

## Review on Image Steganalysis Using INRIA Dataset

Hanaa Mohsin Ahmed<sup>1</sup> and Halah H. Mahmoud<sup>2\*</sup>

<sup>1</sup> Computer Science Department, University of Technology, Baghdad-Iraq.

<sup>2</sup> Computer Center, University of Baghdad, Baghdad-Iraq.

\* Corresponding Author: halah@uob.edu.iq.

### Abstract

Steganography can be defined as the science of private communication while steganalysis is the science of detection of embedded messages in digital media with the use of steganography (Each of steganography and steganalysis are under much interest from media and law enforcement entities). Blind steganalysis had the attempt of differentiating steganographic images from cover images without knowing the method of steganography.

Popular data-sets are common in various areas like image processing, artificial intelligent, and security which allow the researchers to validate their approaches. INRIA dataset is one type of image dataset which include Holiday, Copydays and BIGANN evaluation datasets. This paper presents study on number of researches using INRIA dataset for image/ information retrieval and especially blind image steganalysis. All these works depends on statistical properties of image. No one use machine learning tools like deep learning especially convolution neural network to detect attack in image using INRIA dataset. [DOI: [10.22401/ANJS.21.4.13](https://doi.org/10.22401/ANJS.21.4.13)]

Keywords: image steganalysis, INRIA, image retrieval, image detection.

### Introduction

Steganography is a method of hiding secret information into seemingly innocent cover medium like a digital image, video, audio and text in such a way that existence of message is not visible [1]. It is derived from Greek words "Steganos" and "Graphy" which mean hidden and writing. The cover file with secret message hidden in it is referred as "stego" file [2].

The modern steganography algorithms are extremely secure which makes it difficult to visually detect any changes in the stego file. However, statistical properties of the cover images are modified through these steganography algorithms, which can be exploited to detect steganography. This process of identifying these statistical changes is known as steganalysis which is an extremely challenging task and demands in-depth computational analysis [3].

The significance of steganalysis approaches which are capable of a reliable detection of the existence of embedded data in images increases continuously. Steganalysis is utilized in different areas, such as forensics, cyber warfare, tracking criminal activities via the internet and obtaining evidence concerning investigations specifically in anti-social elements cases [4, 5].

Image steganalysis algorithms are basically of two types, which are: Specific and Generic [6].

#### Specific

This method is a category of image steganalysis approaches which are highly dependent on the underlying steganography method that has been utilized and are of a high rate of success in the detection of the existence of the hidden message if the message is embedded using the method for which the approaches are intended for.

#### Generic

It is a category of image steganalysis approaches which are not dependent on the underlying steganographic method that has been utilized for message embedding and guarantees good results in the detection of the existence of a hidden message that has been embedded with the use of new and/or unconventional steganography methods.

Image steganalysis approaches under both those classes are typically designed for the detection of the existence of a hidden message and the decoding of the same is considered complementary not mandatory [6].

Blind steganalysis is comprised of two processes Feature Extraction and

Classification [7]. During the training phase of any blind steganalysis, a set of features of each of Stego and Cover images are passed to a statistical classifier. The learning classifier then develops a model that maps the training set to one of the predefined class labels. In the stage of testing, the same group of properties of unknown images is fed as input to the trained classifier for deciding if these images contain any secret messages and accordingly discriminate.

### INRIA Dataset

Many datasets and protocols of various sizes become standard for comparison of the image, video, text and, etc. for evaluation. Popular data-sets are common in various areas like Image Processing, AI, Cyber Security, and so on. Those data-sets which are publicly available give the ability to researchers for validating, benchmarking, and reproducing approaches which have been previously reported or presented, thereby emphasizing transparency and integrity of academic researches. Image datasets are a highly important part of image classification researches. They are needed for the learning of visual image models and assessing the efficiency of classification and detection [8]

In 2008 [10], INRIA data-sets were produced in the context of the ANR RAFFUT project. Grouping two companies INRIA (for contacting: “Herve Jegou”) and the Advestigo lead to support The “Agence Nationale de la Recherche”. In their work these enhances new approaches for large scale image search. Standard approaches are built based on the bag-of-properties image representation. Initially, they analyze bag-of-Properties in the structure of approximate nearest neighbor searching, which proves the sub-optimality of this type of representation to match descriptors and lead to deriving a more accurate representing according to [9]:

- 1) **Hamming Embedding (HE):** offers binary signatures refining the matching according visual words.
- 2) **Weak Geometric Consistency constraints (WGC):** filters matching descriptors which are inconsistent according to angle and scale.

Combining of both two above in inverted file and are sufficiently utilized for every image, even with huge data-sets. Experiments that have been done on a data-set of million images showed great deal of enhancement because of the binary signature and the weak geometrical consistency constraints in addition to their sufficiency. Estimating the entire geometrical transforming, in other words, a reranking stage on a small set of images is complementary to the weak geometrical consistency constraints and offers additionally improving the precision [9].

They propose in the following a binary signature generation procedure that distinguish between Off-line learning procedure, which is performed on a learning dataset and generates a set of fixed values, and Binary signature computation itself The offline procedure is performed as follows: [9]

#### 1. Random matrix generation:

A  $d_b \times d$  orthogonal projection matrix P is generated. They randomly draw a matrix of Gaussian values and apply a QR factorization to it. The first  $d_b$  rows of the orthogonal matrix obtained by this decomposition form the matrix P.

#### 2. Descriptor projection and assignment:

A large set of descriptors  $x_i$  from an independent dataset is projected using P. These descriptors ( $z_{i1}, \dots, z_{id_b}$ ) are assigned to their closest centroid  $q(x_i)$ .

#### 3. Median values of projected descriptors:

For each centroid l and each projected component

$h = 1, \dots, d_b$ , they compute the median value  $\tau_{l,h}$  of the set  $\{z_{ih}|q(x_i) = l\}$  that corresponds to the descriptors assigned to the cell l. The fixed projection matrix P and  $k \times d_b$  median values  $\tau_{h,l}$  are used to perform the HE of a given descriptor x by:







1. Assigning x to its closest centroid, resulting in  $q(x)$ .
2. Projecting x using P, which produces a vector  $z = P x = (z_1, \dots, z_{d_b})$ .
3. Computing the signature  $b(x) = (b_1(x), \dots, b_{d_b}(x))$  as

$$b_i(x) = \begin{cases} 1 & \text{if } z_i > \tau_{q(x),i} \\ 0 & \text{otherwise.} \end{cases}$$

Table (1) shows Results for various queries.

Table (1)

Queries from the Holidays data-set and some corresponding results for Holidays+1M distractors Flickr1M [8].

Query		Correct results and their ranks			
	baseline		6195		49563
	WGC		1949		7114
	HE		50		30657
	HE+WGC		41		22036
	Re-ranked		1		22036
	baseline		5890		43064
	WGC		173		1234
	HE		4		55
	HE+WGC		3		13
	Re-ranked		2		7

The re-ranking is based on the estimation of an affine transformation with their implementation of [11].

Three datasets are provided by INRIA: [10]

- The **INRIA Holidays** data-set for evaluating image search, See Fig. (1).
- The **INRIA Copy-days** data-set for evaluating copy detecting. See Table (2).

The **BIGANN evaluating** data-set for evaluating Approximate nearest neighbors search.



Fig.(1): Sample of images from the INRIA Holidays Dataset. [10].



**Table (2)**  
*Sample of images from INRIA Copydays Dataset [10].*



Table (3) explains INRIA dataset, Description and Statistical [10] and Table (4) shows result after submit a query image according to a specific method.

