure 67 Case 3 - Color Filter Array Analysis

Color filter array graphs of the questioned image (a), and the exemplar (b). While not present in the suspect image, valleys in the graph (indicated by black circles) were present at differing strengths in many of the exemplar images.

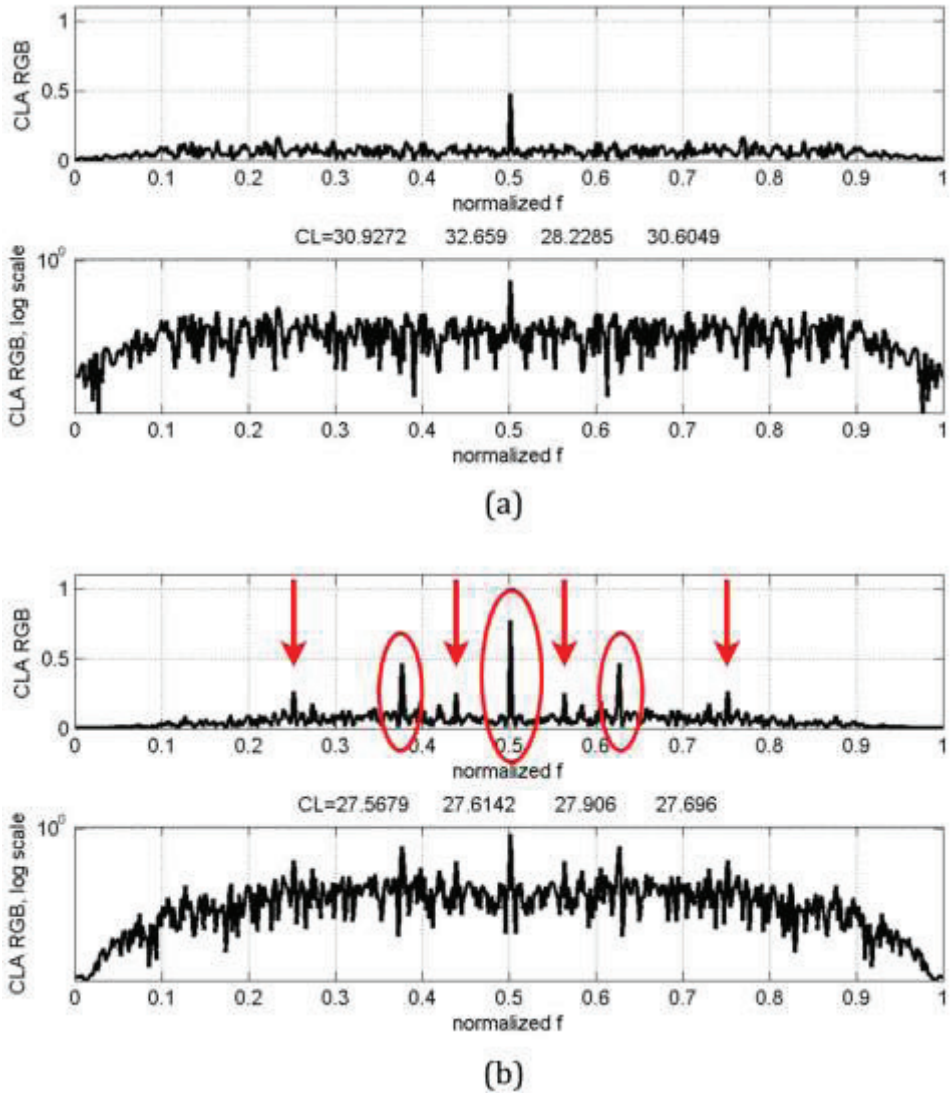


Figure 68 Case 3 - Compression Level Analysis

Compression level analysis of the suspect image (a) and an exemplar (b). For each panel, the top graph represents the plot using a linear scale, while the lower one uses a logarithmic scale. Notice the spikes in the graph of the exemplar that are a product of the normal operation of the camera. These spikes are not present in the suspect image graph.

DCT coefficient analysis of the suspect image compared to the exemplars shows some inconsistencies [Figure 69]. The values of the DC coefficient for the suspect image go above and below the -1000 to 1000 [Figure 69 (a)]. The values of the DCTs of the exemplar images never went above 1000 or below -1000 [Figure 69 (c)]. While the exact cause of this difference is unknown, it still indicates a discrepancy between the suspect image and the exemplars. Conversely, a close up view of the DC components show spikes at the same integer values, indicative that the coefficients are grouped into evenly spaced multiples of the same quantizer step size [Figure 69 (b and d)]. In addition, there is no indication of repeating patterns indicative of second-generation artifacting in the DCT components. However, the discrepancy of the DC values cannot be ignored.

Analyses of the DCT map for the suspect image show no clear indication of manipulation [Figure 70]. There do exist large dark areas in the DCT map, however these are usually artifacts caused by over or under saturation of the pixel at the time of exposure. In this particular case, the white of the boxes are somewhat overexposed while the area underneath the desk is severely underexposed. There are no other areas of interest surrounding the bells in the image.

Error level analysis revealed no indications of JPEG ghosting [Figure 71]. This means that at least, the image was not manipulated by placing a previously, lower recompressed portion of an image, into the suspect one.

To search for defective pixels, 10 images were taken with the lens cap on with the camera settings set to the specifications indicated by the EXIF. The images were then averaged together to remove shot noise. There was a single hot pixel found on the test images that was also found in the same location as the suspect photo [Figure 72]. Although a single defective pixel is not a strong identifier for source camera identification, this is an indication that the suspect camera could have created the image.

A PRNU test was not performed because the strength of the PRNU signature using one image is not reliable with the tools available to the author.

