Steganography and Steganalysis in Digital Images

Brijesh Singh
Ph.D. Research Scholar

Department of Computer Science and Engineering
IIT Guwahati

Monday 18th November, 2019

Outline

Steganography

Steganography

- Steganography = Steganos (covert) + graphia (writing)
- A way of secret transmission of data through innocent looking carrier or cover

History of steganography and steganalysis

- Steganography methods have been used for centuries

History of steganography and steganalysis

- Steganography methods have been used for centuries
- In ancient Greek times, **slaves tattooed message on their shaved heads** and **send when hair is regrown**

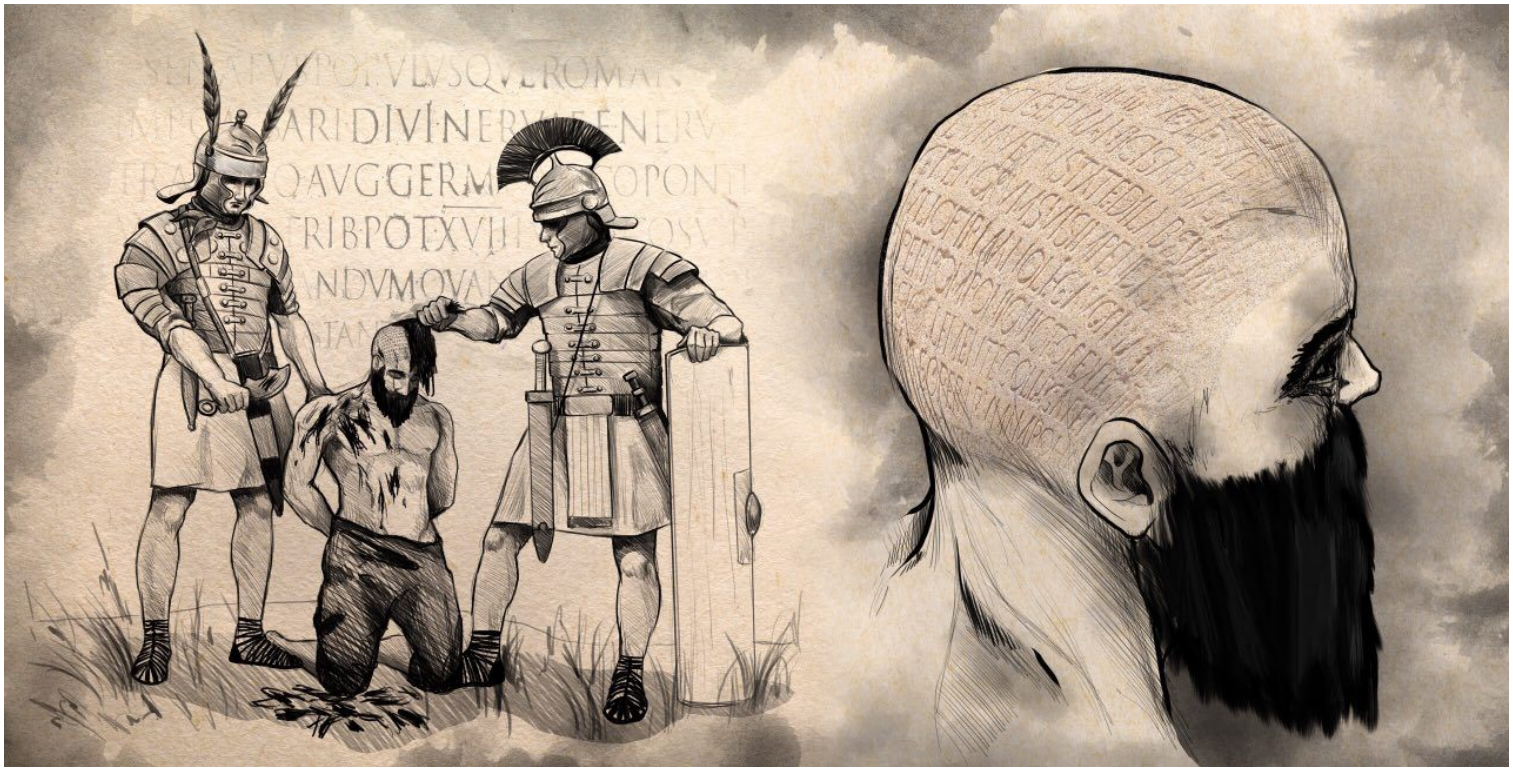


Figure: image src: googleimages

History of steganography

- Message to be hidden was written on the **wood** and was covered with new **wax layer**

History of steganography

- Message to be hidden was written on the **wood** and was covered with new **wax layer**
- During **World War-II, Milk, Fruit Juices, Vinegar** were used for writing secret messages

History of steganography

- Message to be hidden was written on the **wood** and was covered with new **wax layer**
- During **World War-II, Milk, Fruit Juices, Vinegar** were used for writing secret messages
- During 1990's secret messages are hidden into some **digital files**

Characteristics

- **Security**. presence of hidden message should not be detected with probability more than random guess
- **Capacity**. must be as high as possible
- **Imperceptibility**. should not be perceviable by human eye

Classification of Steganography

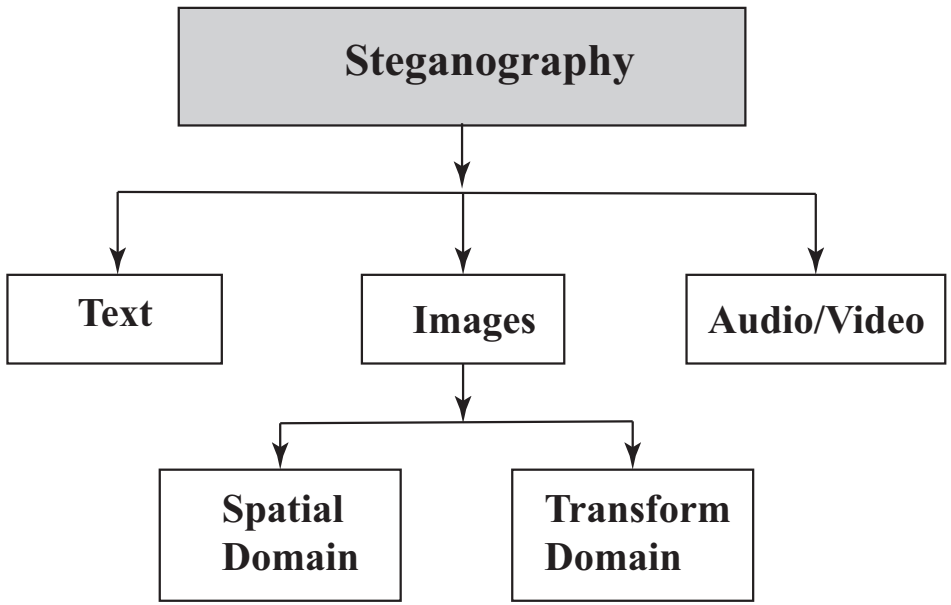


Figure: Classification of Steganography methods

Text Steganography

- Hiding secret information in text

Example!!