**Table (3)**  
*INRIA dataset, Description and Statistical [10].*

Name	Description	Statistical
<b>Holidays</b>	<ul style="list-style-type: none"> <li>• Remaining ones have been taken on deliberately for testing how robust they are to different types of transformations such as: rotations, view-point and variations in illumination, and blurring.</li> <li>• Personal holiday images of INRIA researchers of about 1491 images.</li> <li>• It includes 500 image sets (i.e. queries), every one of which denotes a separate scene (for the sake of incorporating diversity) and 991 corresponding relevant images.</li> <li>• It so many scene categories (man-made, water and fire effects, natural, and so on) and high resolution.</li> <li>• Number of generated descriptors: 4455091 SIFT descriptors of 128 dimensional.</li> <li>• First image of every one of the groups is the query image and the precise retrieving outputs are the rest of the images of the set.</li> </ul>	<ul style="list-style-type: none"> <li>• Descriptors with size equal to (526MB).</li> <li>• Images: jpg1.tar.gz and jpg2.tar.gz (1.1GB and 1.6 GB).</li> <li>• Visual dictionaries (100, 200, 500, 1K, 2K, 5K, 10K, 20K, 50K, 100K and 200K visual words, 144MB) learned on Flickr60K.</li> <li>• Descriptors of Flickr60K (independent dataset, 636MB).</li> <li>• Pre-computed features for one million images, stored in 1000 archives of 1000 feature files each (235GB in total).</li> </ul>
<b>Copydays</b>	<ul style="list-style-type: none"> <li>• Every one of the images has undergone 3 types of artificial attacks, which are: JPEG, cropping and "strong".</li> <li>• It's a group of images that is entirely made up of personal holidays' images.</li> <li>• It's assessed via merging the original images in a large data-base of images because of its small image.</li> <li>• It's utilized for the evaluation of the indexing algorithms behavior for most widely used image copies.</li> </ul>	<ul style="list-style-type: none"> <li>• Cropped images, from 10% to 80% of the image surface removed, 2.0GB).</li> <li>• Original images (208MB).</li> <li>• 229 Strongly attacked images (60MB), print and scan, blur, paint.</li> <li>• Scale+JPEG attacked images (77MB), scale: 1/16 (pixels), JPEG quality factors: 75, 50, 30, 20, 15, 10, 8, 5, 3.</li> </ul>
<b>BIGANN evaluation</b>	<ul style="list-style-type: none"> <li>• Specifically, there is a huge group of a billion vectors, which is the largest group available for evaluating Artificial Neural Network approaches.</li> <li>• It offers numerous evaluation groups for the evaluation of the quality of approximate nearest neighbors search approach on various data types and different sizes of data-bases.</li> <li>• Every one of which includes three sub-sets of vectors: <ul style="list-style-type: none"> <li>▪ Query vectors.</li> <li>▪ Base vectors: vectors where the search is done.</li> <li>▪ Learning vectors: detect parameters that are involved in a specific approach.</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• ANN_SIFT1M: sift.tar.gz, (161MB)</li> <li>• ANN_SIFT10K: siftsmall.tar.gz, (5.1MB).</li> <li>• ANN_GIST1M: gist.tar.gz, (2.6GB)</li> <li>• ANN_SIFT1B: Base set, (92 GB) <ul style="list-style-type: none"> <li>▪ Learning set, (9.1 GB)</li> <li>▪ Query set, (964 KB)</li> <li>▪ Groundtruth, (512 MB)</li> </ul> </li> </ul>



## Related Works

Digital multimedia application like data retrieving plays an important part in the area of image and video retrieving for retrieving data which is useful to a user query. In the past years, a wide range of researches have been designed for digital multi-media data retrieval and object Recognition which has captures a

great deal of interest in each of computer vision and robotics communities taking its motivation from applications in autonomous driving, augmented reality or geo-localizing archiving imagery [12]. Table (5) and Table (6) show previous works used INRIA dataset for information / image retrieval and image steganalysis and detection respectively.

**Table (5)**

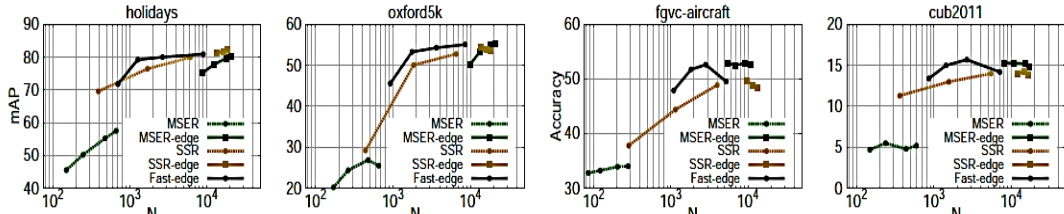
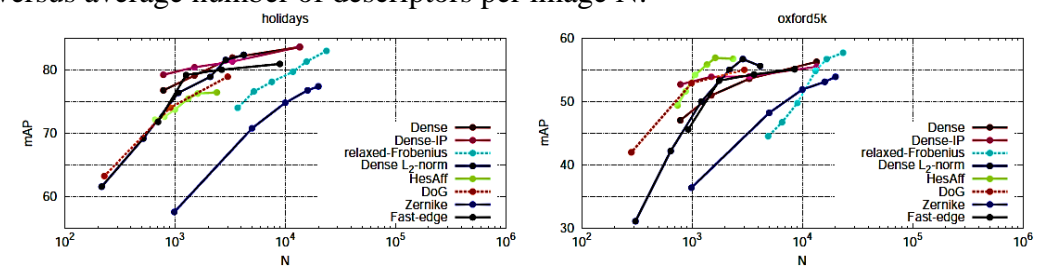
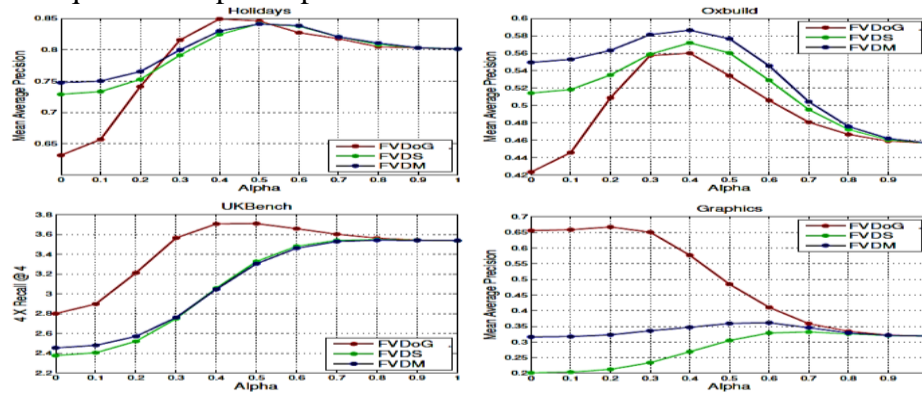
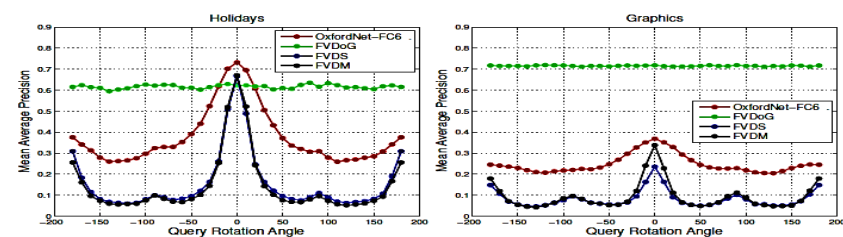
***Related works using Holiday dataset for Image / Information retrieval and Detection.***