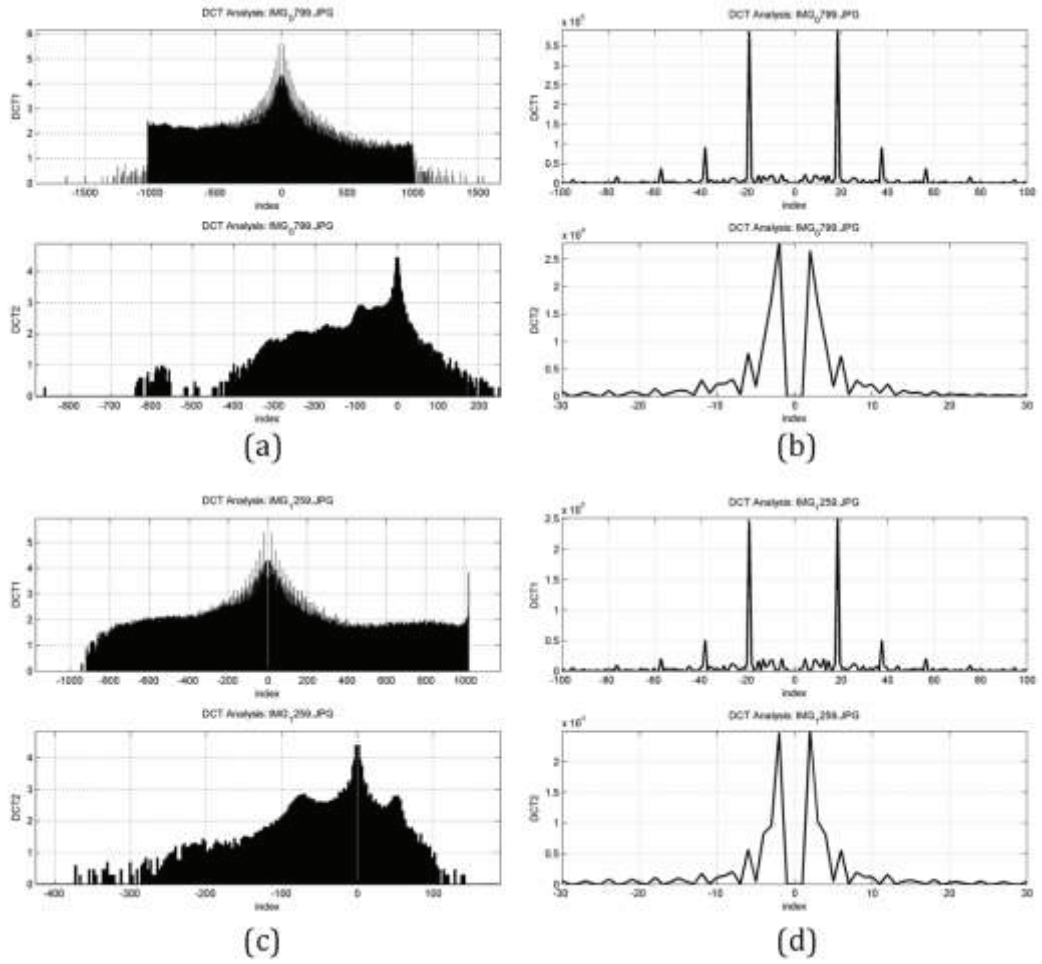


Figure 69 Case 3 - DCT Coefficients

Histogram of the DCT coefficients of the questioned image (a, b) and the exemplar (c, d). The left column graphs show the DCT coefficients for the DC component (top) and the AC coefficient of [1, 2] (bottom). The right column is a magnification of the central spike for each respectively. Note that the values of the DC components for the suspect image (a) extend further than those of the exemplar (b).

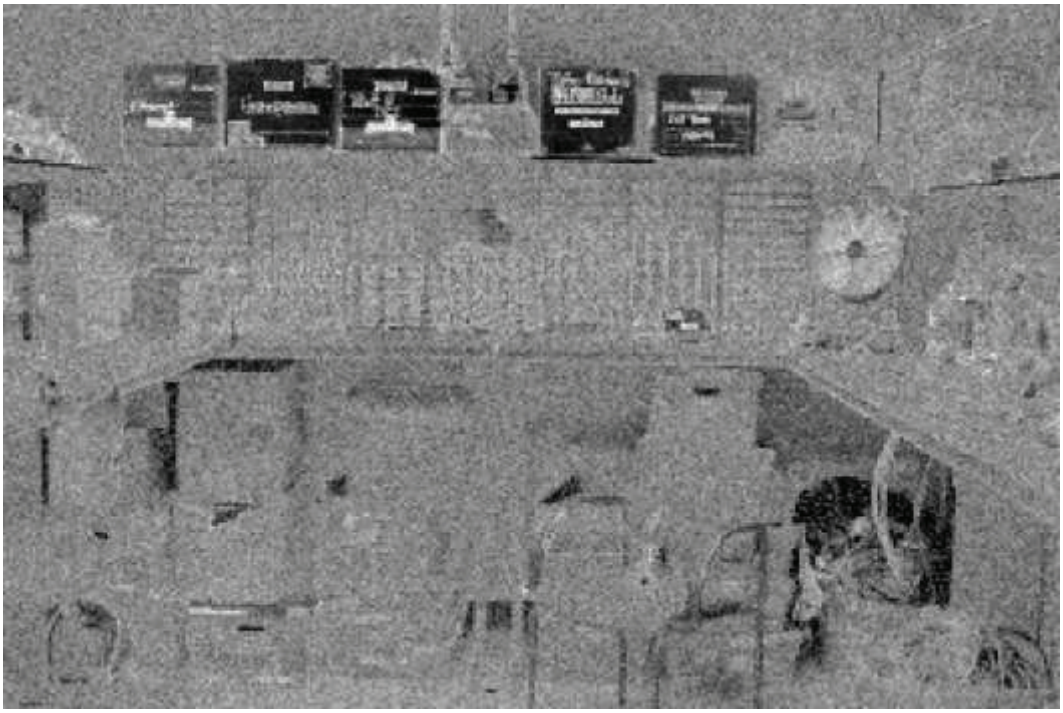


Figure 70 Case 3 - DCT Map

Shown here is the AC component map, which is made from the average of all AC components for each JPEG block. The black portions of the image in the top of the pictures are most likely caused by oversaturation of the pixels of the white boxes. Similarly the large black portion towards the lower right corner attributed to the under saturation of the pixels

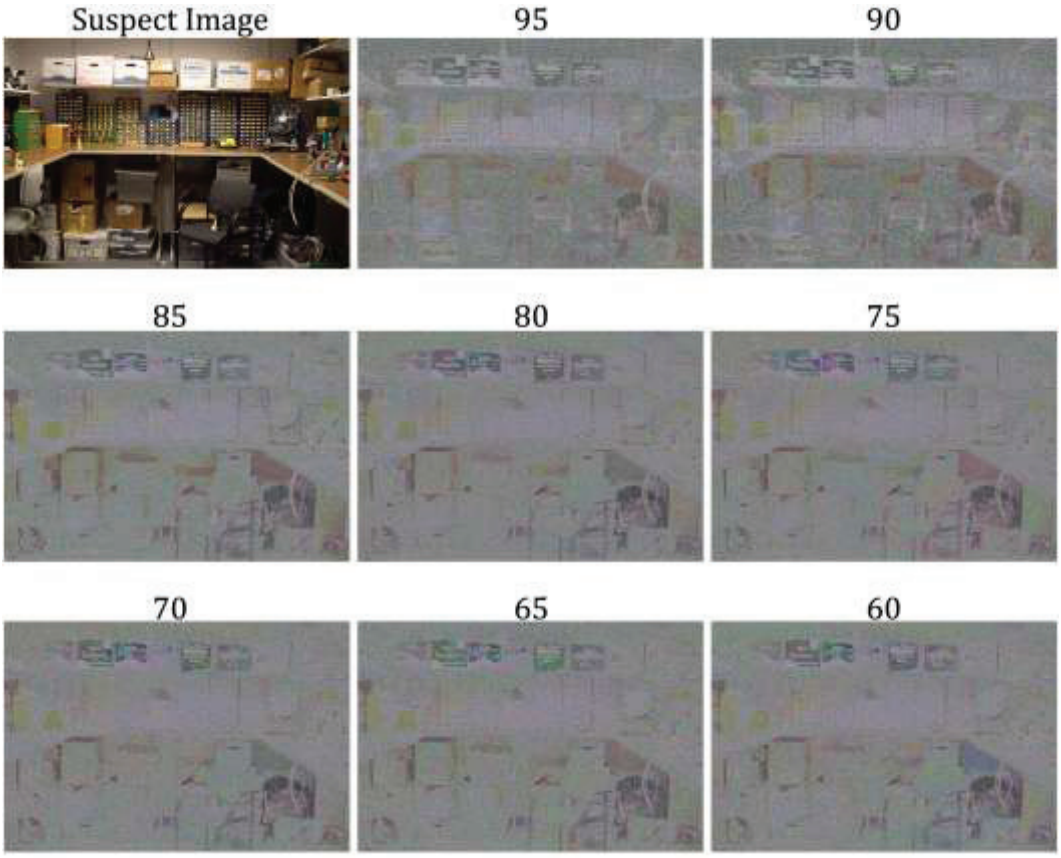


Figure 71 Case 3 – Error Level Analysis
Shown in the upper left corner is the suspect image. The subsequent panels show the difference between the suspect image and recompressed versions at varying compression levels [95-60]. There are no indications of JPEG ghosts in the remaining image levels [55-5] as well.