Example: Alice sends the following e-mail to her friend **Bob**:

My friend Bob,

until yesterday I was using binoculars for stargazing. Today, I decided to try my new telescope. The galaxies in Leo and Ursa Major were unbelievable! Next, I plan to check out some nebulas and then prepare to take a few snapshots of the new comet. Although I am satisfied with the telescope, I think I need to purchase light pollution filters to block the xenon lights from a nearby highway to improve the quality of my pictures.

Cheers,
Alice.

- It seems conversation between two astronomers

- It seems conversation between two astronomers
- Reality: Alice and Bob are spies

- It seems conversation between two astronomers
- Reality: Alice and Bob are spies
- Bob knows that Alice using steganography and follows prearranged **protocol**

- It seems conversation between two astronomers
- Reality: Alice and Bob are spies
- Bob knows that Alice using steganography and follows prearranged **protocol**
- Here, Predecided protocol is:
  1st letter of each word $\Longrightarrow$ expansion of π $\Longrightarrow$ peceding letter

- It seems conversation between two astronomers
- Reality: Alice and Bob are spies
- Bob knows that Alice using steganography and follows prearranged **protocol**
- Here, Predecided protocol is:
  1st letter of each word $\Longrightarrow$ expansion of π $\Longrightarrow$ peceding letter
- Bob starts by listing the **1st letter** of all words.

My **f**riend **B**ob,
until **y**esterday **I** **w**as **u**sing **b**inoculars **f**or **s**targazing. **T**oday, **I** **d**ecided **t**o **t**ry **m**y **n**ew **t**elescope. **T**he **g**alaxies **i**n **L**eo **a**nd **U**rsa **M**ajor **w**ere **u**nbelievable! **N**ext, **I** **p**lan **t**o **c**heck **o**ut **s**ome **n**ebulas **a**nd **t**hen **p**repare **t**o **t**ake **a** **f**ew **s**napshots **o**f **t**he **n**ew **c**omet. **A**lthough **I** **a**m **s**atisfied **w**ith **t**he **t**elescope, **I** **t**hink **I** **n**eed **t**o **p**urchase **l**ight **p**ollution **f**ilters **t**o **b**lock **t**he **x**enon **l**ights **f**rom **a** **n**earby **h**ighway **t**o **i**mprove **t**he **q**uality **o**f **m**y **p**ictures.
Cheers,
Alice.

My **f**riend **B**ob,
until **y**esterday **I** **w**as **u**sing **b**inoculars **f**or **s**targazing. **T**oday, **I** **d**ecided **t**o **t**ry **m**y **n**ew **t**elescope. **T**he **g**alaxies **i**n **L**eo **a**nd **U**rsa **M**ajor **w**ere **u**nbelievable! **N**ext, **I** **p**lan **t**o **c**heck **o**ut **s**ome **n**ebulas **a**nd **t**hen **p**repare **t**o **t**ake **a** **f**ew **s**napshots **o**f **t**he **n**ew **c**omet. **A**lthough **I** **a**m **s**atisfied **w**ith **t**he **t**elescope, **I** **t**hink **I** **n**eed **t**o **p**urchase **l**ight **p**ollution **f**ilters **t**o **b**lock **t**he **x**enon **l**ights **f**rom **a** **n**earby **h**ighway **t**o **i**mprove **t**he **q**uality **o**f **m**y **p**ictures.
Cheers,
Alice.

$\Longrightarrow$ **"mfbuyiwubfstidttmnttgilaumwuniptcosnatpttafsotncaiaswttit-intplpftbtxlfanhtitqompca"**

- Bob writes down expansion of π

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"a**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"at**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"att**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"atta**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"attac**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"attack**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"attack t**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"attack to**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"attack tom**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"attack tomo**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"attack tomor**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"attack tomorr**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"attack tomorro**

- Bob writes down expansion of π
- $\pi = 3.141592653589793...$
- Picks the letters in this sequence from generated text
- i.e, third letter, then the next first letter, then the next fourth letter,...
- Results in: **"buubdlupnpsspx"**
- Finally, he **replaces each letter with its preceeding letter in alphabet**.
- Result: **"attack tomorrow"**

Terminologies

- **Cover** - text, images, audio, video, etc.
- **Stego** - After embedding cover object known as stego
- **Payload** - Amount of data hidden (bits-per-pixel or bpp for images)

Steganography in Digital Images

Why Steganography in Digital Images?

- A gray scale image = A 2-D array of 8-bit values
- Each pixel has 8-bits. i.e, pixel values$\in [0, 255]$

Image

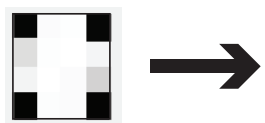
10	248	250	15
249	250	252	253
251	253	251	250
12	249	248	16

Image as array

Steganography in Digital Images

Why Steganography in Digital Images?

- A gray scale image = A 2-D array of 8-bit values
- Each pixel has 8-bits. i.e, pixel values$\in [0, 255]$



Image

10	248	250	15
249	250	252	253
251	253	251	250
12	249	248	16

Image as array

MSB (bit 8) LSB (bit 1)