Paper, Year	Description
[13, 2015]	<ul style="list-style-type: none"> <li>• Proposed methods on the base of super-pixels, edges, and a bank of Zernike filters utilized as detectors.</li> <li>• Results indicated that the regular dense detector is inferior to other approaches in the majority of cases, resulting in improving the standard in comparable setups on standard retrieving and fined-grained benchmarks.</li> <li>• Recent image retrieval and fine-grained classification indicators were evaluated for dissimilar approaches.</li> <li>• Existing blob and super pixel extracting approaches performed with high degree of precision in the case where the patches are obtained along the edges and not surrounding the found areas.</li> </ul>
[14, 2016]	<ul style="list-style-type: none"> <li>• Proposed an extensive research systematically evaluating the Fisher Vectors (FVs) and Convolutional NNs (CNNs) for instant image retrieval.</li> <li>• First part performs a comparison the performances of Fisher Vectors and Convolutional NNs on numerous publicly available datasets and for several criteria. (None of the descriptors are systematically more efficient than the other and that efficiency gains are typically be obtained with the use of a combination of both types.</li> <li>• Second part is focused on the effect of geometry transformations.</li> <li>• Specifically useful and implementable guide-lines to any person who is looking for standard global descriptors which are optimal for their exact image instance retrieving task.</li> <li>• The efficiency of Convolutional Neural Network could rapidly degrade with specific transformations and suggest some means of incorporating the necessary invariances in the pipeline of the Convolutional Neural Network.</li> </ul>
[15, 2016]	<ul style="list-style-type: none"> <li>• First, the task is developing a CNN model which is trainable in an end-to-end way in a direct manner for the place identification process. <ul style="list-style-type: none"> <li>▪ NetVLAD, which is a new generalized layer of VLAD, and takes its inspiration by the “Vector of Locally Aggregated Descriptors” image representation typically utilized in retrieving images.</li> <li>▪ Layer is readily pluggable into any Convolutional Neural Network model and amenable to training by back-propagation approach.</li> </ul> </li> <li>• Second, the process is developing a training process, according to a new weakly supervised ranking loss, for the sake of learning parameters of the model in end-to-end way from images that depict same places over time downloaded from “Google Street View Time Machine”.</li> <li>• This model is greatly superior to the non-learnt image representation and off-the-shelf Convolutional Neural Network descriptors on 2 challenging place identification indicator, and enhances over current standard compact image representations on standard image retrieving representations.</li> </ul>

Paper, Year	Description
[16, 2017]	<ul style="list-style-type: none"> <li>• Gradient Boost Ensemble Classification (GBEC) technique is proposed to solve problem of non-effect for achieving higher precision and recall for video retrieval.</li> <li>• It first takes video query as input.</li> <li>• It used Independent Component Analysis Model for extracting the visual properties like the shape, color, texture in videos for efficient classification.</li> <li>• After extracting visual features, GBEC approach applied Gradient Boost Ensemble Classifier in order for classifying the videos in a certain data-set as similar or dissimilar based on video query which resulted in enhanced classification precision.</li> <li>• Finally, the classified similar videos' are retrieved on the basis of the video query which in turn helps for increasing video retrieval accuracy and recall with the minimal amount of time.</li> <li>• The performance of GBEC approach is measured in according to metrics such as classification precision, time complexity, Precision and recall with aid of three datasets in comparison with the state-of-the-art works.</li> </ul>
[17, 2017]	<ul style="list-style-type: none"> <li>• It alleviates the restrictive compactness needs of hash codes via extending those codes into a 2-level hierarchic coding strategy.</li> <li>• It generates sufficient multi-level hashing codes for image retrieving, on the base of the deep Siamese CNNs (DSCNN).</li> <li>• At execution time, adopting an attention-based approach for selecting some of its most important bits special for every one of the query images for retrieval rather than the use of full hash codes of the 1<sup>st</sup> level.</li> </ul>
[18, 2017]	<ul style="list-style-type: none"> <li>• Proposed a "Bootstrap Aggregative Learning Classifier" (BALC) approach.</li> <li>• It is designed by enhancing the efficiency of digital multi-media data retrieving with the use of machine learning classifier method.</li> <li>• At first, it takes video query as input and applies "Kernel-principal component analysis" (Kernel-PCA) for the extraction of the visual properties like shape, colour, and texture in videos.</li> <li>• Then, BALC approach uses "Bootstrap Aggregation with Support Vector Machine" (BA-SVM) Classifier for the classification of videos in a certain data-set as relevant or irrelevant with the use of video query with enhanced classification precision.</li> <li>• Finally, the BALC approach performs retrieval of the classified relevant videos' according to the query.</li> <li>• This procedure helps BALC approach in improving video retrieval recall and accuracy in the minimal amount of time.</li> <li>• The BALC approach works on conducting experiments on parameters like the precision of the classification, time complexity, accuracy and recall with the use of 3 data-sets with higher classification precision and minimal amount of time complexity for multi-media data retrieving in comparison with the standard works.</li> </ul>
[19, 2018]	<ul style="list-style-type: none"> <li>• Proposed an innovative model for achieving competitive retrieval efficiency.</li> <li>• Present different masking approaches, such as SIFT-mask, SUMmask, and MAX-mask, for selecting a representative sub-set of local convolutional properties and eliminate many redundant properties from a feature map. They are effectively for addressing the burstiness problem and enhance the accuracy of retrieval.</li> <li>• Proposed employing recent embedding and aggregating approaches for further enhancement of property discriminability.</li> <li>• Include a hashing module to produce compact binary image representations which are effective for the retrieval.</li> </ul>

Table (6)

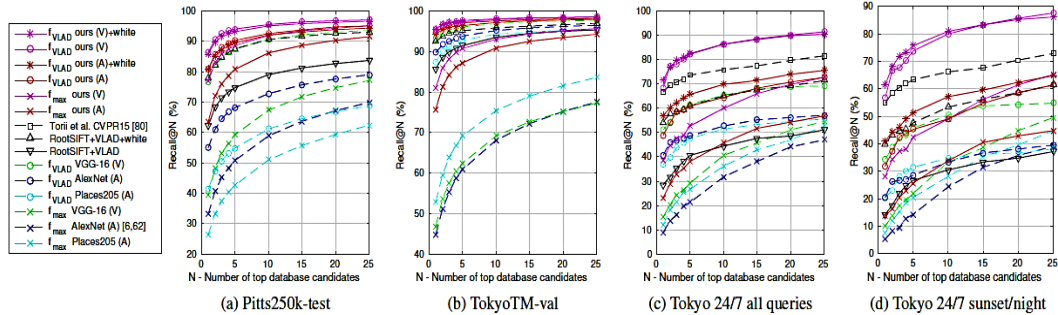
Results of Related works using Holiday dataset for Image / Information retrieval and Detection.