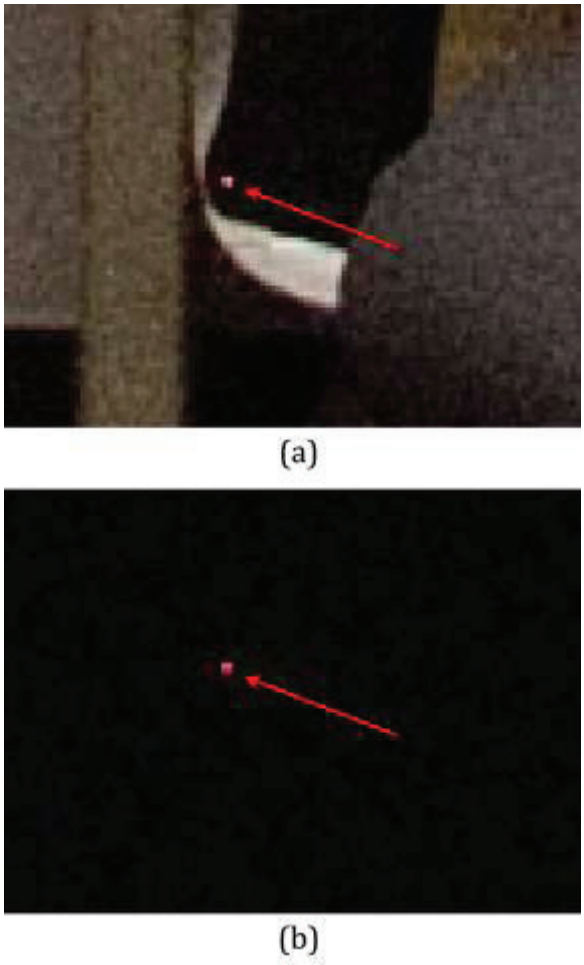


Figure 72 Case 3 - Defective Pixel
A single defective pixel was using test images from the suspect camera with the lens cap on. This defective pixel was also found in the same location on the suspect image.

While clear indications of manipulation do not exist in the *Local Structure* Analyses, there are enough red flags to cast doubt upon the content integrity [Figure 73]. While a majority of the analyses of the *file structure* revealed consistencies with an exemplar images taken from the Canon 7D, the fact that the thumbnail was absent using the Canon image viewer and that the file did not

open in Photoshop, indicate that at least some sort of corruption has occurred within the image file structure. In addition there are inconsistencies in the compression level analysis and the DCT coefficients between the suspect image and exemplars that indicate the characteristics exhibited by the suspect are operating outside of the normal operations for the Canon 7D. Therefore, the results of this investigation reveal that there exist inconsistencies in the suspect image when compared to exemplar image files taken with the camera. While the exact nature of the inconsistencies are unknown, the image file was at minimum not created by the Canon 7D.[4]

File Structure	File Format	✗
	Hex Data	✓
	EXIF Data	✓
Global Structure Analysis	Compression Level Analysis	✗
	Color Filter Array	○
	Quantization Tables	✓
	DCT Coefficients	✗
Local Structure Analysis	Copy and Paste Detection	✓
	Error Level Analysis	✓
	DCT Map	✓
Source Camera Identification	Defective Pixels	✓
	PRNU	N/A

Figure 73 Case 3 - Authentication Table Results

[4] The original image and how it was manipulated is explained in Appendix D

4. Conclusion

The analysis of digital images is not an easy task. Image authentication involves a process of determining what contents of an image are relevant for analysis, and how the observed characteristics relate to existing knowledge. A trained analyst performs this cognitive decision-making through testing hypotheses against known circumstances in order to determine the best explanation for what is observed. Image analysts must ensure that they are competent in their field by demonstrating proficiency under the supervision of a trained analyst. In addition they must maintain that proficiency by continuing to develop their education, and by regular competency testing. This will guarantee that analysts stay up to date on matters relevant in the forensic community by ensuring that they are using validated tools, techniques and procedures used in image analysis. This will help establish that the analyst's conclusions are supported by both experience and expertise.

The nature of digital image files is a mathematical one and not a physical one. A basic understanding of the image creation process is imperative to determine what artifacts and features are relevant to investigate. When an image is taken, a data stream of information is stored onto a storage medium, like a hard drive or memory card. Computer software translates this data stream into a visual image and produces an image onto a monitor. However, there is more information in a digital image file than just the image information. Digital images are a product of mathematics and computer language, both of which operate in a predefined way. Image authentication is about determining if any aspects of this operation have been disturbed. Manipulations can be made to the image, to the digital file, and to the events surrounding the image capture. Therefore, image analysis for authentication purposes can be broken down into four areas: file structure, global structure, local structure, and source identification.

There are many different digital image file formats in use today and each format encodes digital information differently as a computer file. The file format determines how these files will be encoded, stored and retrieved from a storage medium. This computer file not only contains the digital information representative of the image, it also contains a list of the contents of the file, the address of the contents contained within, instructions on how to reassemble the image, and metadata about the image. For a forensic image analyst, the structure of the file, and the information that resides within, can provide important clues in determining authenticity and verification of the events

concerning the image acquisition. In an original image, the equipment used to create the image file produces this information.

File structure analyses investigate the container that the image information resides in. Each image file format is structured differently and organizes the information in a distinctive way. Depending on what type of camera is used and how the settings are configured, image file characteristics such as file type, resolution and naming convention will operate within a defined threshold. File size, for example, will depend on the scene content, compression and resolution of the image. These thresholds can be determined by testing the suspected camera in order to quantify these limits.

Analysis of the hex data is another area of the file structure that should be investigated. Sometimes software programs alter this information and leave traces of their interaction somewhere in the hex data. This information can be searched for known keywords, in order to determine if specific software has interacted with the digital file after creation. Unfortunately, hex data can be easily manipulated with a hex editor, which allows a user to manually change the file information on the binary level. However, a clear understanding of what is being changed is needed so a file does not become corrupted and/or unrecognizable by a computer or software.

Another part of the image structure to search for information is the EXIF, which resides in the header of the digital file. It is used in almost all modern cameras to record equipment model, date and time the image was taken, f-stop, ISO speed, resolution, metering, GPS coordinates, and other information relevant at the time of the image acquisition. For forensic image analysis, the EXIF is an important part of the file structure to inspect because information in the EXIF can be used to validate information about the acquisition of the digital image. Since the EXIF is not standardized, each camera manufacturer can populate the EXIF with custom fields. The EXIF data is a fragile part of the file structure that can easily be corrupted by a device, or software, that interacts with the file. Of course, the EXIF should be analyzed by comparing an EXIF of an original image from the same make and model camera. This way, any particularities in the suspect image EXIF can be compared to that of the exemplar to determine if similarities, or differences, exist between the two. However, a hex editor can easily alter the EXIF.

Although not as reliable, MAC time stamps can be used to determine some information about a digital file. However, this information should be approached with caution, as date and times are only relevant to the computer the image file resides on. The MAC times are not a part of the image file, but are a record kept by the operating system of the computer, and are influenced by the time settings of the system.

