MEDIA SECURITY PROJECT REPORT

Detection of LSB Replacement and LSB Matching Steganography Using Gray Level Run Length Matrix

1. A brief of Algorithm

A method to detect the typical LSB (Least Significant Bit) embedding and the LSB matching steganography methods applied to grayscale images. The proposed method determines the changes made to some selected features extracted from the gray level run length matrix. The gray level run length matrix (G_{RL}) is an $N_g \times N_r$ matrix, whose (i,j) element gives the number of times a gray level i-1 appears in the image with run length j . N_g is the number of possible gray levels and N_r is the length of the longest run. It is shown that the run length characteristics can significantly be affected by the embedded message bits, so can be employed as a measure that is quite sensitive to the arrangements of the image pixel values.

Features used for steganalysis method:

- GLNU (Gray Level Non-Uniformity)
- RLN (Run Length Non-Uniformity)
- Run Percentage/ Number of possible runs
- Maximum length of run

To design the steganalyzer, four images are constructed from the main image. The first image is constructed from the first LSB bit plane, the second image from the first and second LSB bit planes and so on. Then, the gray level runs length matrix is computed for each of these four images. In addition, for the first and second images, the gray level run length matrix is computed in 90^{o} direction. So, totally, six gray level run length matrices are computed.

The extracted features are examined by a nonlinear SVM (Support Vector Machine) classifier with quadratic kernel that can distinguish between stego and clean images.

2. Experiment Setting

- Database of 300 gray scale images in JPG format.
- Images resized to 256×256.
- Training the classifier - 200 images from the database were randomly divided into five disjoint, 40-image groups for five different embedding rates of 20%, 40%, 60%, 80% and 100%.
- LSB embedding and the LSB matching algorithms used for embedding.
- Testing Steganalyzer - the remaining 100 images from the database were used. LSB matching and embedding were applied to this collection at the embedding rates of 40%, 60%, 80% and 100% and the classifier's outputs were evaluated.
- The result of test phase is reported in table I. In this table, the false alarm of steganalyzer is obtained by testing the classifier with 100 clean images.

3. Results

Results obtained for the experiment setting given above are shown in table I.

Table I : Performance of Steganalyzer

Embedding Rate	Detection Rate ±1 LSB	Detection Rate LSB Embedding
100%	85	80
80%	80	75
60%	90	70
40%	70	65
Accuracy of Detecting Clean Signal	74	61

- The experiment was repeated by training the classifier using 200 images from the database divided into five disjoint, 20-image groups for four different embedding rates of 20%, 40%, 60%, 80% and 120-image groups for 100% embedding rate. Results are shown in table II.

Table II : Performance of Steganalyzer

Embedding Rate	Detection Rate ±1 LSB	Detection Rate LSB Embedding
100%	92.5	87.5
80%	85	75
60%	90	75
40%	70	65
Accuracy of Detecting Clean Signal	71	64