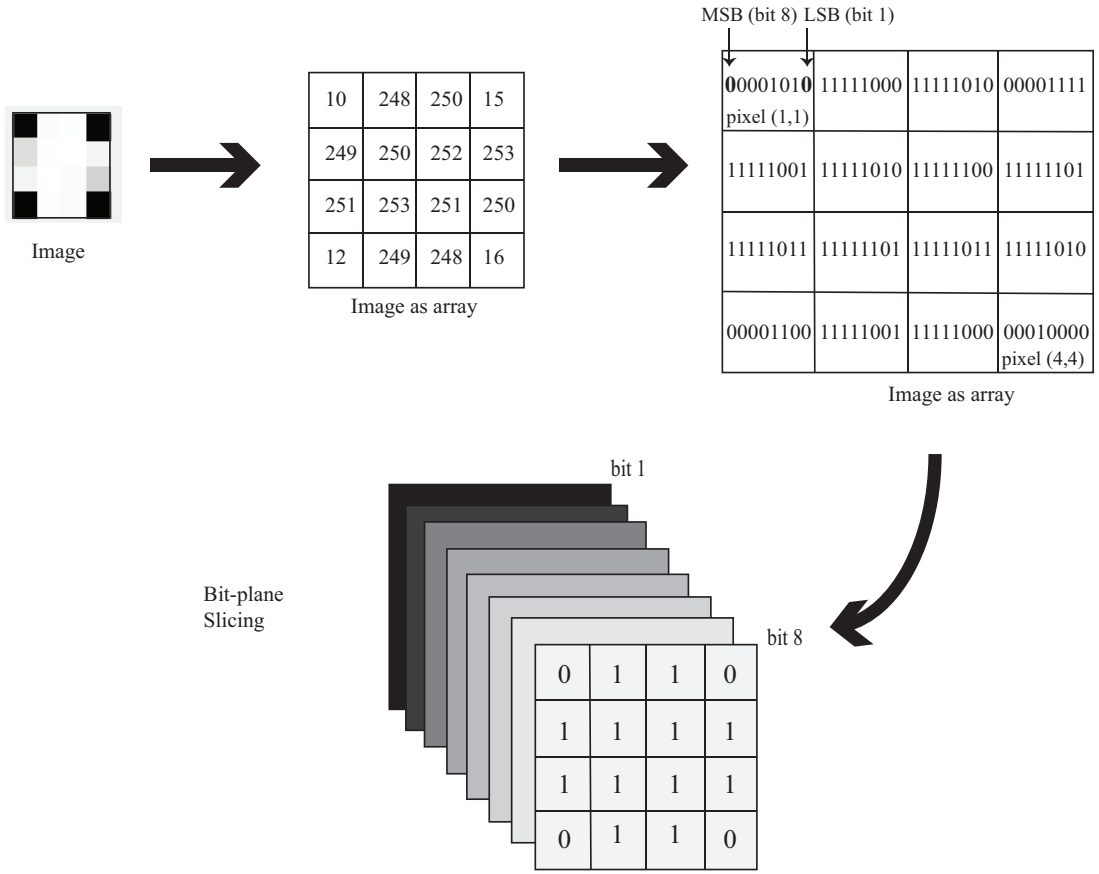
00001010**0** pixel (1,1)	11111000	11111010	00001111
11111001	11111010	11111100	11111101
11111011	11111101	11111011	11111010
00001100	11111001	11111000	00010000 pixel (4,4)

Image as array

Steganography in Digital Images

Why Steganography in Digital Images?

- A gray scale image = A 2-D array of 8-bit values
- Each pixel has 8-bits. i.e, pixel values$\in [0, 255]$

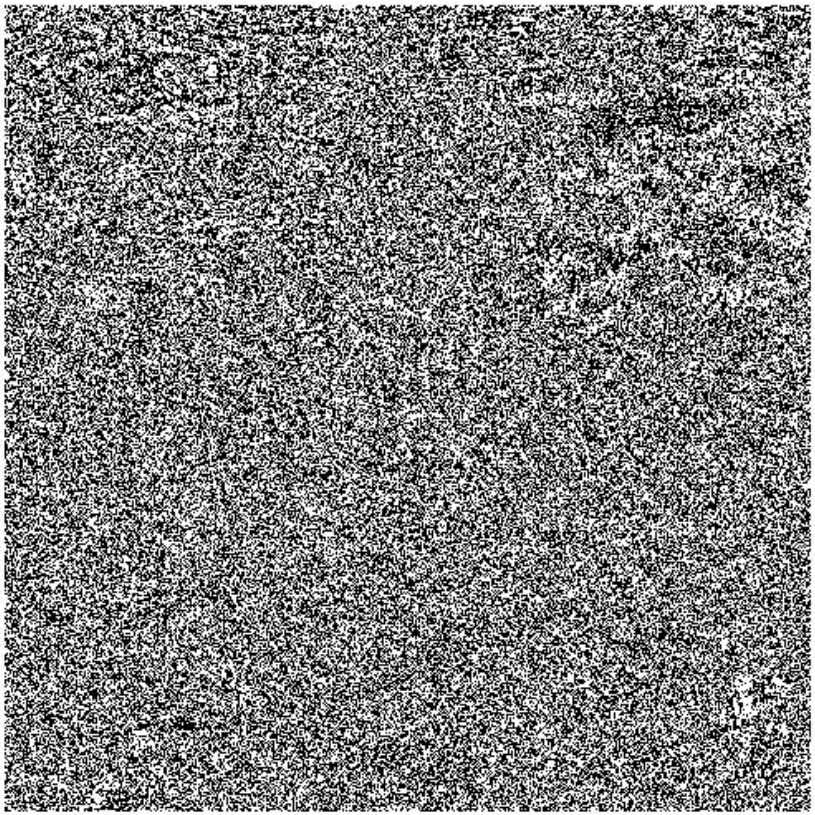


Bit-plane slicing of an image[1]

Original Image

[1] Image src: BOSSBase

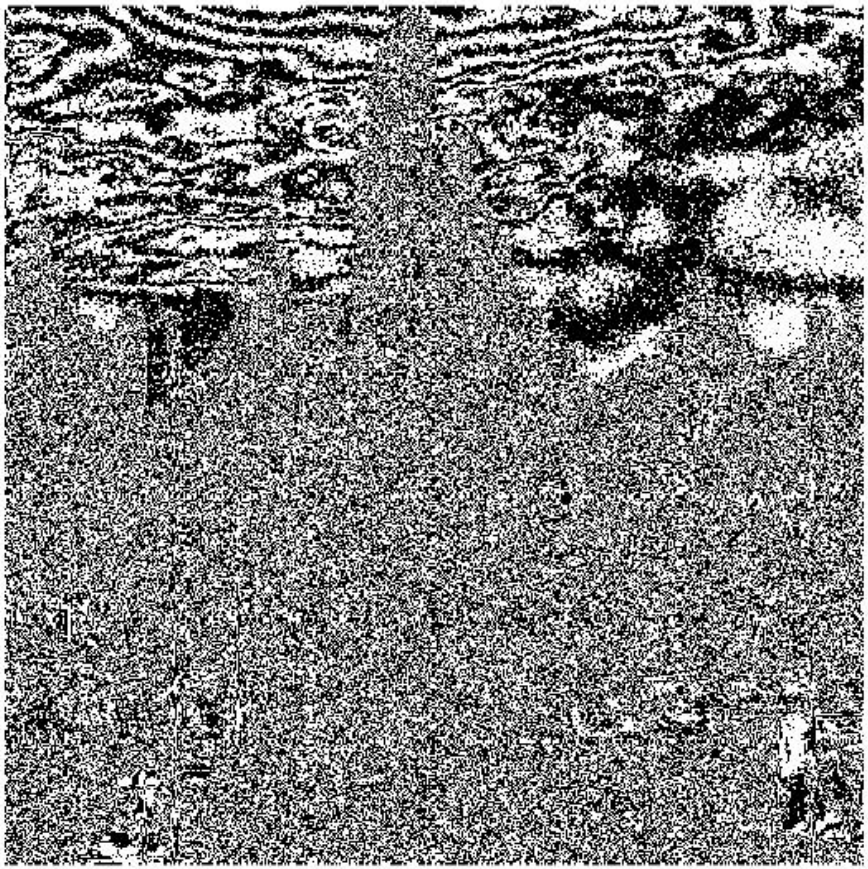
Bit-plane slicing of an image[1]



