https://doi.org/10.2352/ISSN.2470-1173.2017.6.MOBMU-303

A Forensic Mobile Application Designed for both Steganalysis and Steganography in Digital Images

Enping Li; Department of Computer Science, Bridgewater State University, Bridgewater, Massachusetts, USA
Jun Yu; Marvell Semiconductor, Inc; Marlborough, MA, USA

Abstract

With the rapid development of mobile devices and multimedia processing technologies, digital multimedia applications has become increasingly more popular in our daily life. Due to the nature of digital media, digital images can be easily modified without leaving obvious traces. Digital image forensics is an emerging research field which aims to address the major problems as forgery detection, source identification, image recovery and detecting the existence of hidden information, which is also referred as steganalysis. Steganography is the science of covert communication which aims to conceal the existence of the secret information hidden in the communication. Steganalysis is the study of detecting hidden information, which is embedded by using steganography techniques. In this paper we present a forensic mobile application developed on iOS platform. This application is designed to perform both steganalysis and steganorgraphy tasks in digital images; that is to conduct information analysis of given images and determine if there is any secret information hidden in the given images, and also fulfill the task of hiding information invisibly into digital images. There have been a number of well-established techniques in digital image steganalysis and steganography fields during recent years. However, there are very few mobile forensic tools that have been developed to comprehensively adopt these methods. Our forensic mobile application aims to systematically include significant methods both in steganography and steganalysis fields, so that people can use it as a forensic tool to determine if an image contains a hidden message, as well as use it as a security tool to perform covert communication by hiding information in an image.

Introduction

There has been an explosive growth in mobile device usage in recent years. With the rapid development of mobile devices and multimedia processing techniques, the operations once performed only on computers can now be carried out on the new platforms. People have increasingly relied on mobile devices for communication and multimedia editing. Digital multimedia, such as images and videos, are becoming one of the major information carriers. The ease of the digital multimedia acquisition, modification and synthesis brings convenience for users as well as raises security concerns and opens new avenues for research in mobile security.

Steganography, a promising field in security, has attracted more research attention recently. Steganography refers to the science of performing covert communication by hiding a message into a carrier medium, for example, to conceal a text message in an image [1, 2]. Comparing to cryptography, which aims to keep the communication confidential from an eavesdropper, steganography strives to conceal the existence of the message from a censor [3, 4]. Steganography is a double-edged sword: it can be used to enforce the communication security by hiding secret information in innocent-looking carriers; on the other hand, it can also be employed by malware to smuggle malicious information. Either of the aforementioned uses of steganography may pose a challenge to digital forensics.

There are many different carrier formats which can be used for steganography, such as images, IP packets, audio and games [5, 6, 7]. Images are the most popular ones because they are widely used in various communication scenarios and have comparatively larger data payload [8, 9, 10, 11]. In the ever changing environment, some image steganography applications are launched in the mobile markets and start to draw attention from mobile users. A common use of these applications is to enable the user to embed a secret message, such as text and image, into a carrier image without causing perceptible changes. The user can achieve covert communication by sending the modified carrier image.

There has been an great amount of work done and progress achieved in the fields of image steganography and steganalysis [12]. Most of the current mobile applications are focusing on implementation of the very basic steganography methods, such as LSB embedding [13]. The major goal of this paper is to develop a mobile application which systematically includes significant methods in both steganography and steganalysis fields [14, 15, 16]. So that users can use this application to perform forensics task of detecting hidden message in the digital images as well as use it as a security tool to hide secret message in the digital images to achieve a covert communication.

The Proposed Forensic Mobile Application

In this section, we introduce the proposed forensics application. This application is designed to perform two major tasks: detecting secret information from digital images (steganalysis) and hiding information in digital images (steganography).

There are different types of methods in the steganography and steganalysis fields. The methods that we are interested in including in the forensics application are summarized and illustrated in the following subsections.

Steganography Methods

Based on the domains that the steganography methods are implemented, there are three major types: spatial domain methods, transform domain methods and adaptive methods, which combine the usage of both spatial and transform domain.