Paper, Year	Results
[13,2015]	<div style="display: flex; justify-content: space-around;">  </div> <ul style="list-style-type: none"> <li>Performance comparison between approaches that describe a detected region by fitting a single ellipse and approaches that sample patches along the region edges. Fast-edge is another detector that samples patches on edges. We show performance versus average number of descriptors per image N.</li> </ul> <div style="display: flex; justify-content: space-around;">  </div> <ul style="list-style-type: none"> <li>Performance comparison on image retrieval versus number of descriptors per image N.</li> </ul>
[14,2016]	<ul style="list-style-type: none"> <li>4 data sets: Holidays, UKBench, Oxford buildings and Stanford Mobile Visual Search to study how well CNN-based approaches generalize compared with FVs. Image is equal to 640 pixels prior to FV extraction.</li> </ul> <div style="display: flex; flex-wrap: wrap;">  </div> <ul style="list-style-type: none"> <li>Combining different FVs with OxfordNet fc6 with early fusion. FV and OxfordNet features are concatenated with weights <math>\alpha</math> and <math>1-\alpha</math> respectively. <math>\alpha = 0</math> refers to just using FVDoG, FVDS or FVDM above, while <math>\alpha = 1</math> refers to just the OxfordNet fc6 feature. We observe that retrieval performance improves on all data sets by combining FV and CNN.</li> </ul> <div style="display: flex; justify-content: space-around;">  </div> <ul style="list-style-type: none"> <li>Comparison of OxfordNet fc6 and FV for rotated queries on the Holidays and Graphics data sets. The FVDoG is robust to rotation, while OxfordNet, FVDS, FVDM suffer a sharp drop in performance.</li> </ul>



**Paper, Year** **Results**

- Use two base architectures which are extended with Max pooling (fmax) and our NetVLAD (fV LAD) layers: AlexNet and VGG-16; both are cropped at the last convolutional layer (conv5), before ReLU. For NetVLAD we use K = 64 resulting in 16k and 32k-D image representations for the two base architectures, respectively.



[15,2016]

Comparison of our methods versus off-the-shelf networks and state-of-the-art. The base CNN architecture is denoted in brackets: (A)lexNet and (V)GG-16. Trained representations (red and magenta for AlexNet and VGG-16) outperform by a large margin off-the-shelf ones (blue, cyan, green for AlexNet, Places205, VGG-16), fV LAD (-o-) works better than fmax (-x-), and our fV LAD+whitening (-\*-) representation based on VGG-16 sets the state-of-the-art on all datasets. [80] only evaluated on Tokyo 24/7 as the method relies on depth data not available in other datasets.

**Comparison with state-of-the-art.** Figure above also shows that our trained fV LAD representation with whitening based on VGG-16 (magenta -o-) convincingly outperforms Root-SIFT+VLAD+whitening, as well as the method of Torii et al. and therefore sets the state-of-the-art for compact descriptors on all benchmarks. Note that these are strong baselines that outperform most off-the-shelf CNN descriptors on the place recognition task.

**VLAD versus Max.** By comparing fV LAD (-o-) methods with their corresponding fmax (-x-) counterparts it is clear that VLAD pooling is much better than Max pooling for both off-the-shelf and trained representations.

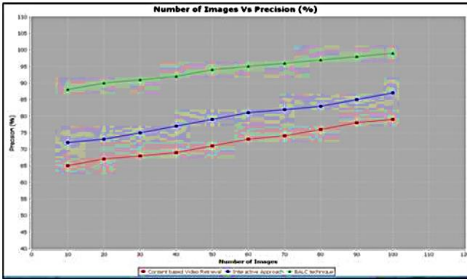
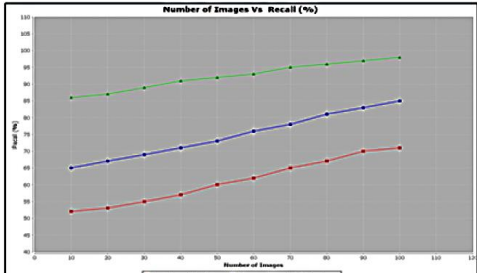
NetVLAD performance decreases gracefully with dimensionality:

128-D NetVLAD performs similarly to 512-D . Max (42.9% vs 38.4% recall@1 on Tokyo 24/7), resulting in four times more compact representation for the same performance. Furthermore, NetVLAD+whitening outperforms Max pooling convincingly when reduced to the same dimensionality (60%).

[16,2017]

- Gradient Boost Ensemble Classifier helps to enhance the accuracy of video classification with minimum time.

Number of Images	Classification Accuracy (%)		
	Content Based Video Shot Classification	Large-Scale Video Retrieval	GBEC technique
10	63.54	72.35	87.32
20	64.58	73.58	88.87
30	65.98	74.69	89.54
40	66.47	76.12	90.12
50	67.69	77.65	91.47
60	68.87	78.96	92.56
70	69.84	79.51	93.65
80	72.54	81.25	94.75
90	74.56	82.54	95.87
100	75.98	83.69	97.86