Bit Plane 1

[1] Image src: BOSSBase

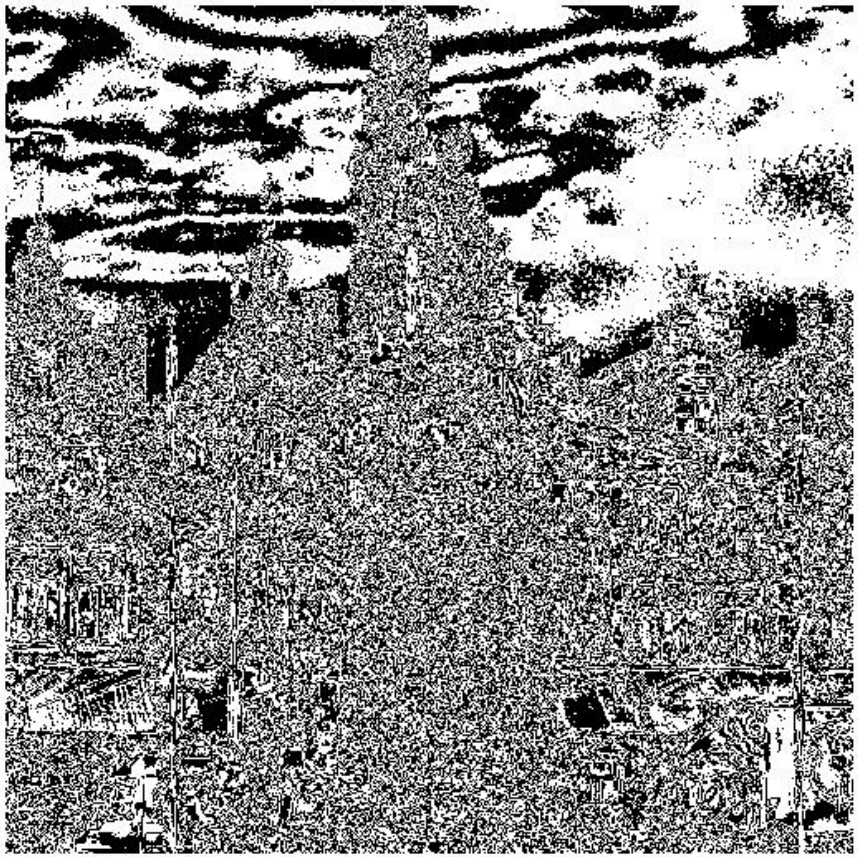
Bit-plane slicing of an image[1]



Bit Plane 2

[1] Image src: BOSSBase

Bit-plane slicing of an image[1]



Bit Plane 3

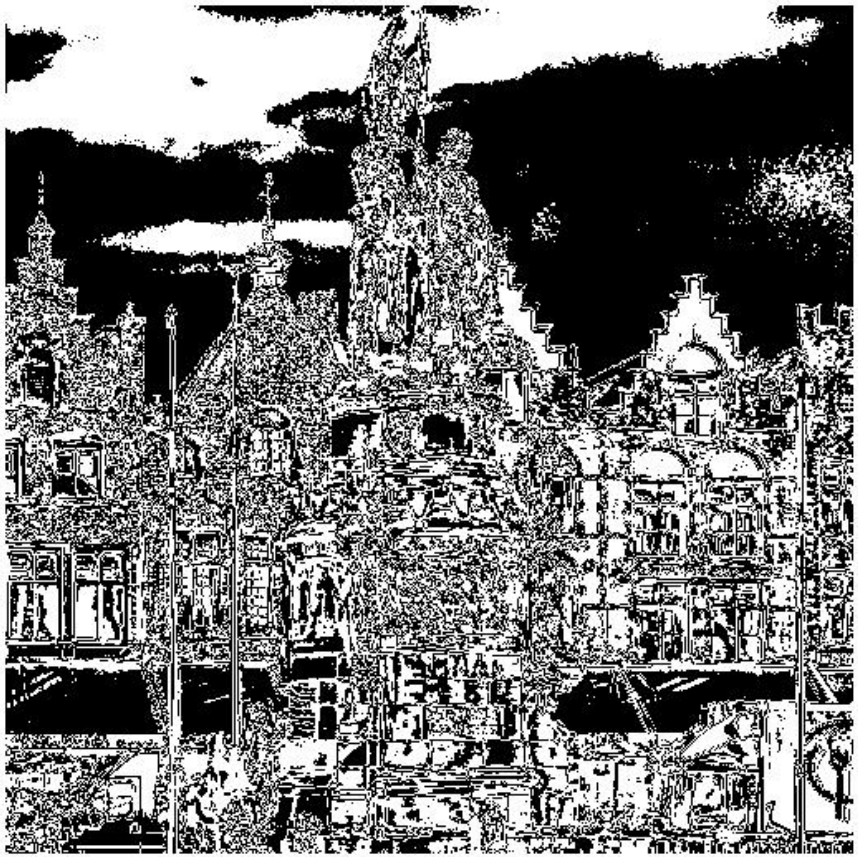
[1] Image src: BOSSBase

Bit-plane slicing of an image[1]