The main idea of the spatial domain methods is to modify the bits of pixel values to embed secret information. The spatial

domain methods that will be included in our forensics mobile application is:

- *Steganography in Spatial Domain*
  - Least significant bit (LSB) steganography: LSB embedding is the most basic steganography method that hides a secret message in the LSBs of pixels values. The advantages of LSB embedding are that the distortions induced by LSB embedding are imperceptible and it is very easy to implement. While, the disadvantage is that LSB embedding is not reliable and embedding message can be easily destroyed by comment image processing, such as compression.

There are different types of transform domains in image processing, such as discrete cosine transform (DCT) which is used by JPEG compression and discrete wavelet transform (DWT) which is used by JPEG2000 compression. We focus on the methods in DCT domain since JPEG is the most popular image format which is based on DCT.

- *Steganography in DCT Domain*
  - JSteg: this method is similar with LSB embedding, except that JSteg uses the LSBs of non-zero DCT coefficients for embedding instead of using LSBs of pixel values.
  - F5: matrix embedding is introduced in this method, which is able to embed multiple bits by making at most one bit modification. This method greatly improves the embedding efficiency and decreases the possibility of being detected.

Steganalysis Methods

Steganalysis is the study of detecting the existence of the hidden messages. There are two major types of steganalysis: targeted steganalysis and blind steganalysis. Targeted steganalysis is applied when we know what steganography method has been used for embedding, while blind steganalysis does not require knowing the steganography algorithm being used.

The well-know steganalysis methods for the above mentioned steganography methods are summarized as follows:

- *Targeted Steganalysis*
  - Histogram Analysis: this method is based on the observation that if an image is fully embedded by using LSB embedding, the number of pixels or coefficients with $2i$ and $2i+1$ values should be approximately the same. The histogram analysis method works well for fully LSB embedding in both spatial and DCT domain.
  - Sample pairs analysis: utilizing the features of the spatial correlation within images to explore the artifacts of embedding. This method is more reliable and accurate, which works well when the images are partially embedded.

- *Blind Steganalysis*
  - JPEG Steganalysis using calibration: this method utilizes the feature of JPEG image compression that quantized DCT coefficients are robust to small distortions. So that we can estimate the cover image (the original image which is used for embedding) from the stego image (the modified image which contains the embedded information). The estimated cover image further can be used for construction of features for blind steganalysis, which can also be used as a targeted steganalysis against F5.

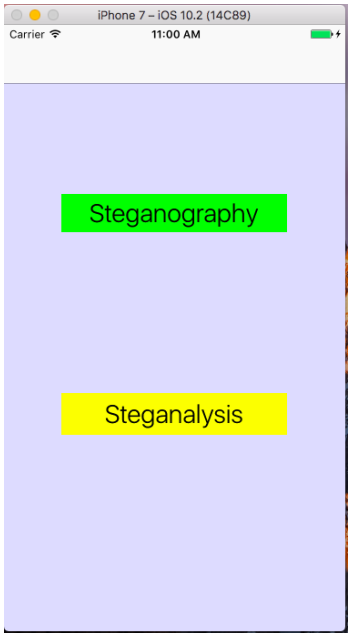

Figure 1. *Main User Interface*

Figure 2. *The steganography option user interface*

Application Development

Our forensic mobile application is developed on iOS platform. We used Xcode, Apple's default IDE, and swift 3.0 programming language for the application development. In addition to the default APIs, we have mainly used Core Image framework, vImage framework and libjpeg library for image processing, and used iOS-Charts library for plotting figures. The main user interface is shown in figure 1. There are two major options: Steganography and Steganalysis. Steganography option enables the user to perform information hiding and information detection via three different methods. Steganalysis option performs analysis for steganography methods. The features and usage of these two major options are further explained as follows:

- *Steganography Option*: the user interface is shown in figure 2.
    1. *Embed* switch: user needs to decide which operation will be performed first. When the switch is turned on, the embedding function will be perform, otherwise the detection function will executed.
    2. *Choose* button: user needs to choose an image from the photo album for message embedding or detection as shown in figure 3. The selected image will be shown in the image viewer as shown in figure 4.
    3. *Message Input/Output* window: if *embed* switch is turned on, user can input the text message for embedding; if *embed* is turned off, the detected message will be displayed in the window. If the window is empty, the default message used for embedding is uniform random bits and fully embedding will be performed.
    4. *info* field: shows the operation status.
    5. *LSB/JSteg/F5* buttons: performs the corresponding algorithm for embedding or detection. For example, If *Embed* is turned on and any of the *LSB/JSteg/F5* buttons is pressed, the message in the text window will be embedded to the original image to generate a stego image. The stego image will be automatically saved to photo album and loaded in the image viewer.
    6. Examples of performing message embedding and detection are shown in figure 4 and figure 5.

- *Steganalysis Option*: the user interface is shown in figure 6.
    1. *Choose* button: user needs to choose an image from

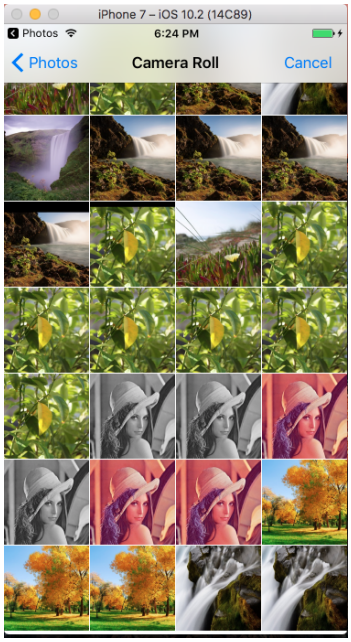

Figure 3. *Selecting image from photo album for information hiding or detection*

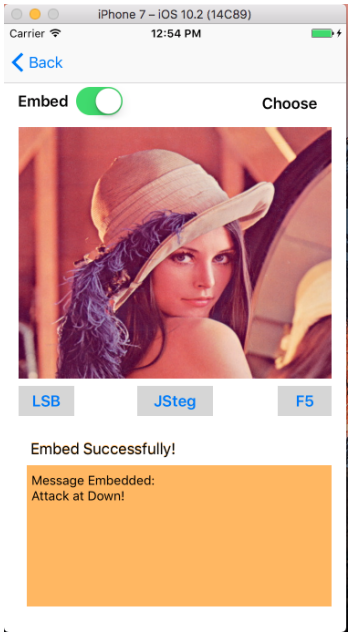

Figure 4. *Message is embedded successfully*

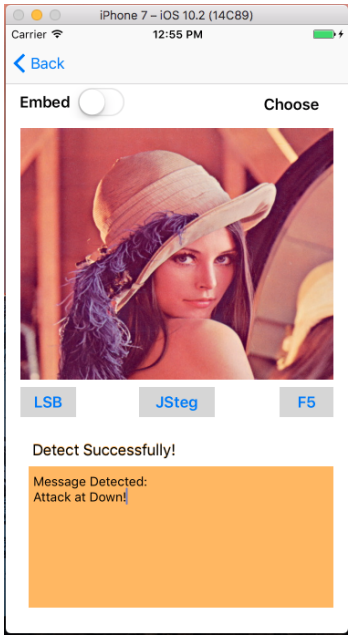

Figure 5. *Message is detected successfully*

the photo album for steganalysis. The selected image will be shown in the image viewer as shown in figure 7.

2. *Histogram/Calibrated/Pattern* buttons: performs the corresponding steganalysis algorithm. By clicking on *Histogram* button, the histogram segue will be triggered and lead to the histogram view, further in the histogram view, user can choose to perform histogram analysis either in spatial domain or DCT domain as shown in figure 8 and figure 9. The user can also specify the histogram range to have a better view on the details. *Calibrated* and *Pattern* buttons are designed to include the calibrated feature analysis and further pattern classification analysis for future work.