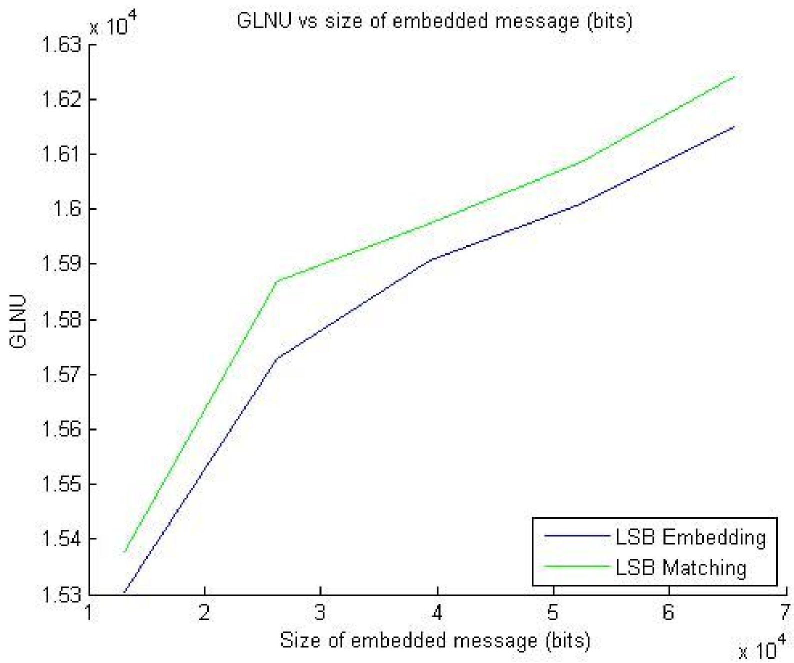

Fig 1 : Variation of GLNU as a function of embedded message size

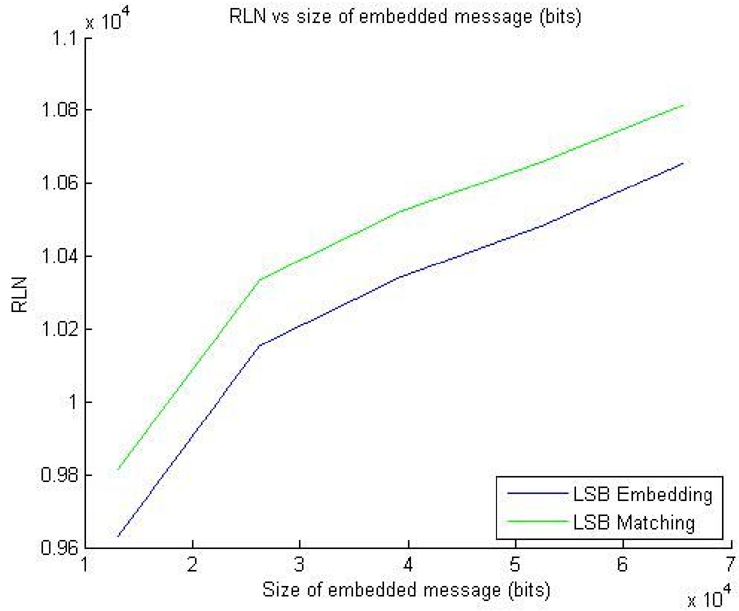

Fig 2 : Variation of RLN as a function of embedded message size

We used a multilayer feed-forward backpropagation neural network to plot the ROC. The inputs of the neural network are merely the vectors obtained from the training data.

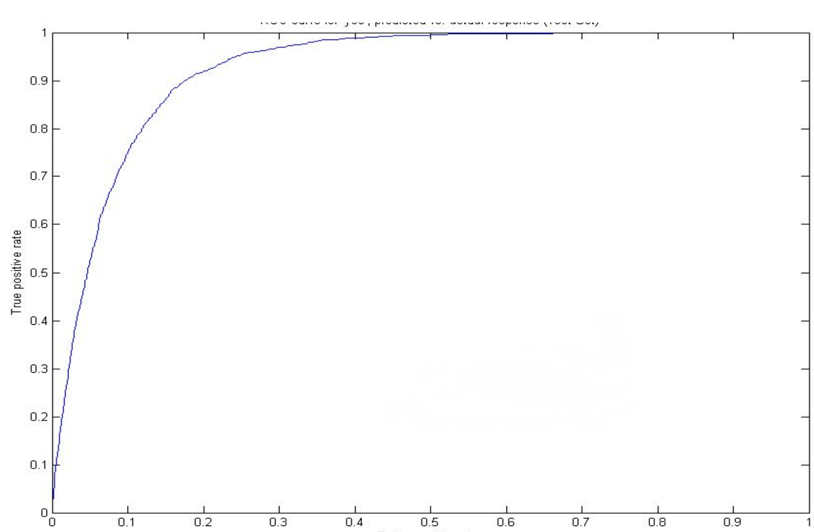

Fig 3 : LSB matching

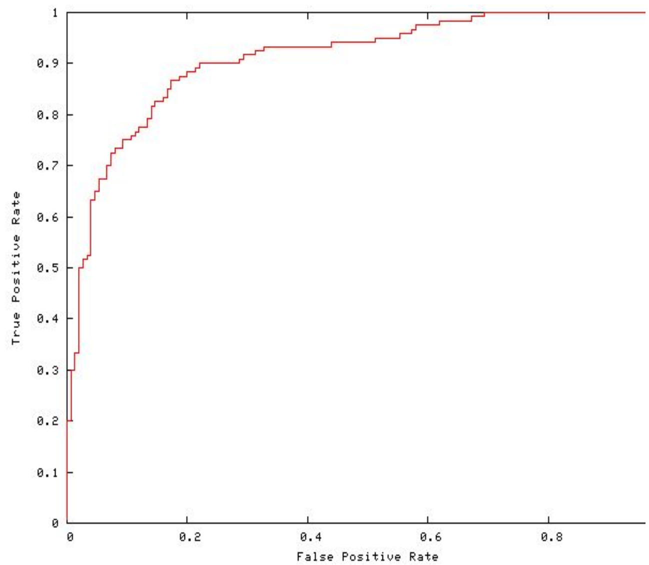

Fig 4 : LSB embedding

Comparison of ROC curves for LSB matching algorithm. The different curves stand for the cases in which:

- All features (s1)