Paper, Year	Results																																																																																																																																																																																																						
[17,2017]	<table border="1"> <thead> <tr> <th>Method</th> <th>12bits</th> <th>24bits</th> <th>32bits</th> <th>48bits</th> </tr> </thead> <tbody> <tr> <td>CNNH [19]</td> <td>43.9</td> <td>51.1</td> <td>50.9</td> <td>52.2</td> </tr> <tr> <td>CNNH+ [19]</td> <td>46.5</td> <td>52.1</td> <td>52.1</td> <td>53.2</td> </tr> <tr> <td>SDH [15]</td> <td>-</td> <td>-</td> <td>20.83</td> <td>-</td> </tr> <tr> <td>NINH [26]</td> <td>55.2</td> <td>56.6</td> <td>55.8</td> <td>58.1</td> </tr> <tr> <td>DSCH [43]</td> <td>-</td> <td>61.3</td> <td>61.7</td> <td>61.9</td> </tr> <tr> <td>DRSCH [43]</td> <td>-</td> <td>62.1</td> <td>62.8</td> <td>63.0</td> </tr> <tr> <td>DSH [21]</td> <td>61.5</td> <td>65.1</td> <td>66.7</td> <td>67.5</td> </tr> <tr> <td>DPSH [20]</td> <td>68.2</td> <td>68.6</td> <td>72.5</td> <td>73.3</td> </tr> <tr> <td><math>H_g</math> code</td> <td>74.3</td> <td>79.9</td> <td>80.8</td> <td>81.2</td> </tr> <tr> <td><math>HDH\_shallow_{global}</math></td> <td>-</td> <td>-</td> <td>-</td> <td>37.2</td> </tr> <tr> <td><math>HDH\_shallow_{rerank}</math></td> <td>-</td> <td>-</td> <td>-</td> <td>40.1</td> </tr> <tr> <td>HDH (baseline)</td> <td>77.2</td> <td>83.2</td> <td>84.6</td> <td>84.9</td> </tr> <tr> <td>HDH (wLinCb-HD)</td> <td>75.2</td> <td>82.3</td> <td>84.0</td> <td><b>86.3</b></td> </tr> <tr> <td>HDH (wAtt-HD)</td> <td><b>77.8</b></td> <td><b>83.9</b></td> <td><b>84.9</b></td> <td>85.9</td> </tr> </tbody> </table> <p>Comparison with state-of-the-art deep hashing methods</p>	Method	12bits	24bits	32bits	48bits	CNNH [19]	43.9	51.1	50.9	52.2	CNNH+ [19]	46.5	52.1	52.1	53.2	SDH [15]	-	-	20.83	-	NINH [26]	55.2	56.6	55.8	58.1	DSCH [43]	-	61.3	61.7	61.9	DRSCH [43]	-	62.1	62.8	63.0	DSH [21]	61.5	65.1	66.7	67.5	DPSH [20]	68.2	68.6	72.5	73.3	$H_g$ code	74.3	79.9	80.8	81.2	$HDH\_shallow_{global}$	-	-	-	37.2	$HDH\_shallow_{rerank}$	-	-	-	40.1	HDH (baseline)	77.2	83.2	84.6	84.9	HDH (wLinCb-HD)	75.2	82.3	84.0	<b>86.3</b>	HDH (wAtt-HD)	<b>77.8</b>	<b>83.9</b>	<b>84.9</b>	85.9																																																																																																																											
Method	12bits	24bits	32bits	48bits																																																																																																																																																																																																			
CNNH [19]	43.9	51.1	50.9	52.2																																																																																																																																																																																																			
CNNH+ [19]	46.5	52.1	52.1	53.2																																																																																																																																																																																																			
SDH [15]	-	-	20.83	-																																																																																																																																																																																																			
NINH [26]	55.2	56.6	55.8	58.1																																																																																																																																																																																																			
DSCH [43]	-	61.3	61.7	61.9																																																																																																																																																																																																			
DRSCH [43]	-	62.1	62.8	63.0																																																																																																																																																																																																			
DSH [21]	61.5	65.1	66.7	67.5																																																																																																																																																																																																			
DPSH [20]	68.2	68.6	72.5	73.3																																																																																																																																																																																																			
$H_g$ code	74.3	79.9	80.8	81.2																																																																																																																																																																																																			
$HDH\_shallow_{global}$	-	-	-	37.2																																																																																																																																																																																																			
$HDH\_shallow_{rerank}$	-	-	-	40.1																																																																																																																																																																																																			
HDH (baseline)	77.2	83.2	84.6	84.9																																																																																																																																																																																																			
HDH (wLinCb-HD)	75.2	82.3	84.0	<b>86.3</b>																																																																																																																																																																																																			
HDH (wAtt-HD)	<b>77.8</b>	<b>83.9</b>	<b>84.9</b>	85.9																																																																																																																																																																																																			
[18,2017]	<ul style="list-style-type: none"> <li>Impact of Classification Accuracy</li> </ul> <p>Tabulation of Classification Accuracy for Image Retrieval</p> <table border="1"> <thead> <tr> <th rowspan="2">Number of Images (N)</th> <th colspan="3">Classification Accuracy (%)</th> </tr> <tr> <th>Content based Video Retrieval</th> <th>Interactive Approach</th> <th>BALC technique</th> </tr> </thead> <tbody> <tr><td>10</td><td>66</td><td>72</td><td>88</td></tr> <tr><td>20</td><td>68</td><td>74</td><td>89</td></tr> <tr><td>30</td><td>71</td><td>75</td><td>90</td></tr> <tr><td>40</td><td>73</td><td>76</td><td>91</td></tr> <tr><td>50</td><td>75</td><td>78</td><td>92</td></tr> <tr><td>60</td><td>76</td><td>80</td><td>94</td></tr> <tr><td>70</td><td>77</td><td>81</td><td>95</td></tr> <tr><td>80</td><td>79</td><td>82</td><td>96</td></tr> <tr><td>90</td><td>80</td><td>83</td><td>97</td></tr> <tr><td>100</td><td>81</td><td>87</td><td>98</td></tr> </tbody> </table> <ul style="list-style-type: none"> <li>Impact of Precision and Impact of Recall</li> </ul> <div style="display: flex; justify-content: space-around;">   </div> <p>Measurement of Precision for Image Retrieval Measurement of Recall for Image Retrieval</p>	Number of Images (N)	Classification Accuracy (%)			Content based Video Retrieval	Interactive Approach	BALC technique	10	66	72	88	20	68	74	89	30	71	75	90	40	73	76	91	50	75	78	92	60	76	80	94	70	77	81	95	80	79	82	96	90	80	83	97	100	81	87	98																																																																																																																																																							
Number of Images (N)	Classification Accuracy (%)																																																																																																																																																																																																						
	Content based Video Retrieval	Interactive Approach	BALC technique																																																																																																																																																																																																				
10	66	72	88																																																																																																																																																																																																				
20	68	74	89																																																																																																																																																																																																				
30	71	75	90																																																																																																																																																																																																				
40	73	76	91																																																																																																																																																																																																				
50	75	78	92																																																																																																																																																																																																				
60	76	80	94																																																																																																																																																																																																				
70	77	81	95																																																																																																																																																																																																				