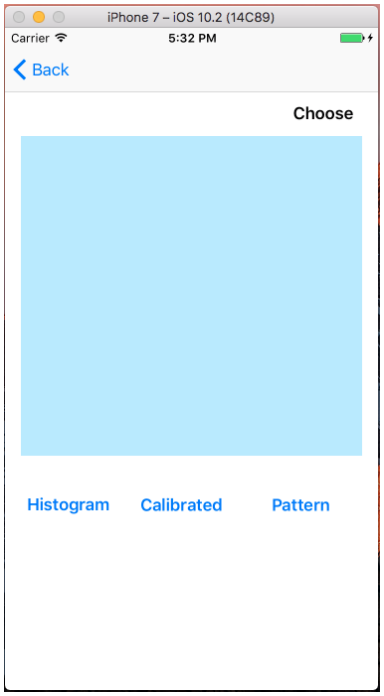

Figure 6. *The steganaylysis option user interface*

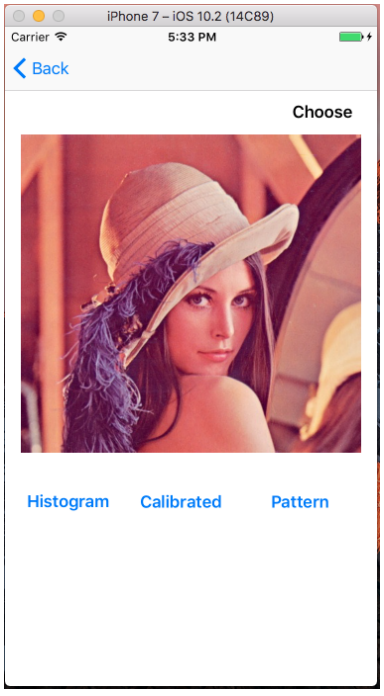

Figure 7. *Load the target image for steganalysis*

Experimental Results

Since our forensic mobile application aims to perform image steganalysis for forensics purposes as well as to implement steganography algorithms for security purposes. So we run two types of experiments on our forensic mobile application.

- *Part I: Experiments on Steganography*
  1. Select a set of different types of digital images as the cover images.
  2. Embed secret information into the cover images by using each of the above-mentioned steganography algorithms
  3. Check to see if there is any visible artifacts caused by the embedding
  4. Apply the common image processing of JPEG conversion into the stego images.
  5. compare the reliability and robustness of different Steganography algorithms.

The nature of the LSB, JSteg and F5 algorithms indicates no obvious visual artifacts after the embedding since all the algorithms use the LSB plane (either in spatial domain or DCT domain) to embed the message. Our experiments conform to this fact. We used each of the algorithms to test on different images. By visually checking on the contents, there are no obvious differences between the cover images and stego images. One common image processing operation in multimedia communication is JPEG conversion. We tried to convert the formats of BMP and PNG into highest quality of JPEG format (default saving format on iOS) for LSB embedding, the embedded message has been greatly impaired. While JSteg and F5 perform message embedding in DCT domain, if the modified DCT coefficients are saved directly without further recompression (we used libjpeg library for this purpose), the messages remain intact.

- *Part II: Experiments on Steganalysis*
  1. Get the stego images obtained from Part I experiments
  2. Perform forensics analysis by using the steganalysis algorithms.
  3. Compare the reliability and accuracy of steganalysis algorithms.

Pixel histogram is used to analyze LSB embedding. For example, we used lena image as a test image and performed a full embedding using random bits. Figure 8 shows the pixel histogram at the range $[20, 50]$ for the stego lena image. In this histogram figure, there is a clear staircase (the adjacent two bins are approximately evened out) pattern, which can be used to identify the LSB embedding. DCT coefficient histogram is used for both JSteg and F5 algorithms. Figure 8 shows an example of DCT coefficient

histogram at the range of $[-20, 20]$ for the stego lena image fully embedded with random bits by using JSteg algorithm. In this histogram, there is also a stair case effect for all the bins except peak bins at 0 and 1, which can be used to identify the JSteg algorithm. However, histogram analysis is not successful on F5, since F5 has a much better embedding efficiency (one bit modification is caused by at least multiple bits embedding), the distribution shape of the histogram is not changed significantly and there is no stair case pattern as shown in figure 9.

