Deep Learning Based Image Steganalysis in Spatial domain

A Deep CNN framework for steganalysis

contents:

- Introduction to steganography
- Steganography vs cryptography
- Basics of steganography
- Applications of steganography
- Steganalysis
- ML based steganalysis
- DL based steganalysis
- Architecture
- Results
- Applications and Conclusion
- Future directions

Introduction:

- Steganography: literal meaning is hidden writing

  More precisely it is defined as the technique of embedding messages in an imperceptible way in media (like text, images, audio, video etc) so that the resulting stego is similar the media used to hide it, statistically and visually.

- NOTE: anyone with sufficient knowledge and tools can analyse the stego and may recover the hidden data
- NOTE: steganography is different from cryptography

Steganography vs cryptography:

- Steganography hides the message in some cover whereas cryptography transforms the message in unintelligible data
- Output of steganography is stego data which contains hidden message which is imperceptible and hides communication.
- In case of cryptography the encrypted data is visible and available but it relies on the hardness of the underlying cryptosystem for security
- Fundamentally steganalysis tries to hide any sign of communication being taking place altogether.

example:

- Steganography:

  Message + cover image → stego image(visibly and statistically same)

  Hello + →

- cryptography:

  Encrypt(Message) → ciphertext (unintelligible text)

  Encrypt(hello) → igopt

Basics of steganography

- Terms and definitions
- Basic methods
- Adaptive methods

- Cover image: it is the image used to hide the message. It can be of any format but those formats are preferred in which lossless compression is done so jpeg is the least preferred and bitmap (bmp) is most preferred.
- Stego image: It is the image obtained after hiding the message data in the cover image
- Stego image is required to be imperceptible to human eyes.
- LSB: least significant bit. For an image with 8 bits per pixel the last bit represents the LSB.
- LSB plane: It is the 2D matrix formed by the LSBs of all the pixels of MxN 2D image

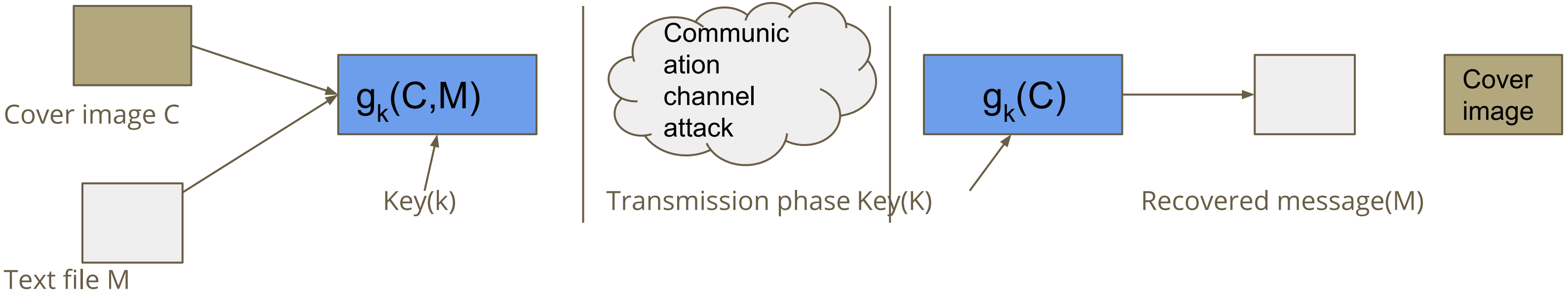
Steganography methods:

Basic steganography:

C be the cover image

M be the message to hide

K be the key



Steganography methods:

Steganography can be based on following image characteristics :

- Steganography exploiting the image format
- Steganography in the image spatial domain
- Steganography in the image frequency domain
- Adaptive steganography

NOTE: we are following only the area of spatial domain

Steganography in image spatial domain

- In spatial domain we use the pixels of the cover image to hide the message.
- There are various techniques for spatial domain:
    - LSB based methods
        - LSB matching
        - LSB replacement
    - Color palette based methods
        - In this method also LSBs are modified based on their positions in the color palette index
    - Histogram based methods(uses correlation between adjacent pixels)
        - HS: histogram shifting
        - DE: difference expansion
        - PEHS: prediction error histogram shifting

LSB based methods:

- LSB based methods are rely on the fact that if LSB of a pixel value is changed, the perceptual quality of the pixel does not change significantly
- To understand this consider a grayscale image where each pixel is represented by 8 bit binary value
- With 8 bits 256 gray levels can be represented
- If we want to embed 1 bit msg in LSBs of the image pixels then,

  If a pixel value is 212 which is 11010100

  If Msg bit is 1 then

  after embedding, the pixel value will become 213 which is 11010101

- it can be seen that a change in pixel value from 212 to 213 does not affect the pixel visibly

  Since gray values so close are visibly indistinguishable to naked eyes

Steganalysis

- Introduction
- Basics of Steganalysis
- ML Based Steganalysis
- Deep Learning Based Steganalysis

Introduction:

- Steganalysis is the art of detecting hidden messages in a media.
- Why steganalysis works:
    - Natural images have some level of spatial correlation among the pixels
    - Pixels have some statistical correlations
    - Devices used for taking images do some processing on images which also creates some artifacts
    - Embedding alters these statistical properties which can be exploited for steganalysis

Ex: flip-embed/LSB-matching creates POVs which leads to chi-squared attack

Basics of Steganalysis:

Two categories of steganalysis:

- **Blind Steganalysis:** No knowledge of algorithm or embedding rate.
  - Very difficult and need sophisticated machine learning tools
- **Selection Channel Aware:** Knowledge of specific algorithm and embedding rate used is available
  - It is more reliable as compared to blind steganalysis

ML Based Steganalysis:

SRM: Spatial Rich Models

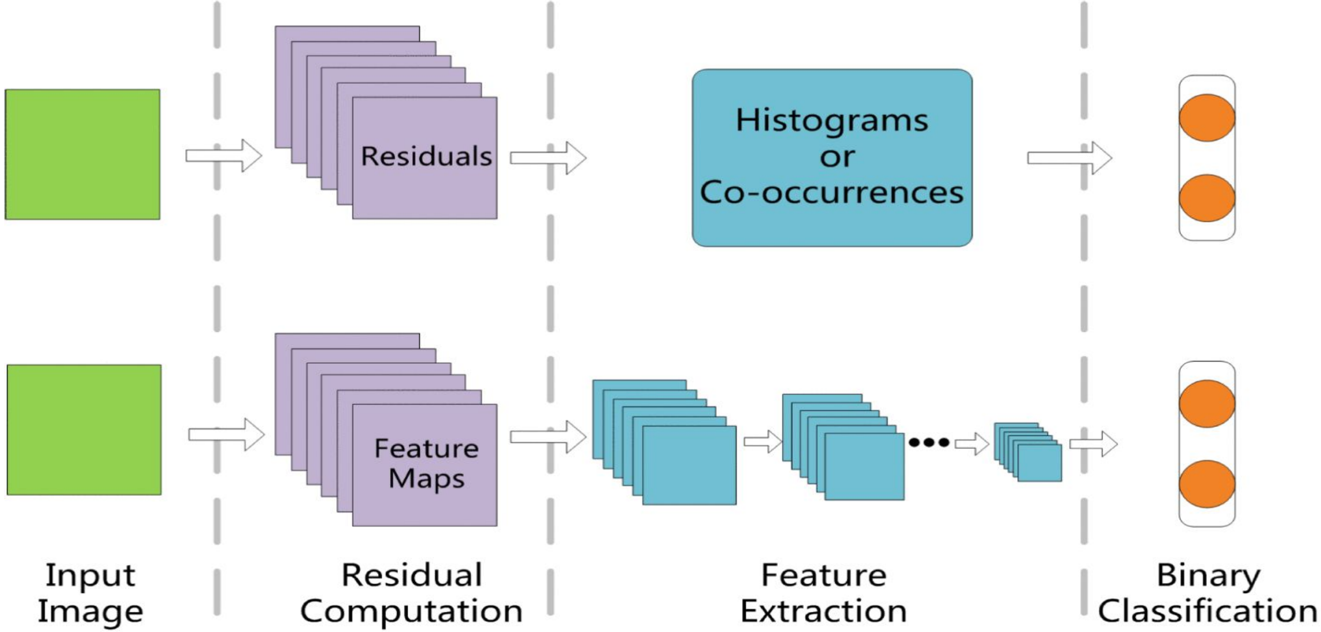
- SRMs are constructed by by assembling many diverse submodels which picks up various noise residuals caused by embeddings

maxSRM: Selection Channel Aware variant of SRM

DL Based Steganalysis:

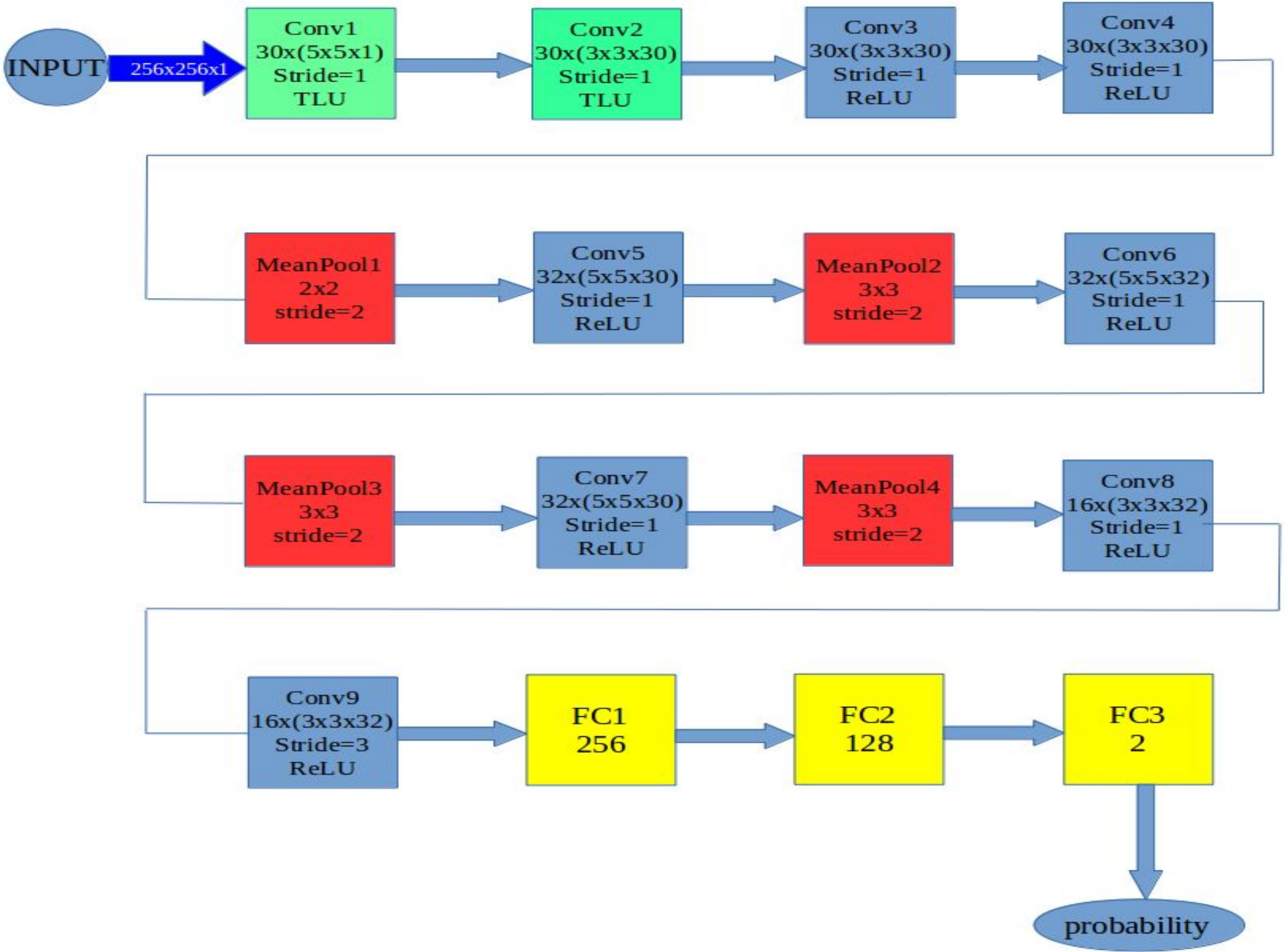
- Deep learning based models are efficient to learn hierarchical representations
- Increased depth of model leads to better feature representations

Statistical methods Vs CNN



Architecture:

- 12 Layer CNN model is used with following configurations
    - Input size 256*256 single channel image(grayscale)
    - 9 conv layers with relu activation
    - 2 fully connected layers with 0.5 dropout
    - 1 fully connected classification layer with softmax activation
    - From layers 3 to 11 relu activation is applied
    - **For layer 1 and 2 modified relu is applied**
    - SGD optimization is used with default parameters
    - First convolution layer is initialized with 30 custom filters
    - Initialization with custom filters makes the model converge faster compared to random initialization


INPUT
256x256x1
Conv1
30x(5x5x1)
Stride=1
TLU
Conv2
30x(3x3x30)
Stride=1
TLU
Conv3
30x(3x3x30)
Stride=1
ReLU
Conv4
30x(3x3x30)
Stride=1
ReLU
MeanPool1
2x2
stride=2
Conv5
32x(5x5x30)
Stride=1
ReLU
MeanPool2
3x3
stride=2
Conv6
32x(5x5x32)
Stride=1
ReLU
MeanPool3
3x3
stride=2
Conv7
32x(5x5x30)
Stride=1
ReLU
MeanPool4
3x3
stride=2
Conv8
16x(3x3x32)
Stride=1
ReLU
Conv9
16x(3x3x32)
Stride=3
ReLU
FC1
256
FC2
128
FC3
2
probability

Architecture:

- This model is trained on NVIDIA GTX 1080 titan GPU
- Dataset used is BOSSbase image database of 10000 grayscale images
- Images were cropped to 256*256
- Stego image dataset is created using original database
- Two classes of dataset each containing 10 000 images
- 10% of total images were used as validation set

Our contribution:

- We have used modified relu which is mentioned as truncated relu in TLU [1]
- We have experimented with various combinations of TLU with different clipping values for min and max and at different layers
- TLU is modified form of relu:

  Defined as : $f(x) = T$ *if* $x > T$,

  $-T$ *if* $x < -T$

  x otherwise

- TLU when applied to first two consecutive layers was found to be the best combination

Results:

- Applying TLU with cutoff { -7, 7 } at first convolution layer proved to be very effective in suppressing the image contents
- Since embedding causes the value of a pixel to change by at most 1
- Applying TLU with cutoff { -1, 1 } on convolutional layer 2 helps picking up artifacts introduced due to embedding
- Effectively TLU at layer 2 enhances the embedding noise which we are trying to learn
- **When compared to the model having TLU { -1 , 1 } activation applied to layer 1 only, our model reduced the error by approx. 2% but the overall performance is still to be optimized**

Applications:

- One of the possible area of application can be the medical image diagnosis, where it is required to pick up very low SNR features in the image
- For Multimedia forensics

Find the implementation of the model here:

https://github.com/aminfazy/DL_steganalysis.git

Future work:

- More deeper models need to be tested for increasing the accuracy
- There is much scope of testing different activation functions which can enhance the accuracy
- Using more sophisticated kernel initializers for first layer, which will reduce the training time substantially

References :

1. Jian Ye, Jiangqun Ni, Yang Yi. ***Deep Learning Hierarchical Representations for Image Steganalysis.*** IEEE Transactions on Information Forensics and Security Volume: 12, Issue: 11, Nov. 2017
2. Fangjun Huang, Jiwu Huang, Yun-Qing Shi, ***New Framework For Reversible Data Hiding In Encrypted Domain***, IEEE Transactions on Information Forensics And Security, Dec 2016.
3. Yun-Qing Shi, Xiaolong Li, Xinpeng Zhang, Hao-Tian Wu, Bin Ma, ***Reversible Data Hiding: Advances in the Past Two Decades***, IEEE special section on latest advances and emerging application of data hiding, July 2016.
4. Andreas Westfeld, Andreas Pfitzmann, ***Attacks On Steganographic Systems: breaking the steganographic utilities EzStego, Jsteg, Steganos, and S-Tools and some lessons learned,*** Springer-Verlag, Berlin Heidelberg 2000.
5. Der-Chyuan Lou, Chen-Hao Hu, ***LSB Steganographic Methods Based on Reversible Histogram Transformation Function For Resisting Statistical Steganalysis***, Information Sciences, Elsevier, 2012.
6. Jessica Fridrich, Miroslav Goljan, Rui Du, ***Reliable Detection of LSB Steganography in Color and Grayscale Images***, Multimedia And Security, ACM 2001

Thank you