Bit Plane 4

[1] Image src: BOSSBase

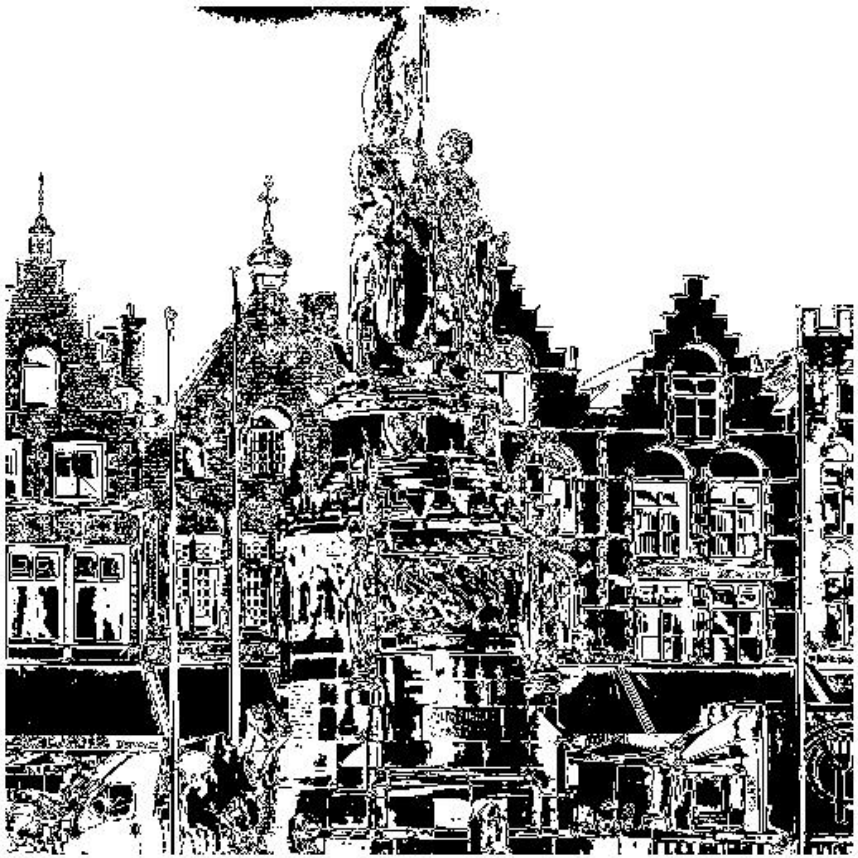
Bit-plane slicing of an image[1]



Bit Plane 5

[1] Image src: BOSSBase

Bit-plane slicing of an image[1]



Bit Plane 6

[1] Image src: BOSSBase

Bit-plane slicing of an image[1]

Bit Plane 7

[1] Image src: BOSSBase

Bit-plane slicing of an image[1]

Bit Plane 8

[1] Image src: BOSSBase

Construction of image with bit-planes

Bit Plane 8

Construction of image with bit-planes

Image with bit planes 7 and 8

Construction of image with bit-planes

Image with bit planes 6,7 and 8

Construction of image with bit-planes

Image with bit planes 5,6,7 and 8

Construction of image with bit-planes

Image with bit planes 4,5,6,7 and8

Construction of image with bit-planes

Image with bit planes 3,4,5,6,7 and 8

Construction of image with bit-planes

Image with bit planes 2,3,4,5,6,7 and 8

Construction of image with bit-planes

Image with all bit planes

Why Steganography in images?

- **Observation**
  1. Image can be represented with only 4-bit planes nicely $\Longrightarrow$ **large amount of redundant bits**

Why Steganography in images?

- **Observation**
  1. Image can be represented with only 4-bit planes nicely $\Longrightarrow$ **large amount of redundant bits**
  2. Redundant bits can be utilized to hide secret message

Why Steganography in images?

- **Observation**
  1. Image can be represented with only 4-bit planes nicely $\Longrightarrow$ **large amount of redundant bits**
  2. Redundant bits can be utilized to hide secret message
- images frequently used over Internet

Why Steganography in images?

- **Observation**
  1. Image can be represented with only 4-bit planes nicely $\Longrightarrow$ **large amount of redundant bits**
  2. Redundant bits can be utilized to hide secret message
- images frequently used over Internet
- HVS cannot differentiate between normal and stego image

Steganography (How?)

