



Extracting Hidden Messages in Steganographic Images

By

Tu-Thach Quach

Presented At

The Digital Forensic Research Conference

DFRWS 2014 USA Denver, CO (Aug 3rd - 6th)

DFRWS is dedicated to the sharing of knowledge and ideas about digital forensics research. Ever since it organized the first open workshop devoted to digital forensics in 2001, DFRWS continues to bring academics and practitioners together in an informal environment. As a non-profit, volunteer organization, DFRWS sponsors technical working groups, annual conferences and challenges to help drive the direction of research and development.

<http://dfrws.org>

Extracting Hidden Messages in Steganographic Images

Tu-Thach Quach

Sandia National Laboratories

DFRWS 2014

August 4, 2014



Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Simple LSB Steganography

0	1	1
---	---	---

7	8	7
7	8	9
8	8	9

Simple LSB Steganography

0	1	1
---	---	---

0111

7	8	7
7	8	9
8	8	9

Simple LSB Steganography

0	1	1
---	---	---

0110

6	8	7
7	8	9
8	8	9

Simple LSB Steganography

0	1	1
---	---	---

1000

8	8	7
7	8	9
8	8	9

Simple LSB Steganography

0	1	1
---	---	---

1000

8	8	7
7	8	9
8	8	9

Simple LSB Steganography

0	1	1
---	---	---

0111

8	7	7
7	8	9
8	8	9

Simple LSB Steganography

0	1	1
---	---	---

0111

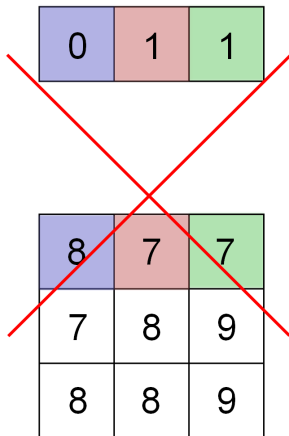
8	7	7
7	8	9
8	8	9

Simple LSB Steganography

0	1	1
---	---	---

8	7	7
7	8	9
8	8	9

Simple LSB Steganography

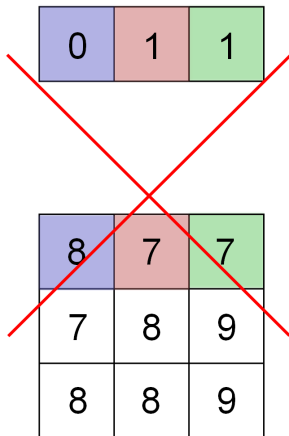


The diagram illustrates a flaw in simple LSB steganography. It shows a 3x3 grid of pixel values. The top row is highlighted with colored backgrounds: blue for the first cell (0), red for the second (1), and green for the third (1). The rest of the grid contains the following values:

0	1	1
8	7	7
7	8	9
8	8	9

A large red 'X' is drawn over the entire grid, indicating that this method is flawed or incorrect.

Simple LSB Steganography

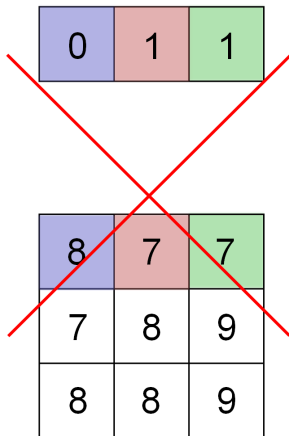


The diagram illustrates a 3x3 grid of pixel values. The top row is highlighted with colored backgrounds: blue for the first cell (0), red for the second cell (1), and green for the third cell (1). The remaining cells are white. A large red 'X' is drawn over the entire grid, indicating that the distortion is not distributed over the image.

0	1	1
7	8	9
8	8	9

- Distortion not distributed over image

Simple LSB Steganography



The diagram illustrates a 3x3 grid of pixel values. The top row is highlighted with colored backgrounds: blue for the first cell (0), red for the second cell (1), and green for the third cell (1). A large red 'X' is drawn over the entire grid, indicating that the distortion is not distributed over the image.