80	79	82	96																																																																																																																																																																																																				
90	80	83	97																																																																																																																																																																																																				
100	81	87	98																																																																																																																																																																																																				
[19,2018]	<table border="1"> <thead> <tr> <th rowspan="2">Hashing</th> <th rowspan="2">Embedding</th> <th rowspan="2">Dim.</th> <th colspan="4">Oxford5k</th> <th colspan="4">Paris6k</th> <th colspan="4">Holidays</th> </tr> <tr> <th>64</th> <th>128</th> <th>256</th> <th>512</th> <th>64</th> <th>128</th> <th>256</th> <th>512</th> <th>64</th> <th>128</th> <th>256</th> <th>512</th> </tr> </thead> <tbody> <tr> <td rowspan="4">ITQ [35]</td> <td><math>\phi_{\Delta}(32,20)</math></td> <td>512</td> <td>18.3</td> <td><b>31.6</b></td> <td>45.0</td> <td>57.3</td> <td>32.9</td> <td>49.7</td> <td>63.0</td> <td>74.7</td> <td>57.5</td> <td>70.7</td> <td>79.5</td> <td>83.5</td> </tr> <tr> <td><math>\phi_{\Delta}(64,18)</math></td> <td>1024</td> <td>18.7</td> <td>29.5</td> <td>42.9</td> <td>55.8</td> <td>33.5</td> <td>49.0</td> <td>61.0</td> <td>72.4</td> <td>58.7</td> <td>71.5</td> <td>79.4</td> <td>82.9</td> </tr> <tr> <td><math>\phi_{\Delta}(64,34)</math></td> <td>2048</td> <td>18.3</td> <td>27.7</td> <td>38.4</td> <td>50.3</td> <td>28.4</td> <td>44.3</td> <td>57.9</td> <td>68.6</td> <td>57.9</td> <td>71.1</td> <td>79.1</td> <td>82.3</td> </tr> <tr> <td><math>\phi_{\Delta}(64,66)</math></td> <td>4096</td> <td>16.0</td> <td>23.0</td> <td>33.6</td> <td>45.8</td> <td>26.1</td> <td>40.1</td> <td>52.9</td> <td>65.4</td> <td>56.1</td> <td>70.2</td> <td>78.8</td> <td>80.5</td> </tr> <tr> <td rowspan="4">RBA [36]</td> <td><math>\phi_{\Delta}(32,20)</math></td> <td>512</td> <td>17.8</td> <td>31.3</td> <td><b>45.3</b></td> <td>57.1</td> <td>31.5</td> <td><b>50.8</b></td> <td><b>63.7</b></td> <td>74.9</td> <td>56.7</td> <td>71.6</td> <td>78.5</td> <td>83.2</td> </tr> <tr> <td><math>\phi_{\Delta}(64,18)</math></td> <td>1024</td> <td>18.4</td> <td>30.1</td> <td>42.7</td> <td>55.7</td> <td>32.8</td> <td>49.4</td> <td>61.3</td> <td>72.8</td> <td>57.6</td> <td>71.1</td> <td>79.3</td> <td>82.7</td> </tr> <tr> <td><math>\phi_{\Delta}(64,34)</math></td> <td>2048</td> <td>17.3</td> <td>30.9</td> <td>39.1</td> <td>53.5</td> <td>29.0</td> <td>45.0</td> <td>59.1</td> <td>68.6</td> <td>57.1</td> <td>70.8</td> <td>78.8</td> <td>81.8</td> </tr> <tr> <td><math>\phi_{\Delta}(64,66)</math></td> <td>4096</td> <td>15.7</td> <td>25.1</td> <td>39.0</td> <td>48.3</td> <td>25.8</td> <td>39.3</td> <td>55.6</td> <td>64.6</td> <td>55.6</td> <td>67.5</td> <td>77.5</td> <td>80.1</td> </tr> <tr> <td rowspan="4">KMH [70]</td> <td><math>\phi_{\Delta}(32,20)</math></td> <td>512</td> <td>18.5</td> <td>26.5</td> <td>39.1</td> <td>54.4</td> <td>32.0</td> <td>45.7</td> <td>61.6</td> <td>75.3</td> <td>53.4</td> <td>65.0</td> <td>75.0</td> <td>80.8</td> </tr> <tr> <td><math>\phi_{\Delta}(64,18)</math></td> <td>1024</td> <td>18.1</td> <td>28.1</td> <td>41.4</td> <td>53.5</td> <td>30.0</td> <td>48.3</td> <td>61.4</td> <td>73.7</td> <td>54.6</td> <td>68.0</td> <td>78.0</td> <td>82.4</td> </tr> <tr> <td><math>\phi_{\Delta}(64,34)</math></td> <td>2048</td> <td>15.7</td> <td>26.7</td> <td>38.5</td> <td>51.4</td> <td>26.0</td> <td>40.8</td> <td>56.7</td> <td>69.2</td> <td>51.2</td> <td>68.4</td> <td>76.4</td> <td>81.2</td> </tr> <tr> <td><math>\phi_{\Delta}(64,66)</math></td> <td>4096</td> <td>13.4</td> <td>20.7</td> <td>31.2</td> <td>47.0</td> <td>21.1</td> <td>33.3</td> <td>49.8</td> <td>63.7</td> <td>50.0</td> <td>63.2</td> <td>72.8</td> <td>80.1</td> </tr> </tbody> </table> <ul style="list-style-type: none"> <li>Comparison when the final representations are binary values. Values in brackets in Embedding column indicate dimension of local conv. features after PCA and the codebook size, respectively. Dim. column indicates dimension of real-valued representations before subjecting into a hash function. Evaluate binary representations at code lengths 64; 128; 256; 512 with three hashing [ITQ, RBA, and KMH]. Results reported on Oxford5k, Paris6k and Holidays dataset.</li> </ul>	Hashing	Embedding	Dim.	Oxford5k				Paris6k				Holidays				64	128	256	512	64	128	256	512	64	128	256	512	ITQ [35]	$\phi_{\Delta}(32,20)$	512	18.3	<b>31.6</b>	45.0	57.3	32.9	49.7	63.0	74.7	57.5	70.7	79.5	83.5	$\phi_{\Delta}(64,18)$	1024	18.7	29.5	42.9	55.8	33.5	49.0	61.0	72.4	58.7	71.5	79.4	82.9	$\phi_{\Delta}(64,34)$	2048	18.3	27.7	38.4	50.3	28.4	44.3	57.9	68.6	57.9	71.1	79.1	82.3	$\phi_{\Delta}(64,66)$	4096	16.0	23.0	33.6	45.8	26.1	40.1	52.9	65.4	56.1	70.2	78.8	80.5	RBA [36]	$\phi_{\Delta}(32,20)$	512	17.8	31.3	<b>45.3</b>	57.1	31.5	<b>50.8</b>	<b>63.7</b>	74.9	56.7	71.6	78.5	83.2	$\phi_{\Delta}(64,18)$	1024	18.4	30.1	42.7	55.7	32.8	49.4	61.3	72.8	57.6	71.1	79.3	82.7	$\phi_{\Delta}(64,34)$	2048	17.3	30.9	39.1	53.5	29.0	45.0	59.1	68.6	57.1	70.8	78.8	81.8	$\phi_{\Delta}(64,66)$	4096	15.7	25.1	39.0	48.3	25.8	39.3	55.6	64.6	55.6	67.5	77.5	80.1	KMH [70]	$\phi_{\Delta}(32,20)$	512	18.5	26.5	39.1	54.4	32.0	45.7	61.6	75.3	53.4	65.0	75.0	80.8	$\phi_{\Delta}(64,18)$	1024	18.1	28.1	41.4	53.5	30.0	48.3	61.4	73.7	54.6	68.0	78.0	82.4	$\phi_{\Delta}(64,34)$	2048	15.7	26.7	38.5	51.4	26.0	40.8	56.7	69.2	51.2	68.4	76.4	81.2	$\phi_{\Delta}(64,66)$	4096	13.4	20.7	31.2	47.0	21.1	33.3	49.8	63.7	50.0	63.2	72.8	80.1
Hashing	Embedding				Dim.	Oxford5k				Paris6k				Holidays																																																																																																																																																																																									
		64	128	256		512	64	128	256	512	64	128	256	512																																																																																																																																																																																									
ITQ [35]	$\phi_{\Delta}(32,20)$	512	18.3	<b>31.6</b>	45.0	57.3	32.9	49.7	63.0	74.7	57.5	70.7	79.5	83.5																																																																																																																																																																																									
	$\phi_{\Delta}(64,18)$	1024	18.7	29.5	42.9	55.8	33.5	49.0	61.0	72.4	58.7	71.5	79.4	82.9																																																																																																																																																																																									
	$\phi_{\Delta}(64,34)$	2048	18.3	27.7	38.4	50.3	28.4	44.3	57.9	68.6	57.9	71.1	79.1	82.3																																																																																																																																																																																									
	$\phi_{\Delta}(64,66)$	4096	16.0	23.0	33.6	45.8	26.1	40.1	52.9	65.4	56.1	70.2	78.8	80.5																																																																																																																																																																																									
RBA [36]	$\phi_{\Delta}(32,20)$	512	17.8	31.3	<b>45.3</b>	57.1	31.5	<b>50.8</b>	<b>63.7</b>	74.9	56.7	71.6	78.5	83.2																																																																																																																																																																																									
	$\phi_{\Delta}(64,18)$	1024	18.4	30.1	42.7	55.7	32.8	49.4	61.3	72.8	57.6	71.1	79.3	82.7																																																																																																																																																																																									
	$\phi_{\Delta}(64,34)$	2048	17.3	30.9	39.1	53.5	29.0	45.0	59.1	68.6	57.1	70.8	78.8	81.8																																																																																																																																																																																									
	$\phi_{\Delta}(64,66)$	4096	15.7	25.1	39.0	48.3	25.8	39.3	55.6	64.6	55.6	67.5	77.5	80.1																																																																																																																																																																																									
KMH [70]	$\phi_{\Delta}(32,20)$	512	18.5	26.5	39.1	54.4	32.0	45.7	61.6	75.3	53.4	65.0	75.0	80.8																																																																																																																																																																																									
	$\phi_{\Delta}(64,18)$	1024	18.1	28.1	41.4	53.5	30.0	48.3	61.4	73.7	54.6	68.0	78.0	82.4																																																																																																																																																																																									
	$\phi_{\Delta}(64,34)$	2048	15.7	26.7	38.5	51.4	26.0	40.8	56.7	69.2	51.2	68.4	76.4	81.2																																																																																																																																																																																									
	$\phi_{\Delta}(64,66)$	4096	13.4	20.7	31.2	47.0	21.1	33.3	49.8	63.7	50.0	63.2	72.8	80.1																																																																																																																																																																																									