Conclusions

The rapid growth in the usage of mobile devices and digital multimedia contents has brought great convenience to users. On the other hand, it also brings a series of forensic-related issues as we can no longer take the integrity of digital multimedia contents for granted. Steganography and steganalysis have increasingly attracted attention over recent years. Since the techniques of steganography can be used to secure communication by providing covert communication through secretly embedded information in the digital multimedia contents, and the methods of steganalysis can be used to detect the existence of secret communication, which might be related with illegal activities. The major contribution of this paper is to develop a forensic mobile application which systematically include both the steganography and steganalysis methods of digital images. Our forensic mobile application has two main components: the steganography component enables the user to obtain a covert communication by hiding data in the digital images through different steganography methods, and steganalysis component provides a function of detecting the existence of hidden message through various steganalysis techniques. Our application has shown to provide a convenient way to demonstrate and practice both steganography and steganalysis algorithms.

We included *Calibrated* and *Pattern* templates in our application for future development. Other potential algorithms can be further explored and integrated in our application to make it capable of performing more comprehensive tasks. Since this is a newly emerging area, there are few number of image steganography mobile applications, such as Steganography, Steganography Master, Steganographia, MobiStego, WeHide and Hide It In. There has been very little research regarding the performance of these applications. Another potential future direction in this field is to compare the reliability and robustness of these mobile steganography applications and find out the answers for the following questions: from the communicators side, can the secret message be embedded well without attracting censors attention? Can the secrete message be retrieved successfully after general image processing? From the digital forensics investigators side, is it practical to perform multimedia forensics on these applications by using present forensics tools?

References

[1] G. J. Simmons, "The prisoners' problem and the subliminal channel," in *CRYPTO*, 1983, pp. 51–67.

[2] R. J. Anderson and F. A. P. Petitcolas, "On the limits of steganography," *IEEE Journal of Selected Areas in Communications*, vol. 16, no. 4, pp. 474–481, 1998, special issue on copyright & privacy protection.

[3] W. Stallings, "Cryptography and network security: Principles and practice." Prentice Hall Press, 2010, pp. 301–302.

[4] P. Wayner, *Disappearing Cryptography: Information Hiding: Steganography and Watermarking (2nd Edition)*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2002.

[5] M. Diehl, "Secure covert channels in multiplayer games," in *MM&Sec '08: Proceedings of the 10th ACM workshop on Mul-*

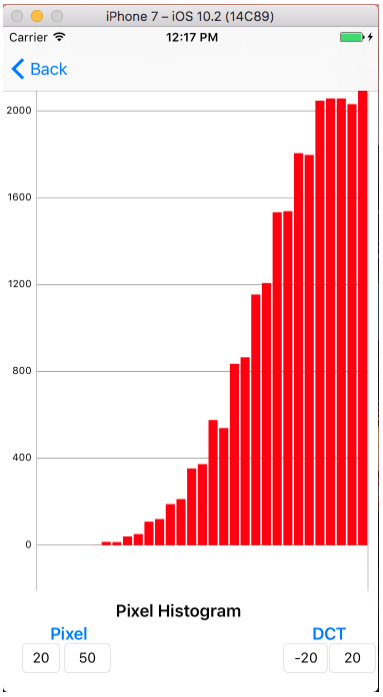

Figure 8. *Histogram analysis for fully LSB embedding*

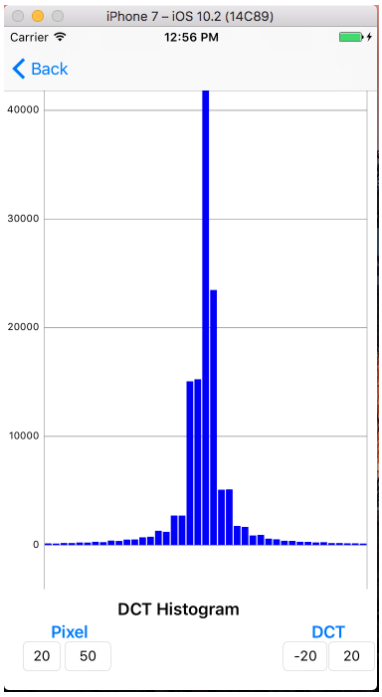

Figure 9. *Histogram analysis for fully JSteg embedding*

timedia and security. New York, NY, USA: ACM, 2008, pp. 117–122.