Global image structure analyses consider the information of the data that represent the actual image content. Analyses of the image data as a whole, helps to determine if the overall structure of the image deviates from the normal operations of the acquisition device. Analyses in the global structure should be compared to unaltered images from the suspected imaging device to determine if any similarities or differences exist. These analyses are relevant to same make and model cameras, and cannot be used to determine the identification of a particular device. However, similar make and model cameras are programmed to process light information the same. This includes how the camera treats information from the CFA, through in camera processing, such as white balance and gamma correction, to JPEG compression (if compressed), until it is finally saved as an image file onto a storage medium. Compression plays an important part of the global structure of an image and a thorough understanding of image file formats, especially the JPEG compression standard, is important information to know.

JPEG compression, resolution resizing, and color filter array demosaicing create interpolation artifacts, which can be caused by a camera's internal processor or by image manipulation software. In order to determine which one, exemplars from the same make and model camera must be taken and compared to the suspect image. Computing the second derivative on an uncompressed image matrix will look significantly different if the image has been reprocessed. In the case of JPEG compression the graph will have significantly more spikes than an uncompressed image, in addition to more noise. Additional recompression will start to alter the second derivative even further. Caution should be taken with this approach because while higher compression settings will cause obvious double compression artifacts, an image recompressed using better compression quality may obfuscate those traces. This is the same for analysis of color filter array as well. In addition, the use of the expectation/maximization algorithm is an excellent tool for identifying interpolation artifacts.

When an image is compressed using the JPEG compression scheme, the process adds additional features for analysis. The quantization table can be compared to those of a camera to determine if similarities exist. In addition, some software programs, like Adobe Photoshop, have unique tables that can be used for identification. Investigation of the DCT coefficients can also identify traces of manipulation. When an uncompressed image is saved using JPEG compression, the DCT coefficient values are grouped into value bins based on the quantization table quantizer step. These values are evenly distributed when sorted in a histogram and are modeled by a Laplacian distribution curve. When the image is double compressed, the values of the DCT coefficients bins are no longer evenly distributed and exhibit artifacts in the form of periodic patterns. This effect looks different for images compressed with a higher quality or lower quality than the original compression setting.

Investigation of the file structure and global structure analyses can provide good clues to the authenticity of the image file. However, inconsistencies between a suspect image and exemplars cannot identify if the image content has been altered. Indications may imply that the image could have been altered from the time the camera took the image to the time the analysts examined it. Malicious manipulations are the application of techniques in an attempt to create an illusion, or deception, of the events in an image. Investigation of the image on the pixel level helps to determine if the local image structure has been altered. Thus, analyses can be performed to determine if scene content has been altered and where.

Copy and paste detection is one of the most common types of techniques used to manipulate image content. By using this technique, the perception of events in an image can be altered by removing a person or object, or inserting one. In doing so, underlying characteristics of an image, like those in the DCT coefficients and PRNU will be different from an unprocessed image. Copy and pasted pixels can be easily detected if the image has not been processed by additive noise or recompression. If recompression is used, JPEG ghosting or DCT mapping can identify differing quantization error in the DCT coefficients. These artifacts are easily discernable and are clear indications of manipulation. Inspection of color aberrations can also help determine traces of image alteration.

In addition to the identification of image alteration, other techniques are available to the analyst for source camera identification. The most robust of these techniques utilize imperfections that are introduced by a digital camera's sensor. Artifacts caused by the sensor will be imprinted on every image taken with the camera. Theses imperfections can be used to uniquely identify if an image originated from that sensor. The photo-response non-uniformity of an image sensor is cause by slight variations in a pixels ability to convert photonic energy to electrical. The result of these imperfections causes each pixel of the sensor to exhibit varying degrees of sensitivity to light. The PRNU is a relatively weak signal and is masked by image content. Therefore it is necessary to have multiple images available to extract the PRNU signature, which is its main drawback. However, research is being conducted to overcome this issue. Once extracted, the PRNU has been shown to provide very accurate identification of the imaging device that created the picture. PRNU has also been shown to be useful in identifying areas in an image that have been altered.

Another identifier of the sensor is pixel defects, which will manifest as white or dark pixels that are consistently in the image from shot to shot. These errors are caused by malfunctioning pixels in the sensor. These defects are hard to identify because they are small and can be masked by scene content. Identification is usually done manually by averaging multiple images and searching the image matrix manually. Once identified however, the pixels can be easily detected from one image to the next.

Throughout this thesis, techniques and concepts have been explained to help guide analysts through a cognitive approach to authenticating digital images. While the material is not an exhaustive list of possible analyses, the general model that should be taken when tasked with image authentication should be a well-rounded one. A digital image is comprised of a finite set of numbers, arranged by a series of mathematical algorithms to create a digital image file. If one part in the chain is altered, it will most likely affect other aspects of this predictability. While certain manipulation techniques can elude one or more analyses, it may be extremely hard to elude them all. The goal of this thesis is to help bridge the gap of science and application by presenting multiple analytical approaches to digital image authentication, and to help explain artifacts and characteristics useful when analyzing digital images. It is important to be able to distinguish between artifacts that are consistent with manipulation and those that are created under the normal operating process of the imaging device. Each analysis cannot be used on its own to make a decision about content integrity. It is imperative that all processes be used in conjunction

to provide the most accurate and thorough assessment of image authenticity. There is no doubt that more sophisticated manipulation techniques will be discovered in the future. However, manipulation in one area of an image can cause disruptions in other areas. Investigating a digital image from all possible angles will give the best hope of discovering traces left behind when an image is manipulated.

This guide is not intended to supplant the existing materials that are referenced in each section. Techniques are explained on a very basic level only to provide an understanding of the concepts of the authentication method. The reader is encouraged to read all of the existing literature referenced in this work, which will provide a more in depth understanding of the mathematics and principles involved for each type of analysis. As the development of analytical tools continues to grow, the hope is that it will get increasingly more difficult to create a successful image forgery.

APPENDIX A

HEX DATA SEARCH TERMS

The following is a known list of terms that are embedded in the hex data by various image processing software programs. This list is not all-inclusive and capitalizations may be used by some image processing programs. It is best to use a caps independent search.

adobe
aperture
ashampoo
bibble
borderfx
capture
coachware
commander
corel
copiks
digikam
digital
gimp
idimager
imagenomic
imageready
kipi
microsoft
noiseware
paint
paintshop
photomapper
photo
photoscape
photoshop
photowatermark
picasa
picnik
pro
professional
quicktime
shop
watermark
windows

APPENDIX B

PRNU VALIDATION TESTING

A PRNU validation test was performed in conjunction with the Target Forensic Services Laboratory (TFSL). Testing of PRNU has been successfully proven to be an accurate identifying mark for source camera identification [56][60-65]. Testing has been confined mainly using different make and model cameras, and not much research has dealt with the strength of the PRNU signal when using same make and model cameras. Target Corporation provided access to multiple same make and model cameras for a small scale PRNU test using 80 cameras. The make and model of the cameras used for this testing are presented in Table 1.

Camera Model	Quantity
Axis 216FD	38
Axis 216MFD	1
Axis M3203	2
Axis M3301	1
Axis M3204	2
Axis 211	36

Table 1 Camera Models Used in PRNU Testing

TEST VIDEO ACQUISITION

All cameras used in this test were IP cameras manufactured by AXIS Communications. These cameras operate by use of a power injector that supplies power to the camera via a CAT5 ethernet cable. In this manner, no external analog to video converted was needed to convert the signal, as all processing is done inside the camera. Video was captured directly from the camera using the AXIS Camera Management Utility without further processing caused by the software program. Configuration of the device was accomplished through this program as well.