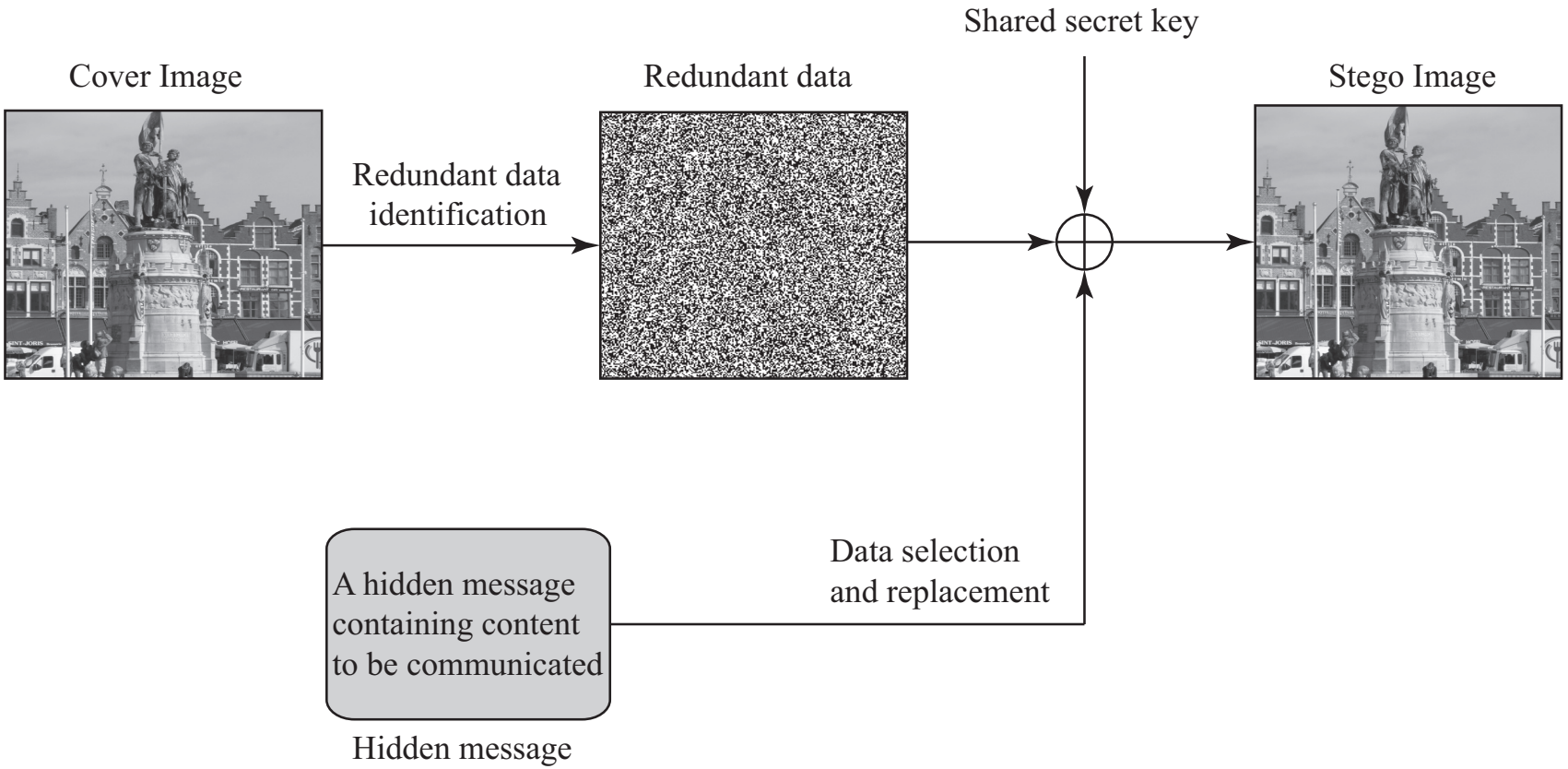


Figure: Steganography process in digital image

N. Provos and P. Honeyman, "Hide and seek: an introduction to steganography," in IEEE Security & Privacy, vol. 1, no. 3, pp. 32-44, May-June 2003.

Types of image steganography

Image steganography can be divided into the two groups:

1. **Spatial domain**- Modifies pixel values of the image
2. **Transform domain**- transforms image (eg., DCT) then embeds in transformed image

Spatial Domain schemes

- Some of the spatial domain methods are:
  1. LSB embedding
  2. HUGO
  3. WOW
  4. S-Uniward

LSB embedding

- simple steganography approach
- Make use of LSB of image pixels
- Intuition: replacing MSB with secret message will have noticable effect
- Replacing LSB produces near duplicate color, which is not noticable by human eye

J. Mielikainen, LSB matching revisited, IEEE Signal Processing Letters, vol. 13, no. 5, pp. 285-287, 2006

Example: LSB embedding

a) Cover Image

Example: LSB embedding

a) Cover Image

'In 1966, an inventive and impromptu steganographic method enabled a prisoner of war, Commander Jeremiah Denton,
to secretly communicate one word when he was forced by his Vietnamese captors to give an interview on TV.
Knowing that he could not say anything critical of his captors, as he spoke, he blinked his eyes in Morse code,
spelling out T-O-R-T-U-R-E.
In 1857, Brewster proposed a very ingenious technique that was actually used in several wars in the nineteenth
and twentieth centuries. The idea is to shrink the message so much that it starts resembling specks of dirt but
can still be read under high magnification. The technological obstacles to use of this idea in practice were
overcome by the French photographer Dragon, who developed technology for shrinking text to microscopic dimensions.
Such small objects could be easily hidden in nostrils, ears, or under fingernails. In World War I, the Germans used
such "microdots" hidden in corners of postcards slit open with a knife and resealed with starch. The modern twentieth
-century microdots could hold up to one page of text and even contain photographs. The Allies discovered the usage of
microdots in 1941. A modern version of the concept of the microdot was recently proposed for hiding information in DNA
for the purpose of tagging important genetic material. Microdots in the form of dust were also recently proposed to
identify car parts. Perhaps the best-known form of steganography is writing with invisible ink. The first invisible
inks were organic liquids, such as milk, urine, vinegar, diluted honey, or sugar solution. Messages written with such
ink were invisible once the paper had dried. To make them perceptible, the letter was simply heated up above a candle.
Later, more sophisticated versions were invented by replacing the message-extraction algorithmwith safer alternatives,
such as using ultraviolet light.'

b) Message text

Example: LSB embedding

a) Cover Image

'In 1966, an inventive and impromptu steganographic method enabled a prisoner of war,
Commander Jeremiah Denton,
to secretly communicate one word when he was forced by his Vietnamese captors to give an
interview on TV.
Knowing that he could not say anything critical of his captors, as he spoke, he blinked
his eyes in Morse code,
spelling out T-O-R-T-U-R-E.
In 1857, Brewster proposed a very ingenious technique that was actually used in several
wars in the nineteenth
and twentieth centuries. The idea is to shrink the message so much that it starts
resembling specks of dirt but
can still be read under high magnification. The technological obstacles to use of this
idea in practice were
overcome by the French photographer Dragon, who developed technology for shrinking text
to microscopic dimensions.
Such small objects could be easily hidden in nostrils, ears, or under fingernails. In
World War I, the Germans used
such "microdots" hidden in corners of postcards slit open with a knife and resealed
with starch. The modern twentieth
-century microdots could hold up to one page of text and even contain photographs. The
Allies discovered the usage of
microdots in 1941. A modern version of the concept of the microdot was recently proposed
for hiding information in DNA
for the purpose of tagging important genetic material. Microdots in the form of dust
were also recently proposed to
identify car parts. Perhaps the best-known form of steganography is writing with
invisible ink. The first invisible
inks were organic liquids, such as milk, urine, vinegar, diluted honey, or sugar
solution. Messages written with such
ink were invisible once the paper had dried. To make them perceptible, the letter was
simply heated up above a candle.
Later, more sophisticated versions were invented by replacing the message-extraction
algorithmwith safer alternatives,
such as using ultraviolet light.'