0	1	1
8	7	7
7	8	9
8	8	9

- Distortion not distributed over image
- Message can be extracted

Simple LSB Steganography

0	1	1
---	---	---

- Use embedding key to distribute payload over image

7	8	7
7	8	9
8	8	9

Simple LSB Steganography

0	1	1
---	---	---

7	8	7
7	8	9
8	8	9

- Use embedding key to distribute payload over image
- Message can be extracted if embedding key is known (shared by sender and receiver)

Group-Parity Steganography

0	1	1
---	---	---

7	8	7
7	8	9
8	8	9

- Use k pixels to embed a single bit
- First bit: $0 = 8 + 8 \mod 2$

Matrix Embedding

0	1	1
---	---	---

7	8	7
7	8	9
8	8	9

- Use k pixels to embed q bits
- Change at most 1 pixel in each group

Forensic Goal

Extract hidden messages

Forensic Goal

Extract hidden messages

Approaches:

- Embedding key search

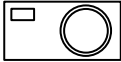
Forensic Goal

Extract hidden messages

Approaches:

- Embedding key search
- Payload location

Scenario



Scenario



C:\Photos
ImageA.tif
ImageB.tif
:
ImageZ.tif

Scenario



Residuals

Cover image: $\mathbf{c} = (c_1, \dots, c_n)$

Stego image: $\mathbf{s} = (s_1, \dots, s_n)$

Residuals

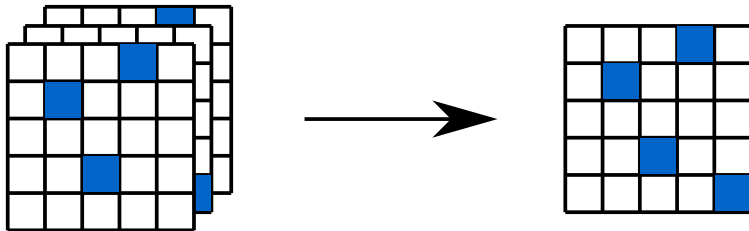
Cover image: $\mathbf{c} = (c_1, \dots, c_n)$

Stego image: $\mathbf{s} = (s_1, \dots, s_n)$

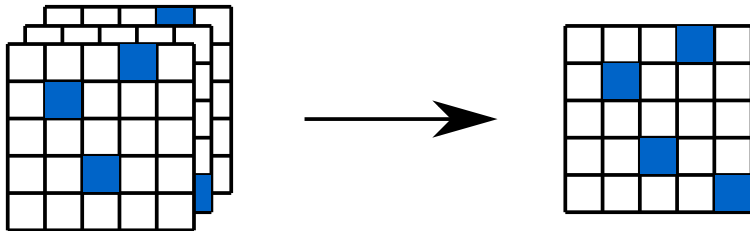
Residual r_i is

$$r_i = |c_i - s_i|.$$

Payload Location: Simple LSB



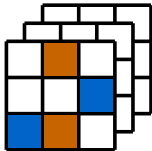
Payload Location: Simple LSB



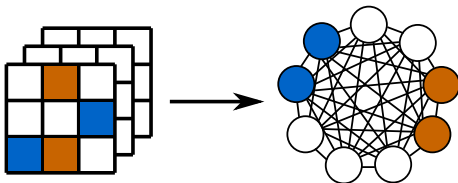
On average: $\log_2 m$ images to locate payload.

Quach, T.-T., "Optimal Cover Estimation Methods and Steganographic Payload Location," *IEEE Trans. Info. Forensics and Security*, 2011.