Although the capabilities for each camera varied, settings for each model were made as uniform as possible. The resolution for each camera was set to *640 x 480*, video format was set to *motion JPEG* at 30 frames per second, and compression settings were set the least amount of compression possible. Auto gain control was enable, white balance was set to *indoor fixed*, and priority for the recorded video was set to image quality over frame rate. To maintain configuration uniformity between cameras of the same model, a template of all settings in the camera was created and loaded into each camera before creating test videos.

Each camera had a manual zoom and focus ring that was utilized for testing. Test videos were taken using a light box to provide uniform illumination of the camera sensor. In addition, the focus for each camera was adjusted to blur all image content. Approximately 35 seconds of video was recorded at 30 fps to acquire at least 1000 individual frames. After each video was recorded, the video file was promptly renamed with the serial number of the camera, along with the make and model.

PRNU PREPARATION AND EXTRACTION

The video file created by capturing the video using the AXIS Camera Management utility was the .asf format. To extract individual frames from each video, the software program Forevid was used. This program has the ability to convert the video file into a sequence of still images. The file format chosen from this program was uncompressed TIFF image files. The majority of cameras had at least 1000 frames extracted from each video.

Once all frames had been extracted as TIFF image files, MATLAB was used to create averaged image files using the recommended 50 images [56]. The amount of averaged images per camera was 20 or more for each camera, with the exception of a few. In addition, the MATLAB script was programmed to use non-sequential images from the extracted frames, thus reducing the possibility of corrupting the PRNU extraction by averaging images containing similar characteristics due to slow camera motion or accidental pauses when the video was recording.

The PRNU was extracted from each of the average image files using MATLAB by applying a Gaussian blur to each averaged image and subtracting it from the original. Additional processing was performed for each extraction by subtracting the average from each row and column as recommended by Fridrich [55]. Each PRNU signature was saved as an uncompressed TIFF file and placed in the database. For testing purposes, one PRNU estimation file for each camera was removed from database, and placed in a separate location. These images will be referred to as the "unknown" set. In all, there were 1586 images in the blurred database and 80 "unknown" images.

PRNU TESTING RESULTS

Using MATLAB each of the images from the *unknown* set was compared against all images in database using the correlation coefficient equation (eq. 6). Once the values were computed, the program identified the three cameras that had the highest correlation scores, and performed an intra-variability test on each of the three cameras. In this way, the correlation score of the *unknown* image when correlated to each of the three selected cameras could be compared to the correlation scores of images known to have come from each camera [Figure 74]. The results of the test indicated a 100% accuracy rate for identification even among cameras of the same make and model. However, the results of this experiment were gathered from test images taken from a controlled environment in optimal conditions, i.e. blurred video taken of an evenly illuminated surface. To validate these findings, future testing will be done using video more closely imitating real life conditions, i.e. focused video of high contrast subject matter.

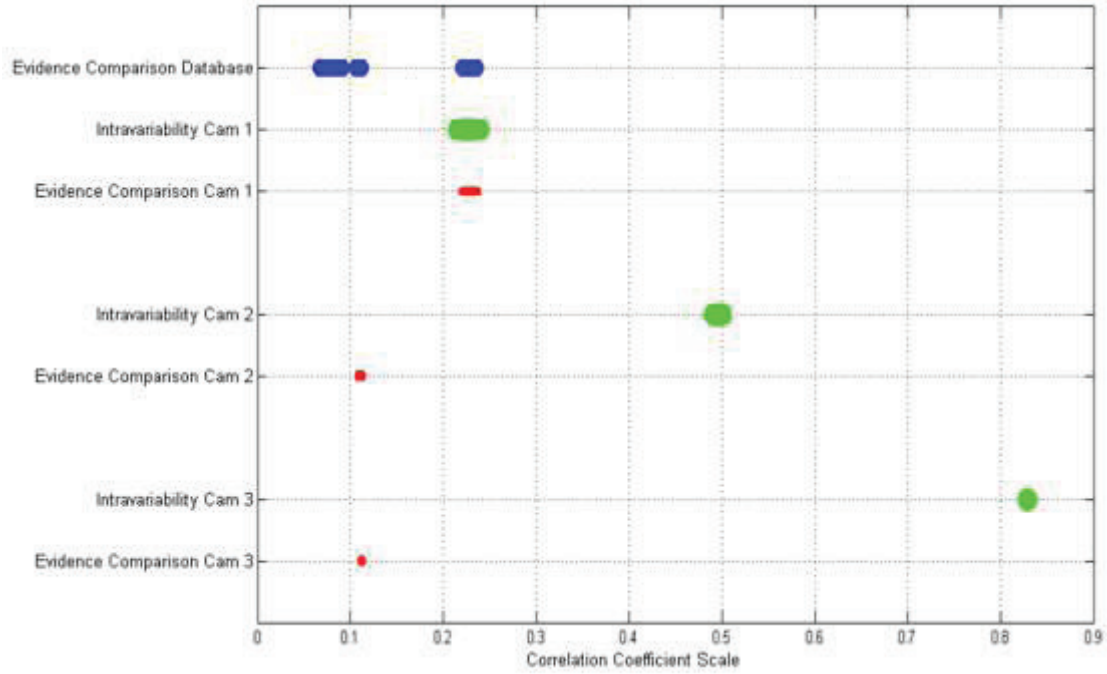


Figure 74 PRNU Validation Test Correlation Results

Results from a single source image identification test for PRNU using the correlation method. The evidence PRNU template is compared to all images in the database (*Evidence Comparison Database* indicated in blue). An intra-variability test is performed using images from the three cameras that have the highest correlation scores when compared to the evidence image (denoted in green). This indicates the intra-variability of images originating from the same camera. The correlation between the evidence image and the images from each three cameras are performed (indicated in red). As can be seen above, the correlation scores of the evidence image when compared to images from Cam 1 fall in the same numerical range. The same test for Cam 2 and Cam 3, show the correlation values to be very different. It can easily be seen that the image shows strong indications that it originated from Camera 1.

APPENDIX C

CASE #2 METHOD OF MANUPULATION

A picture of the original image used in case #2 is shown in Figure 75.[5] The original image in this case was manipulated with the Photoshop CS5 tool 'Content Aware.' Once the woman and dog had been removed, the image was then resaved as a JPEG file using Photoshop with a compression setting of 12. Using a hex editor, the word 'Photoshop' was removed from the header of the file, as well as the date and time the file was altered. Because nothing was added in their place, these fields were left empty in the EXIF [Figure 49].

Figure 75 Case 2 - Original Un-Manipulated Image
This is what the original image used in case #2 looked like before being altered.

[5] Picture provided courtesy of Jeff Smith.

APPENDIX D

CASE #3 METHOD OF MANUPULATION

The image used in case 3 [Figure 62] was created using a composite of 5 different images. The manipulation was performed in several steps. The first step involved setting up the camera on a tripod in a room with no windows. The photos were taken using a remote switch to avoid any slight movement of the camera caused by directly pressing the shutter on the camera for each image. The hand bell was moved around the room and different shots were taken. The images were taken with the resolution settings of the camera set to "small" (2592 x 1728) using the S-RAW file format. The 14-bit raw images were then opened in Adobe Photoshop and converted to 16-bit TIFF files. One image was chosen to be the background image, and the hand bells from the other images were copy and pasted to this 'background' image [Figure 76]. To avoid detection of the compositing, cutting around the hand bell was limited to borders of objects like the table or box edges. No other Photoshop tools were used in the manipulation. The file was then saved as an 8-bit TIFF image and exported for processing.