b) Message text

c) Stego Image

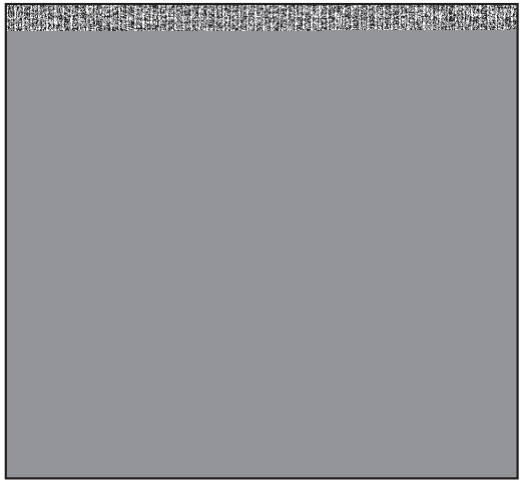
Example: LSB embedding

a) Cover Image

'In 1966, an inventive and impromptu steganographic method enabled a prisoner of war,
Commander Jeremiah Denton,
to secretly communicate one word when he was forced by his Vietnamese captors to give an
interview on TV.
Knowing that he could not say anything critical of his captors, as he spoke, he blinked
his eyes in Morse code,
spelling out T-O-R-T-U-R-E.
In 1857, Brewster proposed a very ingenious technique that was actually used in several
wars in the nineteenth
and twentieth centuries. The idea is to shrink the message so much that it starts
resembling specks of dirt but
can still be read under high magnification. The technological obstacles to use of this
idea in practice were
overcome by the French photographer Dragon, who developed technology for shrinking text
to microscopic dimensions.
Such small objects could be easily hidden in nostrils, ears, or under fingernails. In
World War I, the Germans used
 such "microdots" hidden in corners of postcards slit open with a knife and resealed
with starch. The modern twentieth
-century microdots could hold up to one page of text and even contain photographs. The
Allies discovered the usage of
microdots in 1941. A modern version of the concept of the microdot was recently proposed
for hiding information in DNA
for the purpose of tagging important genetic material. Microdots in the form of dust
were also recently proposed to
identify car parts. Perhaps the best-known form of steganography is writing with
invisible ink. The first invisible
inks were organic liquids, such as milk, urine, vinegar, diluted honey, or sugar
solution. Messages written with such
ink were invisible once the paper had dried. To make them perceptible, the letter was
simply heated up above a candle.
Later, more sophisticated versions were invented by replacing the message-extraction
algorithmwith safer alternatives,
such as using ultraviolet light.'

b) Message text

c) Stego Image



d) Stego noise

Content-Adaptive Steganography (Modern)

- Minimize additive distortion fuction for embedding:
  $D(X, Y) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \rho_{ij}(X_{ij}, Y_{ij}) \, |Y_{ij} - X_{ij}|$
  ρ_{ij} is cost of changing pixel X_{ij} to Y_{ij}
- Distortion (Cost) function indicates where to embed
- Assign the embedding cost to be high to smooth areas and low cost to textured or noisy areas
- e.g., S-UNIWARD, HUGO, WOW, etc.

Example: WOW

Cover Image

V. Holub and J. Fridrich, "Designing steganographic distortion using directional filters," 2012 IEEE International Workshop on Information Forensics and Security (WIFS), Tenerife, 2012, pp. 234-239

Example: WOW

Cover Image

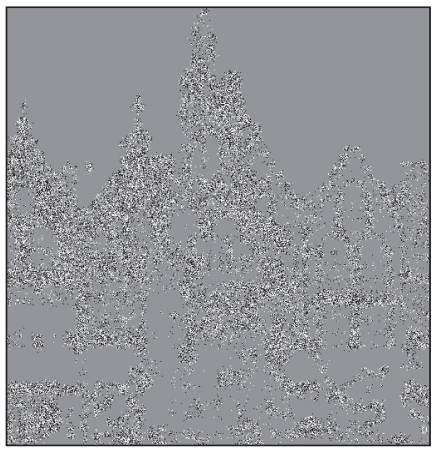
Stego Image

V. Holub and J. Fridrich, "Designing steganographic distortion using directional filters," 2012 IEEE International Workshop on Information Forensics and Security (WIFS), Tenerife, 2012, pp. 234-239

Example: WOW

Cover Image

Stego Image



Stego Noise

V. Holub and J. Fridrich, "Designing steganographic distortion using directional filters," 2012 IEEE International Workshop on Information Forensics and Security (WIFS), Tenerife, 2012, pp. 234-239

Steganalysis

Steganalysis

- Analyzing images with intent of finding trace of hidden data
- Given an image. Is it Stego or Cover ? (Binary classification)
- Works in three steps:
  1. Preprocessing
  2. Feature extraction
  3. Classification

Related Works

Related Works

- Related works can be categorized as
- i) Hand-crafted feature based and ii) Deep feature based

Pevn et al. (SPAM)

- Assumption: steganographic embedding alters the pixel dependencies of the image
- modeled deviation between adjacent pixels in images
- First, filter $[-1, 1]$ is used to suppress the image content and expose the embedding noise, results in noise residual
- Using Markov chain, dependencies between neighboring pixels in residual image is modeled
- The resultant transition probability matrix is used as feature vector
- These feature vectors are then used to train SVM for classification

---

T. Pevn, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," IEEE Transactions on information Forensics and Security, vol. 5, no. 2, pp. 215-224, 2010

Fridrich and Kodovsky (SRM)

- SRM computed residual through 39 different high-pass filters
- An ensemble classifier is trained on these features
- EC is consist of several binary classifiers called base learners
- Fisher Linear Discriminants are used as base learner
- EC reaches its decision by majority voting.

J. Fridrich and J. Kodovsky, "Rich Models for steganalysis of digital images," IEEE Transactions on information Forensics and Security, vol. 7, no. 3, pp. 868-882, 2012

Some of SRM filters

$$spam_{14h,v} = \begin{pmatrix} -1 & +1 \end{pmatrix} \qquad minimax_{22h,v} = \begin{pmatrix} +1 & -1 & +1 \end{pmatrix}$$

$$minimax_{41} = \begin{pmatrix} 0 & +1 & 0 \\ +1 & -1 & +1 \\ 0 & +1 & 0 \end{pmatrix} \qquad KV = \frac{1}{12} \begin{pmatrix} -1 & 2 & -2 & 2 & -1 \\ 2 & -6 & 8 & -6 & 2 \\ -2 & 8 & -12 & 8 & -2 \\ 2 & -6 & 8 & -6 & 2 \\ -1 & 2 & -2 & 2 & -1 \end{pmatrix}$$

Problem with Hand-crafted feature based schemes

- Requires well designed features
- Hand-crafted feature is heavily dependent on user expertise
- Hand-crafted features extraction is time-consuming

Solution of Problems using CNN

- CNN (Convolutional Neural Network) are known to be very good feature extractors
- Features can be learned by the CNN
- Feature extraction and classification can be done using a single network

Qian et al. (GNCNN)

- Preporcessing using KV filter
- Features are extracted using CNN having 4 conv
- 3 fc with ReLU and a softmax layer is used for classification
- Layers used Gaussian activation followed by pooling
- 4% of improvement over SRM