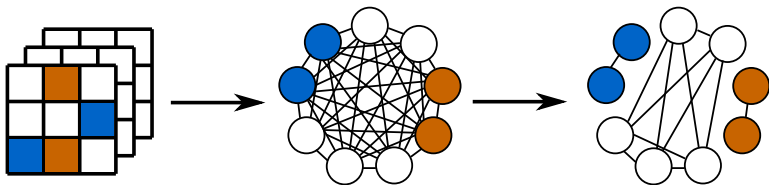
Payload Location: Group-Parity



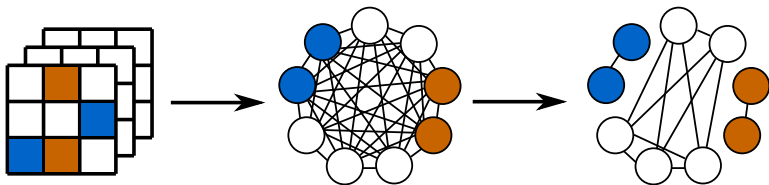
Payload Location: Group-Parity



Payload Location: Group-Parity



Payload Location: Group-Parity



On average: $8k^2 \log(km)$ images to locate payload.

Quach, T.-T., "Locating Payload Embedded by Group-Parity Steganography," *Digital Investigation*, 2012.

Payload Location

No logical information to arrange located payload

Payload Location

No logical information to arrange located payload

Observation:

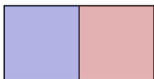
- Residuals provide logical information if payload size is not fixed

Logical Information



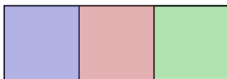
0	0	0
1	0	0
0	0	0

Logical Information



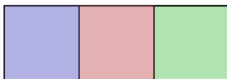
0	1	0
2	0	0
0	0	0

Logical Information



0	2	0
3	0	0
0	0	1

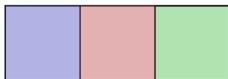
Logical Information



- Payload pixels: 2, 4, 9

0	2	0
3	0	0
0	0	1

Logical Information

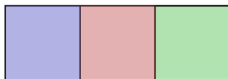


■ Payload pixels: 2, 4, 9

■ $r_4 > r_2 > r_9$

0	2	0
3	0	0
0	0	1

Logical Information



0	2	0
3	0	0
0	0	1

- Payload pixels: 2, 4, 9
- $r_4 > r_2 > r_9$
- Order located payload in descending mean residuals to obtain message

Logical Information

If payload size varies from 1 through m :

$$E[R_i] > E[R_j]$$

for all logical payload pixels i, j where $i < j$.

Logical Information

If payload size varies from 1 through m :

$$E[R_i] > E[R_j]$$

for all logical payload pixels i, j where $i < j$.

If payload size is uniformly distributed:

$$E[R_i] = \frac{m + 1 - i}{2m}.$$

No Cover Images

What if C:\Photos deleted?

No Cover Images

What if C:\Photos deleted?

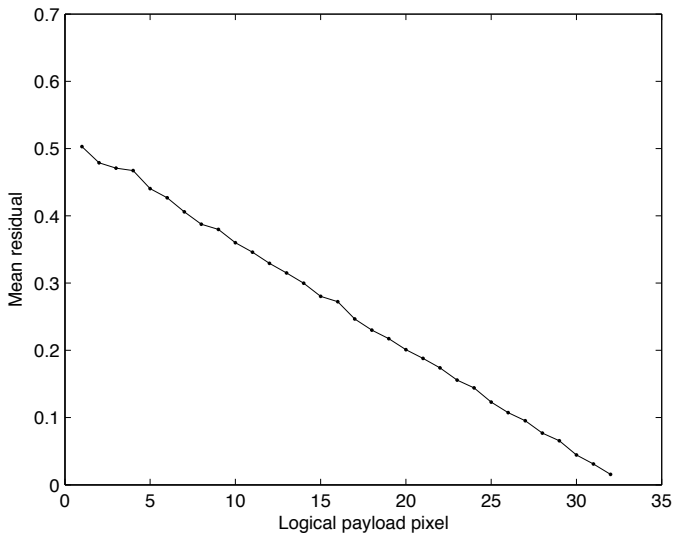
Approach:

- Estimate cover images

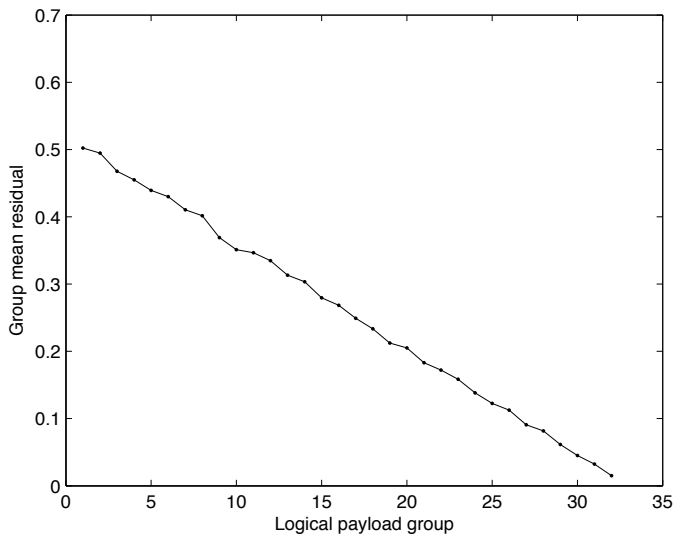
Experiments

- Image set: BOSSbase 9074 grayscale images 512×512
- Embedding algorithms: simple LSB and group-parity steganography
- Payload size: between 1 and 32 (uniformly distributed)
- Metric: Minimum edit distance
- Cover estimator: Markov random field

Known Cover: Simple LSB Residuals



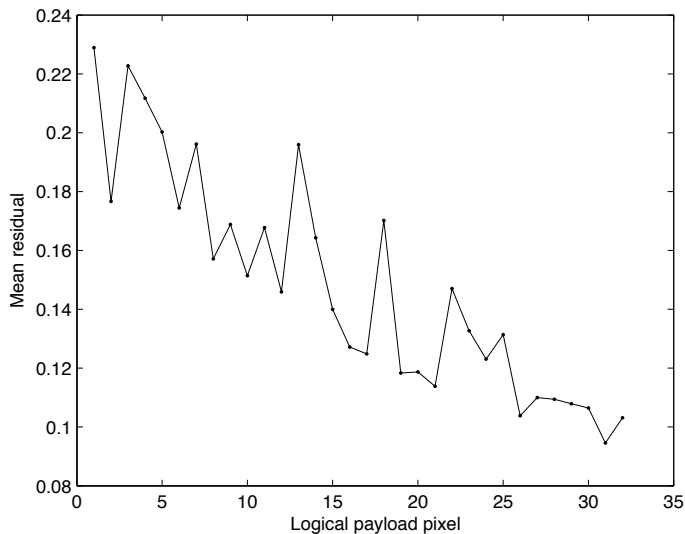
Known Cover: Group-Parity Residuals



Known Cover: Minimum Edit Distance

Images	Simple LSB	Group-Parity
1000	8.0	9.5
2000	5.6	4.2
3000	3.3	2.8
4000	2.0	1.8
5000	1.6	1.2
6000	1.0	0.8
7000	1.0	0.0
8000	0.6	0.0
9000	0.0	0.0

Unknown Cover: Simple LSBR Residuals



Unknown Cover: Minimum Edit Distance

Images	Replacement	Matching
1000	24.7	27.4
2000	24.7	27.3
3000	23.8	26.4
4000	23.3	26.3
5000	23.3	25.7
6000	23.0	25.7
7000	22.3	25.2
8000	21.9	25.2
9000	21.8	25.0

Conclusions

- Exposes vulnerability in block-based embedding algorithms
- Many challenges in practice
- May improve with advances in cover estimation
- Image collection may contain several embedding keys

Thank You

tong@sandia.gov