Using the Canon Digital Photo Professional software, the TIFF image was saved as a JPEG image in the hopes that the Canon software would contain the same quantization tables as the Canon 7D, however, this was not the case. Digital Photo Professional has only 10 different JPEG compression settings, (0 thru 10). The QT with the closest values to those of an original image from the Canon 7D was with a setting of 3 [Figure 77]. The manipulated image was saved as a JPEG image using this setting. Using a hex editor, the quantization table of the manipulated image was replaced with those of an original JPEG image from the Canon 7D for the 'Normal' compression setting. To remove the remaining traces of manipulation, the hex editor was also used to remove all traces of Photoshop and Digital Photo Professional that had been embedded in the hex by these programs. Using EXIFER, the EXIF of the manipulated images was replaced with the EXIF of an original JPEG image from the Canon 7D that had been taken with the 2592 x 1728 resolution, and had been created with the compression setting set to 'Normal'. It is unclear which step in the hex editing process corrupted the JPEG thumbnail and created the "unknown or invalid" JPEG marker type.

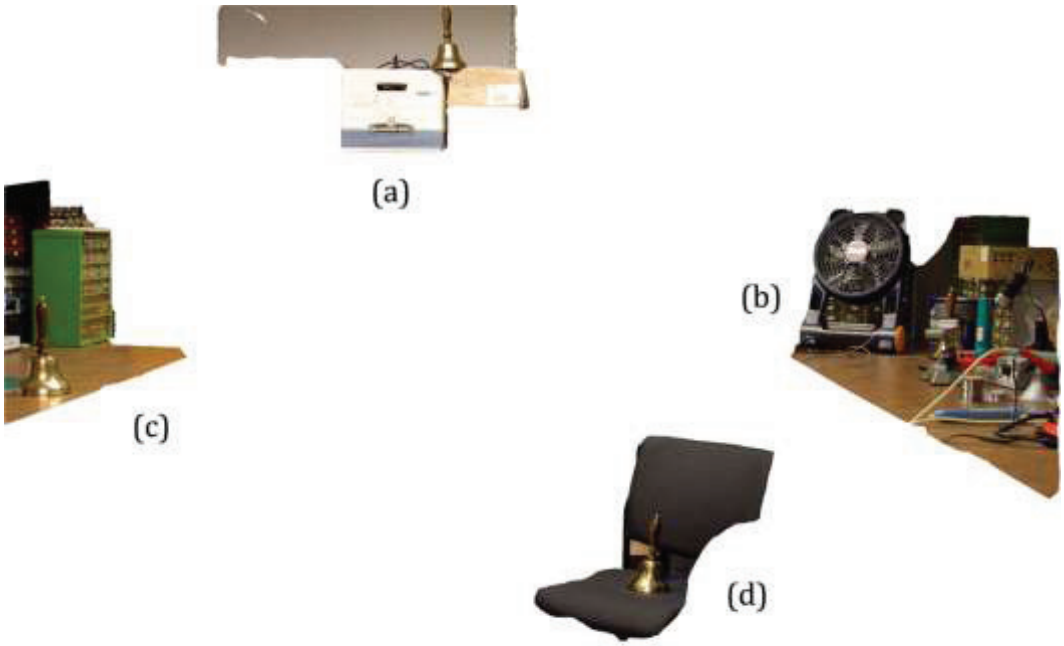


Figure 76 Case 3 - Composited Areas Used
This figure reveals the areas used to manipulate the Case 3 suspect image [Figure 62]. Areas from 4 other images (a, b, c, d) were copy and pasted onto a fifth image. The images above were pasted as seen above with no additional Photoshop tools used to hide the composited additions.

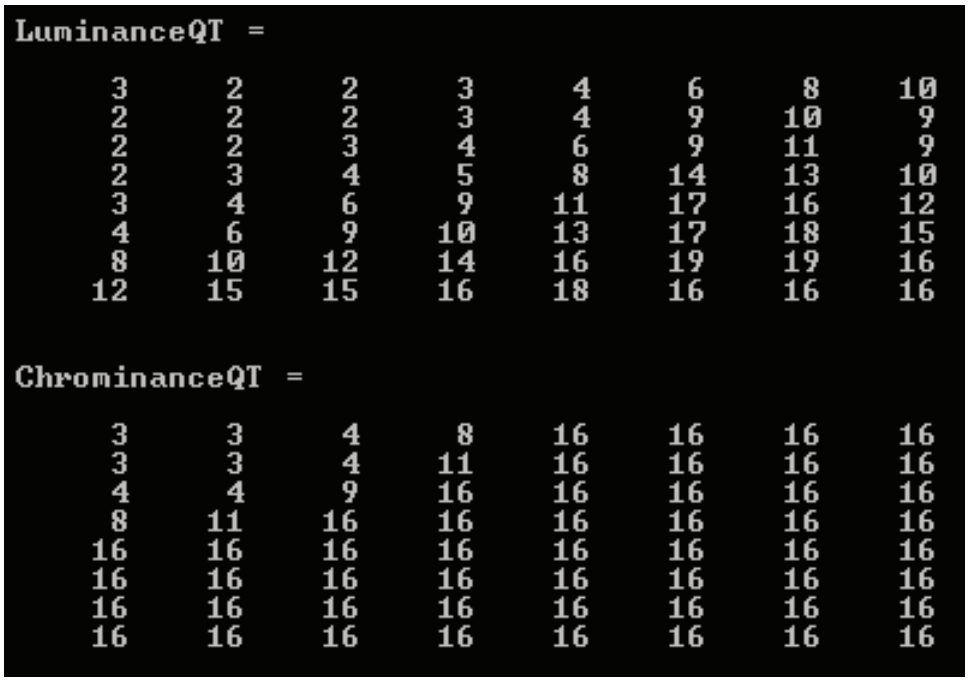


Figure 77 Case 3 - QT Used by Digital Photo Professional
This quantization table was the original QT of the Case 3 manipulated image when it was saved as a JPEG image using Digital Photo Professional.

Bibliography

[1] Scientific Working Groups on Digital Evidence and Imaging Technology. (2009, January 16). Section 16: Best practices for forensic photographic comparison, ver. 1.0. Retrieved October 2, 2011, from the International Association for Identification website: http://www.theiai.org/guidelines/swgit/guidelines/section_16_v1-0.pdf

[2] Rogers, M., & Smith, J. M. (2010). Computer Forensics for the Forensic Audio Professional. *39th International Conference: Audio Forensics; Practices and Challenges* (pp. T-2). Hillrod, Denmark: AES.

[3] Chisum, W. J., & Turvey, B. E. (2000). Evidence dynamics: Locard's exchange principle & crime reconstruction. *Journal of Behavioral Profiling, 1* (1).

[4] Casey, E. (2010). *Handbook of digital forensics and investigations.* Burlington, MA: Elsevier Academic Press.

[5] Scientific Working Groups on Digital Evidence and Imaging Technology. (2011, January 14). SWGDE and SWGIT digital & multimedia evidence glossary. Retrieved October 2, 2011, from the Scientific Working Group on Digital Evidence website: http://www.swgde.org/documents/current-documents/

[6] Ryan, D. J., & Shpantzer, G. (2003). *Legal aspects of digital forensics.* Retrieved October 12, 2011, from http://euro.ecom.cmu.edu/program/law/08-732/Evidence/RyanShpantzer.pdf

[7] Scientific Working Group on Digital Evidence. (2006, July). Best practices for computer forensics, ver. 2.1. Retrieved October 2, 2011 from the Scientific Working Group on Digital Evidence website: http://www.swgde.org/documents/current-documents/

[8] Fed. R. Evid. 1003.