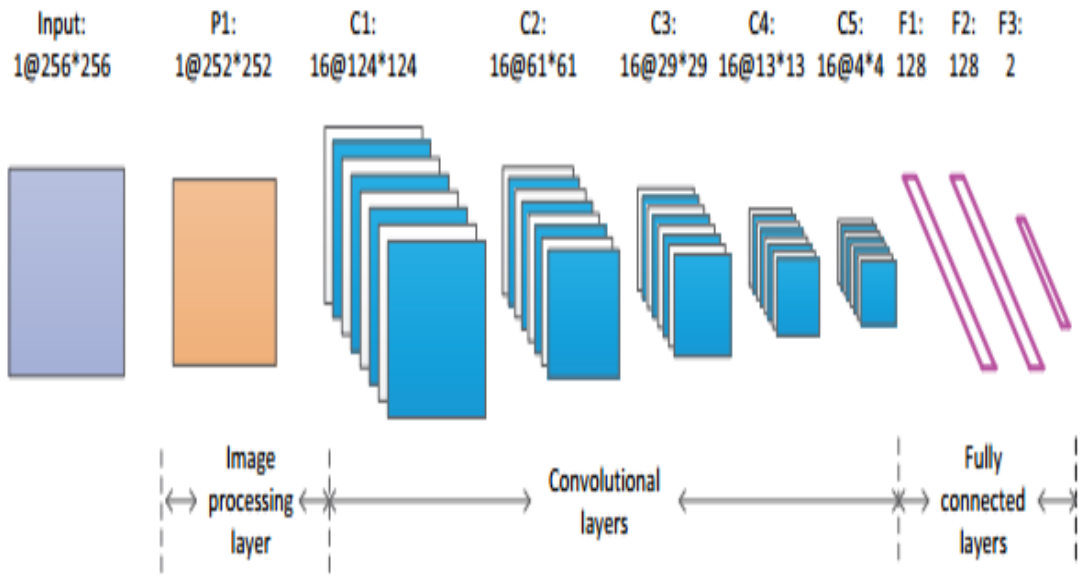


Figure: Qian Network

Y. Qian, J.Dong, W. Wang, and T. Tan, "Deep learning for steganalysis via convolutional neural networks," Proc. SPIE 9409, Media Watermarking, Security, and Forensics 2015 vol. 9409

Xu et al.

- HPF used for preprocessing
- ABS is used preserve the negative values
- BN is used to prevent CNN from falling into local minima

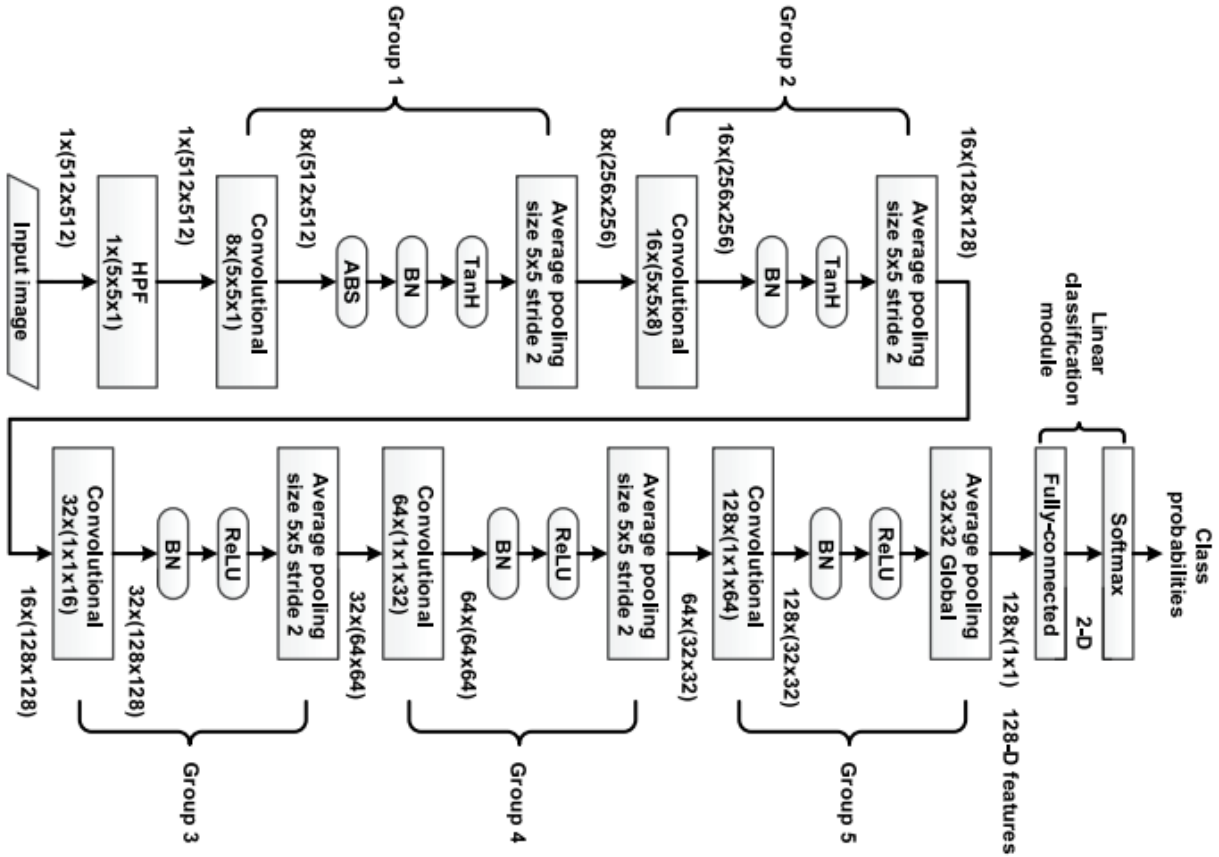


Figure: Xu et al. network

G. Xu, H. Z. Wu and Y. Q. Shi, "Structural Design of Convolutional Neural Networks for Steganalysis," in IEEE Signal Processing Letters, vol. 23, no. 5, pp. 708-712, May 2016.

Qian et al.

- Presented a transfer learning for steganalysis
- Observation: The lower the payload for a steganographic algo, the harder the training of CNN
- Sometimes CNN fails to converge
- GNCNN is trained for detecting a steganographic algo with a high payload
- Feature representation is transfered to to improve the learning of features for the same steganographic algo with low payload
- Reported better performance compared to SRM in detecting the WOW algorithm

Y. Qian, J.Dong, W. Wang, and T. Tan, "Learning and transferring representations for image steganalysis using convolutional neural network," in 2016 IEEE International Conference on Image Processing (ICIP), Sept 2016, pp. 2752–2756

Limitation of existing works

1) Existing schemes used a fixed HPF to compute noise residual

A fixed kernel may not always be optimal for extracting noise residual

2) Some methods used a deeper network for classification, which may lead to vanishing gradient problem

3) Some methods used pooling which may ruin the stego noise

Thank You