**Table (7)**  
**Related works using Holiday dataset for Image Steganalysis and Detection.**

Paper, Year	Description
[20,2010]	<ul style="list-style-type: none"> <li>• Designing multi-classifier that classifies the stego images based on steganographic algorithms of the images, furthermore it does distinguish cover from stego images.</li> <li>• This process of classification depends on the results of steganalysis for the decomposed blocks of image.</li> <li>• While the natural image most of the times be composed of heterogeneous regions, the image's decomposition may lead to smaller image blocks, each one of them is extra homogeneous.</li> <li>• Classifying the blocks of image to multiple classes then discover classifier for each one of the classes to decide if the block is from a stego or cover image with a particular steganographic algorithm.</li> <li>• As a result, it is possible to conduct the steganalysis process for the entire via fusing weighted steganalysis outcomes of all the blocks of image via a voting process.</li> <li>• Test results shall be presented to demonstrate the advantages of utilizing the suggested block-based image steganalysis for a multi-classifier.</li> </ul>
[21,2012]	<ul style="list-style-type: none"> <li>• It offers a recently developed Steganalysis of blind image based on nonlinear support vector machine and Contourlet transform.</li> <li>• Nonlinear support vector machine is utilized to distinguish the stego images and the cover images, and the properties of Contourlet transform are utilized for image's features extraction.</li> <li>• It offers better accuracy than plenty of current stegoanalysis approaches and it utilizes minimum number of characteristics in the transform domain.</li> <li>• The effectiveness of the suggested approach is explained via test results.</li> <li>• The performance is compared with Feature based steganalyzer (FBS), Contourlet based steganalyzer (WBS) and state of the art wavelet based steganalyzer (WBS).</li> <li>• The results display that the suggested approach is very effective in its accuracy of detection and computational cost.</li> </ul>
[22,2012]	<ul style="list-style-type: none"> <li>• For the purpose of making the performance of stegoanalysis better, a Block-based steganalysis of the image was proposed, that perform steganalysis on the smaller homogenous blocks of a specific test image</li> <li>• Nevertheless, it is noticed that when the size of feature is large then the computational complexity of the block-based image steganalysis is also high.</li> <li>• For the purpose of complexity reducing, a content-based feature selection approach is developed for binary classifier.</li> <li>• The major idea is to select significant characteristics based on types of block, that give and explanation of "content dependent" selection. (Make it possible to get finer performance by utilizing smaller number of characteristics also reducing the computational complexity).</li> <li>• Test results were conducted for the purpose of explaining the improvement in performance regarding content-dependent feature selection with high level of detection accuracy.</li> </ul>
[23,2013]	<ul style="list-style-type: none"> <li>• Decomposed block of image for Differentiate stego from cover-image.</li> <li>• Finding classifier regarding each class and classifying the image blocks to multi-classes is done after the image decomposition process to smaller blocks.</li> <li>• Thereafter, a process of integrating the results of image blocks through decision fusion will lead to the steganalysis of the entire image.</li> </ul>

Paper, Year	Description
	<ul style="list-style-type: none"> <li>• Conducting large-scale evaluation for the performance of block-based image steganalysis.</li> <li>• Trade-off between block number and block size through test image.</li> <li>• Suggest using overlap blocks to make the performance of steganalysis better.</li> <li>• Fisher linear discriminant classifier and the logistic classifier outperform linear Bayes classifier by a huge difference.</li> </ul>
[24,2014]	<ul style="list-style-type: none"> <li>• Discovering capability regarding steganography algorithm was utilized for sample stego-image via designing a multi classifier.</li> <li>• Kind of classification was used for the steganalysis of image's smaller blocks. Since the plain images mainly have heterogeneous regions. At first, main image was decomposed to smaller and similar blocks.</li> <li>• After that, those similar blocks were placed in same class.</li> <li>• Thus, a number of various classes were obtained, and for each class of them, suitable classifier was recognized.</li> <li>• This method led to making decisions for the purpose of recognizing and identifying the status of image blocks which were either stego or cover and recognizing steganography algorithm utilized in stego-images.</li> </ul>
[24,2016]	<ul style="list-style-type: none"> <li>• It offers an innovative host-based approach for discriminating and identifying Stegobot profiles.</li> <li>• Identify Stegobot network traffic that is inherently distinct from the legitimate multimedia social network traffic.</li> <li>• Performance distinct social networks dataset with distinct evaluation metrics.</li> <li>• Exploring botnet hidden communication structure is done with the help of proposed multiple aspects of multimedia characteristics.</li> <li>• Profiles of stegobot simulate authentic users and jeopardize others in the social network.</li> <li>• Process of detecting stegobot communication and bot profiles is very difficult when utilizing only one view features thus here a work of multi feature method is utilized.</li> <li>• Furthermore, this study tries to offer help to forensic analysts and experts in network security field to grasp the idea of Stegobot communication and the significant profiles in malicious networks.</li> </ul>

**Table (8)**  
**Results of Related works using Holiday dataset for Image Steganalysis and Detection.**

Paper, Year	Description
[20,2010]	<ul style="list-style-type: none"> <li>• Designing multi-classifier that classifies the stego images based on steganographic algorithms of the images, furthermore it does distinguish cover from stego images.</li> <li>• This process of classification depends on the results of steganalysis for the decomposed blocks of image.</li> <li>• While the natural image most of the times be composed of heterogeneous regions, the image's decomposition may lead to smaller image blocks, each one of them is extra homogeneous.</li> <li>• Classifying the blocks of image to multiple classes then discover classifier for each one of the classes to decide if the block is from a stego or cover image with a particular steganographic algorithm.</li> <li>• As a result, it is possible to conduct the steganalysis process for the entire via fusing weighted steganalysis outcomes of all the blocks of image via a voting process.</li> <li>• Test results shall be presented to demonstrate the advantages of utilizing the suggested block-based image steganalysis for a multi-classifier.</li> </ul>
[21,2012]	<ul style="list-style-type: none"> <li>• It offers a recently developed Steganalysis of blind image based on nonlinear support vector machine and Contourlet transform.</li> <li>• Nonlinear support vector machine is utilized to distinguish the stego images and the cover images, and the properties of Contourlet transform are utilized for image's features extraction.</li> <li>• It offers better accuracy than plenty of current stegoanalysis approaches and it utilizes minimum number of characteristics in the transform domain.</li> <li>• The effectiveness of the suggested approach is explained via test results.</li> <li>• The performance is compared with Feature based steganalyzer (FBS), Contourlet based steganalyzer (WBS) and state of the art wavelet based steganalyzer (WBS).</li> <li>• The results display that the suggested approach is very effective in its accuracy of detection and computational cost.</li> </ul>
[22,2012]	<ul style="list-style-type: none"> <li>• For the purpose of making the performance of stegoanalysis better, a Block-based steganalysis of the image was proposed, that perform steganalysis on the smaller homogenous blocks of a specific test image</li> <li>• Nevertheless, it is noticed that when the size of feature is large then the computational complexity of the block-based image steganalysis is also high.</li> <li>• For the purpose of complexity reducing, a content-based feature selection approach is developed for binary classifier.</li> <li>• The major idea is to select significant characteristics based on types of block, that give and explanation of "content dependent" selection. (Make it possible to get finer performance by utilizing smaller number of characteristics also reducing the computational complexity).</li> <li>• Test results were conducted for the purpose of explaining the improvement in performance regarding content-dependent feature selection with high level of detection accuracy</li> </ul>
[23,2013]	<ul style="list-style-type: none"> <li>• Decomposed block of image for Differentiate stego from cover-image.</li> <li>• Finding classifier regarding each class and classifying the image blocks to multi-classes is done after the image decomposition process to smaller blocks.</li> <li>• Thereafter, a process of integrating the results of image blocks through decision fusion will lead to the steganalysis of the entire image.</li> </ul>