[6] J. C. H. Castro, I. Blasco-Lopez, J. M. Estévez-Tapiador, and A. R. Garnacho, "Steganography in games: A general methodology and its application to the game of go," *Computers & Security*, vol. 25, no. 1, pp. 64–71, 2006.

[7] D. Inoue and T. Matsumoto, "Scheme of standard MIDI files steganography and its evaluation," *SPIE Security and Watermarking of Multimedia Contents IV*, vol. 4675, pp. 194–205, 2002.

[8] S. Craver, "On public-key steganography in the presence of an active warden," in *Information Hiding, Second International Workshop*. Springer, 1996, pp. 355–368.

[9] E. Li and S. Craver, "A supraliminal channel in a wireless phone application," in *Proceedings of the 11th ACM workshop on Multimedia and security*, ser. MM&Sec '09. New York, NY, USA: ACM, 2009, pp. 151–154.

[10] S. Craver, E. Li, J. Yu, and I. Atakli, "Information hiding," K. Solanki, K. Sullivan, and U. Madhow, Eds. Berlin, Heidelberg: Springer-Verlag, 2008, ch. A Supraliminal Channel in a Videoconferencing Application, pp. 283–293.

[11] E. Li and S. Craver, "A square-root law for active wardens," in *Proceedings of the thirteenth ACM multimedia workshop on Multimedia and security*, ser. MM&Sec '11. New York, NY, USA: ACM, 2011, pp. 87–92.

[12] S. Craver, E. Li, and J. Yu, "Protocols for data hiding in pseudorandom state," in *Media Forensics and Security*, 2009, p. 72540.

[13] N. F. Johnson and S. Jajodia, "Exploring steganography: Seeing the unseen," *Computer*, vol. 31, no. 2, pp. 26–34, Feb. 1998.

[14] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*, 2nd ed. Morgan Kaufmann Publishers Inc., 2008.

[15] K. C. Abbas Cheddad, Joan Condell and P. M. Kevitt, "Digital image steganography:survey and analysis of current methods," *Signal processing*, vol. 90, no. 3, pp. 727–752, 2010.

[16] J. H. Bin Li, Junhui He and Y. Shi, "A survey on image steganography and steganalysis," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 2, no. 2, 2011.

Author Biography

Enping Li received the B.S. degree in electrical engineering from North China University of Technology, Beijing, China, in 2002, the M.S. degree in electrical engineering from China University of Petroleum, Beijing, China, in 2006, and the Ph.D. degree in electrical engineering from State University of New York at Binghamton, NY, in 2012. She is currently an Assistant Professor of Computer Science at Bridgewater State University, MA. She has engaged in research on information security, covert communications and multimedia forensics. Dr. Li is an Associate Member of American Academy of Forensic Sciences and a member of ACM and IEEE.

Jun Yu received the B.S. degree in electrical engineering in 2001 and the M.S. degree in electrical engineering in 2004 from Lanzhou University in Lanzhou, Gansu, China, and received the Ph.D. degree in electrical engineering from SUNY Binghamton University, Binghamton, NY, USA, in 2011. He is currently a Senior Software Engineer working on security firmware for embedded systems at Marvell Semiconductor, Inc, Marlborough, MA, USA. His research has been concerned with trusted computing on embedded systems, information hiding, and digital forensics. Dr. Yu is a member of the Sigma Xi society. Dr. Yu was a recipient of an Exceptional Contribution Award by Marvell Semiconductor, Inc, in 2013, for his work of bringing a SoC firmware to pass the FIPS 140-2 Level-3 Certification.