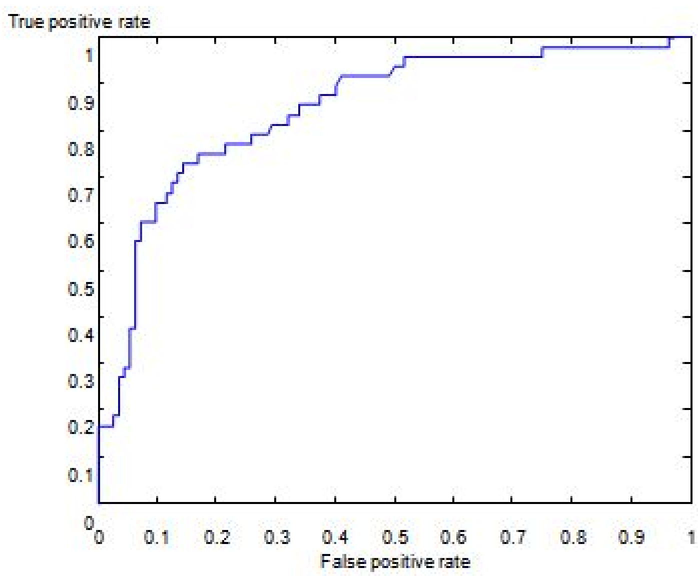

Fig 5 : ROC curve for all features (s1)

- All but 1st feature (s2)

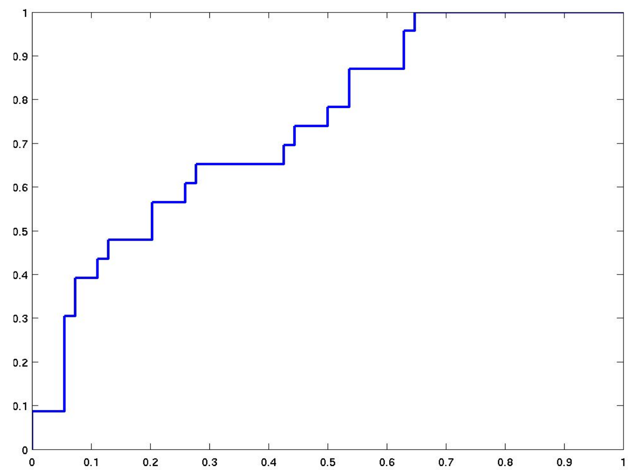

Fig 6 : ROC curve for s2

- All but 1st and 3rd features (s3)

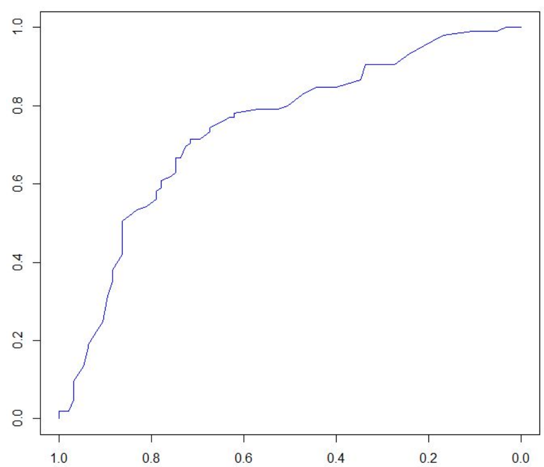

Fig 7 : ROC curve for s3

- All but 1st, 2nd, and 3rd features (s4) are employed

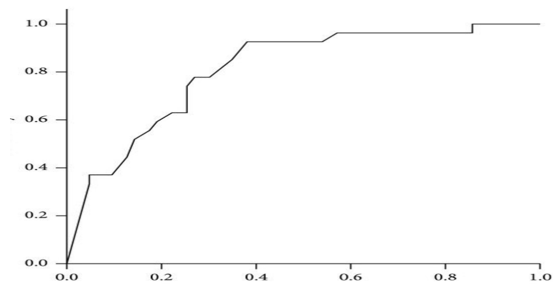

Fig 8 : ROC curve for s4

4. Analysis of Results

- Proposed steganalyzer yields a higher performance in discrimination between stego and clean images at various embedding rates.
- The performance of steganalyzer for the LSB matching algorithm is better than LSB embedding algorithm - refer results of table I and II.
- This is because the LSB embedding only changes the LSB bit plane, while the LSB matching algorithm changes other bit planes as well. So, the LSB matching can have a greater effect on the gray level run length matrix.
- Hiding message bits in the LSB bit plane changes the length of the runs. Embedding breaks long runs into shorter runs. Thus, the number of possible lengths for runs, maximum length of run decreases and sum of the run length values increases. Henceforth RP increases after the embedding.
- Randomness of the secret message bits due to the message encryption, hiding information in the LSB bit plane decreases the uniformity of the gray levels and run length in the stego image and increases both the GLNU and the RLN.

- LSB embedding and the LSB matching algorithms increase the GLNU and the RLN with increase in embedded message size - refer Fig 1, 2
- ROC curves show that as we move from s4 to s1 case, i.e., from using only fourth feature (run percentage/total possible runs ratio) to using all the four features, number of true positives increase - refer Fig 5,6,7,8