[9] Tuthill, H., Graeme, G. (2002) *Individualization: Principles and procedures in criminalistics, 2nd edition*. Salem, Oregon: Lightning Powder Company.

[10] Siegal, J. A., Saukko, P. J., & Knupfer, G. C. (2000). *Encyclopedia of forensic sciences.* San Diego: Academic Press.

[11] American Academy of Forensic Sciences. (2011). American academy of forensic sciences bylaws. Retrieved October 19, 2011, from the AAFS website: http://www.aafs.org/aafs-bylaws#Art2

[12] Baatz, Willfried (1997). *Photography: An illustrated historical overview*. New York: Barron's.

[13] Rudinjanto (2003). *Digital camera starts to gain ground in local market.* Retrieved November 21, 2011, from the Jakarta Post website: http://www.thejakartapost.com/news/2003/12/14/digital-camera-starts-gain-ground-local-market.html

[14] Hannemyr, G. (2006). *IR Photography*. Retrieved November 17, 2011, from the DPanswers website: http://dpanswers.com/content/irphoto.php

[15] Fed. R. Evid. 901.

[16] Lester, P. (1991). *Photojournalism: An ethical approach.* Hillsdale, New Jersey: Lawrence Erlbaum Associates.

[17] MacDougall, C. (1971). *News pictures fit to print … or are they?* Stillwater, Oklahoma: Journalistic Services.

[18] Jaubert, A. (1986). *Making people disappear.* Washington, DC: Pergamon-Brassey's International Defense Publishers.

[19] Connor, K., & Farid, H. (2011). *Photo tampering throughout history*. Retrieved November 17, 2011, from the Four and Six website: http://www.fourandsix.com/photo-tampering-history/

[20] Brink, B. (1988, June). Question of ethics: Where does honesty in photojournalism begin? *News Photographer*, pp. 21-22, 23-33.

[21] Pingdom. (2010, June 18). Exploring the software behind Facebook, the world's largest site. Retrieved October 2, 2011, from the Pingdom website: http://royal.pingdom.com/2010/06/18/the-software-behind-facebook/

[22] Scientific Working Groups on Digital Evidence and Imaging Technology. (2007, January 1). Section 12: Best practices for forensic image analysis, ver. 1.6. Retrieved October 2, 2011, from the International Association for Identification website: http://www.theiai.org/guidelines/swgit/guidelines/section_12_v1-6.pdf

[23] Johnson, M. K., & Farid, H. (2007). Exposing digital forgeries in complex lighting environments. *2* (3), 450-461.

[24] Johnson, M. K., & Farid, H. (2005). Exposing digital forgeries by detecting inconsistencies in lighting. *Proceedings of the 7th Workshop on Multimedia and Security* (pp. 1-10). New York: ACM.

[25] Conotter, V., Boato, G., & Farid, H. (2010). Detecting photo manipulation on signs and billboards. *IEEE Internationl Conference on Image Processing*, (pp. 1741-1744). Hong Kong.

[26] Scientific Working Groups on Digital Evidence and Imaging Technology. (2007, June 4). Section 14: Best practices for image authentication, ver. 1.0. Retrieved October 2, 2011, from the International Association for Identification website: http://www.theiai.org/guidelines/swgit/guidelines/section_14_v1-0.pdf

[27] Audio Engineering Society. (2007). AES27-1996(2007): AES recommended practice for forensic purposes - managing recorded audio materials intended for examination.

[28] Randi, J. (Lecturer). (1992). *James randi lecture at caltech – can't prove a negative*. [Web Video]. Retrieved from http://www.youtube.com/watch?v=qWJTUAezxAI.

[29] Information Technology Industry Council. (1986). *Coded character sets - 7-bit american national standard code for information interchange.* New York: American National Standards Institute.

[30] Kee, E., Johnson, M. K., & Farid, H. (2011). Digital image authentication from JPEG headers. *IEEE Transactions on Information Forensics and Security*, 6 (3), 1066-1075.

[31] Schmidt, F. (2002). *Exifer for Windows.* Retrieved October 2, 2011, from the Exifer website: http://www.exifer.friedemann.info/

[32] Wallace, G. K. (1991). The JPEG still picture compression standard. *IEEE Transactions on Consumer Electronics*, *34* (4), 30-44.

[33] Ahmed, N., Natarajan, T., Rao, K. R. (1974). Discrete cosine transform. *IEEE Transactions on Computers,* C-23 (1), 90-93.

[34] Campbell, F. W., Robson, J. G. (1968). Application of fourier analysis to the visibility of gratings. *The Journal of Physiology,* 197, 551-566.

[35] Mahdian, B., & Saic, S. (2009, 2). Detecting double compressed JPEG images. *IET Seminar Digests*, pp. 12-17.

[36] Gallagher, A. C. (2005). Detection of linear and cubic interpolation in JPEG compressed images. *The 2nd Canadian Conference on Computer and Robot Vision*, (pp. 65-72).

[37] Popescu, A. C., & Farid, H. (2005). Exposing digital forgeries by detecting traces of re-sampling. *IEEE Transactions on Signal Processing*, *53* (2), 758-767.

[38] Foveon, Inc. (2011). *X3 Technology*. Retrieved October 14, 2011, from the Foveon website: http://www.foveon.com/article.php?a=69

[39] Popescu, A. C., & Farid, H. (2005). Exposing digital forgeries in color filter array interpolated images. *IEEE Transactions on Signal Processing*, *53* (10), 3948-3959.

[40] Bayram, S., Sencar, H. T., Memon, N., & Avcibas, I. (2005). Source camera identification based on CFA interpolation. *IEEE International Conference on Image Processing*, *3*, pp. 69-72.

[41] Bayram, S., Sencar, H. T., & Memon, N. (2008). Classification of digital camera-models based on demosaicing artifacts. *Digital Investigation*, *5* (1-2), 49-59.

[42] Farid, H. (2006). Digital image ballistics from JPEG quantization. *Technical Report TR2006-583*. Dartmouth College.

[43] Farid, H. (2008). Digital image ballistics from JPEG quantization: A follow-up study. *Technical Report TR2008-638*. Dartmouth College.

[44] Kornblum, J. D. (2008). Using JPEG quantization tables to identify imagery processed by software. *Digital Investigations*, *5*, S21-S25.

[45] Luo, W., Haung, J., & Qiu, G. (2010). JPEG error level analysis and its applications to digital image forensics. *IEEE Transactions on Information Forensics and Security*, *5* (3), 480-491.

[46] Vose Software. (2007). *Laplace Distribution.* Retrieved September 25, 2011, from the Vose Software website: http://www.vosesoftware.com/ModelRiskHelp/index.htm#Distributions/Continuous_distributions/Laplace_distribution.htm

[47] He, J., Lin, Z., Wang, L., & Tang, X. (2006). Detecting doctored JPEG images via DCT coefficient analysis. *In Proceedings of European Conference on Computer Vision*, (pp. 423-435).

[48] Popescu, A. C., & Farid, H. (2004). Statistical tools for digital forensics. *In 6th International Workshop on Information Hiding*, (pp. 128-147).

[49] Stamm, M. C., Tjoa, S. K., Lin, W. S., & Liu, K. J. (2010). Anti-forensics of JPEG compression. *IEEE International Conference on Acoustics Speech and Signal Processing*, (pp. 1694-1697). Dallas.