Paper, Year	Description
	<ul style="list-style-type: none"> <li>• Conducting large-scale evaluation for the performance of block-based image steganalysis.</li> <li>• Trade-off between block number and block size through test image.</li> <li>• Suggest using overlap blocks to make the performance of steganalysis better.</li> <li>• Fisher linear discriminant classifier and the logistic classifier outperform linear Bayes classifier by a huge difference.</li> </ul>
[24,2014]	<ul style="list-style-type: none"> <li>• Discovering capability regarding steganography algorithm was utilized for sample stego-image via designing a multi classifier.</li> <li>• Kind of classification was used for the steganalysis of image's smaller blocks. Since the plain images mainly have heterogeneous regions. At first, main image was decomposed to smaller and similar blocks.</li> <li>• After that, those similar blocks were placed in same class.</li> <li>• Thus, a number of various classes were obtained, and for each class of them, suitable classifier was recognized.</li> <li>• This method led to making decisions for the purpose of recognizing and identifying the status of image blocks which were either stego or cover and recognizing steganography algorithm utilized in stego-images.</li> </ul>
[25,2016]	<ul style="list-style-type: none"> <li>• It offers an innovative host-based approach for discriminating and identifying Stegobot profiles.</li> <li>• Identify Stegobot network traffic that is inherently distinct from the legitimate multimedia social network traffic.</li> <li>• Performance distinct social networks dataset with distinct evaluation metrics.</li> <li>• Exploring botnet hidden communication structure is done with the help of proposed multiple aspects of multimedia characteristics.</li> <li>• Profiles of stegobot simulate authentic users and jeopardize others in the social network.</li> <li>• Process of detecting stegobot communication and bot profiles is very difficult when utilizing only one view features thus here a work of multi feature method is utilized.</li> <li>• Furthermore, this study tries to offer help to forensic analysts and experts in network security field to grasp the idea of Stegobot communication and the significant profiles in malicious networks.</li> </ul>

### Conclusion

Different techniques have being appeared in the field of information hiding for protect data especially image while transfer through communication.

This study outlines the problem of detecting attacks from embedding information in image. INRIA image dataset consists of scene categories (man-made, water and fire effects, natural, and so on) with high resolution which can be used for image retrieval extract descriptors and search neighbor. It reviews numbers of works for blind image steganalysis which used INRIA dataset.

### References

- [1] Johnson N. F. and Jajodia S., "Exploring Steganography: Seeing the Unseen", *IEEE Computer Society*, 31(2), 26–34, 1990.
- [2] Sumathi C.P., Santanam T. and Umamaheswari,G. "A Study of Various Steganographic Techniques Used for Information Hiding", *International Journal of Computer Science & Engineering Survey (IJCSES)*, 4(6), December 2013.
- [3] Rita R., *Feature Selection for Blind Image Steganalysis*, Phd thesis, Department of Computer Science and Engineering, School of Engineering and Technology, The Northcap University, 2017.

- [4] Fridrich J., M. Goljan, R. Du, "Detecting LSB steganography in color and gray-scale images", *IEEE Multimedia Magaz.*, Special Issue on Security, 22–28, 2001.
- [5] Huaiqing W., Shuozhong W., Cyber warfare: "Steganography vs. steganalysis", *Commun. ACM* 47, 76–82, October 2004.
- [6] Natarajan M. and Lopamudra N., "Steganalysis Algorithms for Detecting the Hidden Information In Image, Audio And Video Cover Media", *International Journal of Network Security & Its Application (IJNSA)*, 2(1), January 2010.
- [7] Sujatha P., "Image Steganalysis using Artificial Neural Networks", Phd thesis, VELS institute of Science Technology and Advanced Studies, Department Of Computer Science, School Of Computing Sciences, Pallavaram, Chennai. 2011
- [8] Kanal E., "Machine Learning in Cyber security", Carnegie Mellon University, Software Engineering Institute, 2017.
- [9] Herve J., Matthijs D. and Cordelia S., "Hamming Embedding and Weak geometry consistency for large scale image search", *Proceedings of the 10th European conference on Computer vision*, October, Springer, 2008, 304–317.
- [10] INRIA Holidays dataset: <http://lear.inrialpes.fr/~jegou/data.php>
- [11] Lowe, D., "Distinctive image features from scale-invariant keypoints". *IJCV* 60, 91–110. 2004
- [12] Wai K. C., "Multimedia Systems: Content Based Indexing and Retrieval". *The Electrical Engineering Handbook*, Elsevier, 2005.
- [13] Ahmet I., Giorgos T., Philippe-Henri G. and Hervé J., "A comparison of dense region detectors for image search and fine-grained classification", *IEEE Transaction on Image Processing*, 24(8), 2015.
- [14] Vijay C., Jie L., Olivier M., Hanlin G., Antoine V., "A practical guide to CNNs and Fisher Vectors for image instance retrieval", *Signal Processing*, Elsevier, 128, 426–439, 2016.
- [15] Relja A., Petr G., Akihiko T., "NetVLAD: CNN architecture for weakly supervised place recognition" Tokyo Tech Tomas Pajdla CTU in Prague Josef Sivic INRIA.
- [16] Vijayan K. and Chandra S.C, 2017, "Gradient boost ensemble classifier for unstructured Multimedia information retrieval", Creative Commons Attribution License, 2017.
- [17] Ge S., Xiaoyang T., "Hierarchical deep hashing for image retrieval", *Higher Education Press and Springer-Verlag Berlin Heidelberg* 11(2): 253–265, 2017.
- [18] Vijayan K., Chandrasekar C., "Digital Multimedia Information Retrieval Using Bootstrap Aggregative Learning Classifier", *International Journal of Scientific Engineering and Science* 1(11), 82-89, 2017.
- [19] Thanh-Toan D., Tuan H., Dang-Khoa L. T., Ngai-Man C., Thanh T., "From Selective Deep Convolutional Features to Compact Binary Representations for Image Retrieval", *Cornell university library*, 2018.
- [20] Seongho C., Jingwei W., and C.-C. J. K., Byung-Ho C., "Block-based image steganalysis for a multi-classifier", 2010.
- [21] Natarajan V. and Anitha R., "Blind Image Steganalysis Based on Contourlet Transform", *International Journal on Cryptography and Information Security (IJCIS)*, 2(3), September 2012.
- [22] Seongho C., Martin G., and C.-C. J. K. "Content-Dependent Feature Selection for Block-Based Image Steganalysis", *IEEE International Symposium on Circuits and Systems*, 2012.
- [23] Seongho C., Byung-Ho C., Martin G., C.-C. J. K, Vis J., *Commun. Image R.*, "Block-based image steganalysis:" Algorithm and performance evaluation, Elsevier, 24, 846–856, 2013.
- [24] Omrani S.L. Bayat P., "A new approach for block based steganalysis using a Multi-classifier", *International Journal on "Technical and Physical Problems of Engineering" (IJTPE)* Published by International Organization of IOTPE, Issue 20, 6 (3), 66-71, 2014
- [25] Natarajan V., Anitha R., "A multi-feature approach to detect Stegobot.", a covert multimedia social network botnet, Springer Science+Business Media New York, 2016.