[50] Lai, S., & Bohme, R. (2011). Countering counter-forensics: The case of JPEG compression. *13th International Conference on Information Hiding. 6958*, pp. 285-298. Prague: Springer.

[51] Fridrich, J., Soukal, D., & Lukas, J. (2003). Detection of copy-move forgery in digital images. *In Proceedings of Digital Forensic Research Workshop.*

[52] Popescu, A. C., & Farid, H. (2006). Exposing digital forgeries by detecting duplicated image regions. *Office*, (2000), 1-11. Citeseer. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.136.2374&rep=rep1&type=pdf

[53] Johnson, M. K., & Farid, H. (2006). Exposing digital forgeries through chromatic aberration. *Proceedings of the 8th Workshop on Multimedia and Security, (*pp. 48-55). Geneva.

[54] Chen, M., Fridrich, J., & Goljan, M. (2007). Digital imaging sensor identification (further study). *Proc. SPIE, Electronic Imaging, Security, Steganography and Watermarking of Multimedia Contents IX*, *6505*, pp. 0P-0Q.

[55] Fridrich, J. (2009, March). Digital image forensics. *IEEE Signal Processing Magazine*, *26* (2), pp. 26-37.

[56] Khanna, N., Mikkilineni, A. K., & Delp, E. J. (2009, January). Forensic camera classification: Verification of sensor pattern noise approach. *Forensic Science Communications*, *11* (1).

[57] Reininger, R. C., & Gibson, J. D. (1983). Distributions of the two-dimensional DCT coefficients for images. *IEEE Transactions on Communications*, *31* (6), 835-839.

[58] Weiqi, L., Zhenhua, Q., Feng, P., & Jiwu, H. (2007). A survey of passive technology for digital image forensics. *Frontiers of Computer Science*, (pp. 166-179).

[59] Farid, H. (2009). Exposing digital forgeries from JPEG ghosts. *IEEE Transactions on Information Forensics and Security*, *4* (1), 154-160.

[60] Alles, E. J., Geradts, Z. J., & Veenman, C. J. (2009). Source camera identification for heavily compressed low resolution still images. *Journal of Forensic Sciences, 54* (3), 628-638.

[61] Chen, M., Fridrich, J., Lukas, J., & Goljan, M. (2007). Imaging sensor noise as digital x-ray for revealing forgeries. *Proceedings of the 9th International Conference on Information Hiding*, (pp. 342-358).

[62] Chierchia, G., Parrilli, S., Poggi, G., Sansone, C., & Verdoliva, L. (2010). On the influence of denoising in PRNU based forgery detection. *Proceedings of the 2nd ACM Workshop on Multimedia in Forensics, Security and Intelligence* (pp. 77-82). ACM.

[63] Goljan, N., Fridrich, J., & Filler, T. (2009). Large scale test of sensor fingerprint camera identification. *Proc. SPIE 7254*, *7254*, pp. OI 1-OI 12. San Jose.

[64] Lukas, J., Fridrich, J., & Goljan, M. (2006). Digital camera identification from sensor pattern noise. *IEEE Transactions on Information Forensics and Security*, *1* (2), 205-215.

[65] Jenkins, N. (2009). *Digital camera identification.*

[66] Fang, Y., Dirik, A. E., Sun, Z., & Memon, N. (2009). Source class identification for DSLR and compact cameras. *IEEE International Workshop on Multimedia Signal Processing*, (pp. 1-5).

[67] Geradts, Z. J., Bijhold, J., Kieft, M., Kurosawa, K., Kuroki, K., & Saitoh, N. (2001). Methods for identification of images acquired with digital cameras. *Proc. SPIE*, *4232*, pp. 505-512.

[68] Suprock Technologies. (2011). *Signal processing & data analysis*. Retrieved November 14, 2011, from the Suprock Technologies website: http://suprocktech.com/solutions/signal-processing-data-analysis.aspx

[69] eXo maKina. (2011). *Photo-anterprétation avancée*. Retrieved November 14, 2011, from the eXo maKina website: http://www.exomakina.com/eXo_maKina/Tungstene.html

[70] Petre, A., Grigoras, C. (2010). *Inregistrarile Audio si Audio-Video. Ed.* Bucharest, Romania: C.H. Beck.

[71] Adobe. (2011). *Adobe photoshop CS5.* Retrieved November 14, 2011, from the Adobe website: http://www.adobe.com/products/photoshop.html

[72] Corel. (2011). *Corel paintshop pro X4 ultimate.* Retrieved November 14, 2011, from the Corel website: http://www.corel.com/corel/product/index.jsp?pid=prod4220093&cid=catalog20038&segid=5900044&storeKey=us&languageCode=en

[73] GIMP. (2011). *GIMP – The GNU image manipulation program.* Retrieved November 14, 2011, from the GIMP website: http://www.gimp.org/

[74] JPEGsnoop. (2011). *ImpulseAdventure – JPEGsnoop – JPEG decoding utility.* Retrieved from the Impulse Adventure website: http://www.impulseadventure.com/photo/jpeg-snoop.html

[75] MATLAB. (2011). *MATLAB – The language of technical computing.* Retrieved from the MathWorks website: http://www.mathworks.com/products/matlab/index.html

[76] Battiato, S., & Messina, G. (2009). Digital forgery estimation into DCT domain - a critical analysis. *ACM Multimedia 2009 Workshop Multimedia in Forensics*, (pp. 37-42).

[77] Farid, H. (2009). A survey of image forgery detection. *IEEE Signal Processing Magazine, 26* (2), 16-25.

[78] Huang, Y., Lu, W., Sun, W., & Long, D. (2011, March 20). Improved DCT-based detection of copy-move forgery in images. *Forensic Science International*, *206* (1-3), pp. 178-184.

[79] Lanh, T. V., Ching, K.-S., Emmanuel, S., & Kankanhalli, M. S. (2007). A survey on digital camera image forensic methods. *IEEE International Conference on Multimedia and Expo*, (pp. 16-19). Beijing.

[80] Nagosky, D. P. (2005, December). The admissibility of digital photographs in criminal cases. *Law Enforcement Bulletin*, *74* (12).

[81] Neelamani, R., Queiroz, R. d., Fan, Z., & Baraniuk, R. (2006). JPEG compression history estimation for color images. *IEEE Transactions on Image Processing*, *15* (6), 1365-1378.

[82] Ng, T.-T., Chang, S.-F., Lin, C.-Y., & Sun, Q. (2006). *Multimedia security technologies for digital rights management chapter 15: Passive-blind image forensics.* (W. Zeng, H. Yu, & C.-Y. Lin, Eds.) Burlington: Academic Press.

[83] Swaminathan, A., Wu, M., & Liu, K. J. (2009, March). Component forensics. *IEEE Signal Processing Magazine*, *26* (2), pp. 38-48.

[84] Swaminathan, A., Wu, M., & Liu, K. J. (2006). Component forensics of digital cameras: A non-intrusive approach. *40th Annual Conference on Information Sciences and Systems*, (pp. 1194-1199). Princeton.

[85] Tjoa, S., Lin, W. S., & Liu, K. J. (2007). Transform coder classification for digital image forensics. *IEEE International Conference on Image Processing*, *6*, pp. 105-108. San Antonio.

[86] Sturak, J. R. (2004). *Forensic analysis of digital image tampering.* Washington, DC: Storming Media.

[87] Popescu, A. C. (2005). *Statistical tools for digital image forensics.* Unpublished PhD dissertation, Hanover, NH: Dartmouth College, Department of Computer Science.