International Journal of Information & Network Security (IJINS)
Vol.2, No.2, April 2013, pp. 138~153
ISSN: 2089-3299

Moments and Similarity Measure Feature Based Image Steganalysis Technique (MSM)

Souvik Bhattacharyya*, Gautam Sanyal**
* Departement of Computer Science and Engineering, University Institute of Technology, The Burdwan University
** Departement of Computer Science and Engineering, National Institute of Technology, Durgapur

Article Info

Article history:

Received Oct 10th, 2012
Revised Nov 20th, 2012
Accepted Dec 10th, 2012

Keyword:

Statistical Moments
Invariant Moments
SVM Classifier
KNN Classifier
MSM (Moments and Similarity Measure)

ABSTRACT

Steganalysis is art and science of detecting messages hidden using steganography. In this article a novel universal image steganalysis technique is proposed in which various moments like invariant, statistical moments and Zernike and various other similarity measure parameters like MSE, PSNR, RMSE, SSIM, Shannon's ENTROPY, DC Coefficient, DCT distance, NAE and Average Difference i.e. total 24 features are selected as the features. The proposed steganalysis methods has been designed based on three classifiers i) Back Propagation Neural Network and ii) SVM classifier and iii) K-nearest neighbor classifier. The proposed universal steganalyzer has been tested against few of the various well-known steganographic techniques which operate in both the spatial and transform domains. The experiments are performed using a large data set of JPEG and BMP images obtained from publicly available websites. The image data set is categorized with respect to different features of the image to determine their potential impact on steganalysis performance. Experimental results demonstrate the effectiveness and accuracy of the proposed technique compared to other existing techniques.

Corresponding Author:

Souvik Bhattacharyya,
Departement of Computer Science and Engineering,
University Institute of Technology,
The University of Burdwan, Golapbag (North), Burdwan-713104, West Bengal, India.
Email: souvik.bha@gmail.com

1. INTRODUCTION

Steganalysis can be considered as a two-class pattern classification problem which aims to determine whether a carrier medium is a cover or a stego one. A targeted steganalysis technique works on some specific type of stego-system and sometimes limited only on the image based analysis. Although the results of most of the targeted steganalysis techniques are very accurate but they are inflexible since they perform only on specific embedding algorithm and not the universal one. On the other hand blind steganalysis technique works on all types of embedding techniques and applicable to any type of carrier format. Mostly a blind steganalysis algorithm works based on learning the difference in the statistical properties of cover and stego carriers. Blind techniques are usually less accurate than the targeted ones, but a lot more flexible and expandable. Semi-blind steganalysis works on a specific range of stego-systems. Apart from distinguishing between cover and stego some steganalysis algorithms also tries to estimate the size of the embedded message and even can predict the content of the message.

Overview of Image Steganalysis Technique

The main objective of image steganalysis is to break steganography principle and detect of stego image. Almost all the image based steganalysis algorithms rely on the statistical differences between cover and stego image. Image steganalysis deals with three important categories: (a) Visual attacks: In these types

of attacks with an assistance of a computer or through inspection with a naked eye it reveal the presence of hidden information, which helps to separate the image into bit planes for further more analysis. (b) Statistical attacks: These types of attacks are more powerful and successful, because they reveal the smallest alterations in an images statistical behavior. Statistical attacks can be further divided into (i) Passive attack and (ii) Active attack. Passive attacks involves with identifying presence or absence of a covert message or embedding algorithm used etc. Mean while active attacks is used to investigate embedded message length or hidden message location or secret key used in embedding. (c) Structural attacks: The format of the data files changes as the data to be hidden is embedded; identifying this characteristic structure changes can help us to find the presence of image.

a) Visual attack

The idea of visual attack [1] is to remove any parts of the image that cover the message in order for the human eye to distinguish where there is any hidden message or still image content. An example for sequential embedding can be to extract the LSB plane of the image and check for any possible suspicious structure in the image. The LSB plane of a natural grey scale bmp image can be seen in Figure 1, where it is clear that there are not any suspicious structures, while viewing the LSB plane of the stego of the same cover image made with sequential embedding, it can seen some sort of structure on the left-most part which needs further investigation in the image.

b) Statistical attack: Chi-Square Analysis:

Andreas Pfitzmann and Andreas Wetfield [2] introduced a method based on statistical analysis of Pair of Values (PoVs) that are exchanged during sequential embedding. This attack works on the stego-system like EzStego [3] and JSteg[4]. Sequential embedding makes PoVs in the values embedded in. For example, embedding in the spatial domain makes PoVs (2i, 2i +1) such that $0 \leftrightarrow 1$, $2 \leftrightarrow 3$, $4 \leftrightarrow 5$, , $252 \leftrightarrow 253$, $254 \leftrightarrow 255$. This will affect the histogram Y_k of the images pixel value k, while the sum of $Y_{2i} + Y_{2i+1}$ will remain unchanged. Thus the expected distribution of the sum of the adjacent values will be as given in equation (1) and the value for the difference between distributions with v – 1 degrees of freedom will be like equation (2). From (1) and (2) the χ^2 statistic for the PoVs can be found out as given in (3).

$$E(Y_{2i}) = \frac{1}{2}(Y_{2i} + Y_{2i+1}) \quad (1)$$

$$\chi^2 = \sum_{i=1}^{v} \frac{(F - E(F))^2}{E(F)} \quad (2)$$

$$\chi^2_{PoV} = \sum_{i=1}^{127} \frac{((Y_{2i}) - (\frac{1}{2}(Y_{2i} + Y_{2i+1})))^2}{(Y_{2i} + Y_{2i+1})} \quad (3)$$

Statistical attack: RS Analysis:

Fridrich et al. [5] introduced a commanding but difficult steganalytic method that is able to accurately estimate the length of the embedded message on a digital image, for several LSB steganographic methods. The method is based on the fact that the content of each bit plane of an image is correlated with the remaining bit planes. In particular, for an 8-bit image, there is some degree of correlation between the LSB plane and the other remaining seven bit planes. When a message is inserted in the LSB plane, its content is considered to become randomized, and thus the correlation between the LSB planes with the remaining bit planes is reduced or lost. Let I be the image to be analyzed having width W and height H pixels. Each pixel has been denoted as P i.e. for a Gray Scale Image (8 bits per pixel image), value of P = 0, 1,. . . 255. Next step is to divide I into G disjoint groups of n adjacent pixels. For instance consider n = 4. Next define a discriminant function f which is responsible to give a real number $f(x_1,\ldots, x_n) \in R$ for each group of pixels $G = (x_1 \ldots . x_n)$. The objective is to capture the smoothness of G using f. Let the discrimination function f can be defined as in equation (4).

$$f(x_1,\ldots,x_n) = \sum_{i=1}^{n-1} |x_{i+1} - x_i| \quad (4)$$

Furthermore, let F1 be a flipping invertible function F1: $0 \leftrightarrow 1$, $2 \leftrightarrow 3$, . . . , $254 \leftrightarrow 255$, and F_{-1} be a shifting function denoted as F_{-1} : $-1 \leftrightarrow 0$, $1 \leftrightarrow 2$, . . . , $255 \leftrightarrow 256$ over P. For completeness, let F0 be the identity function such as $F_0(x) = x$ for all $x \in P$: Define a mask M that represents which function to apply to each element of a group G. The mask M is an n-tuple with values in -1, 0, 1. The value -1 stands for the

application of the function F_{-1}, 1 stands for the function F_1 and 0 stands for the identity function F_0. Similarly, define -M as M's compliment. Next step is to apply the discriminant function f with the functions F-1,0,1 defined through a mask M over all G groups to classify them into three categories Regular (R), Singular (S) and Unchanged (U) depending on how the flipping changes the value of the discrimination function.

$$\begin{aligned} &\bullet \text{ Regular groups: } G \epsilon R_M \Leftrightarrow f(F(G)) > f(G) \\ &\bullet \text{ Singular groups: } G \epsilon S_M \Leftrightarrow f(F(G)) < f(G) \\ &\bullet \text{ Unusable groups: } G \epsilon U_M \Leftrightarrow f(F(G)) = f(G) \end{aligned} \quad (5)$$

In similar manner R_{-M}, S_{-M} and U_{-M} can be defined for -M such that $(R_M + S_M)/2 \leq T$ and $(R_{-M} + S_{-M})/2 \leq T$, where T is the total number of G groups. The conclusion of RS Analysis method describes that, for typical images $R_M \approx R_{-M}$ and $S_M \approx S_{-M}$ and no change in R and S value for embedding messages of various sizes.

c) Structural attack

Structural attacks have been designed for taking the advantage of the high-level properties that are known to exist for a particular steganographic embedding algorithm. For example, both Hide & Seek and Stego Dos were forced to operate only on images of specific dimension [6, 7]. A steganalyst that happens to intercept images of those specific dimensions, may immediately flag them as suspicious which is required for further investigation. Structural attacks hardly ever scrutinize each image on its own merits. Instead, the images are scanned to see if they have generated some known side-effects for the different steganographic algorithms. Images that contain these properties are often subjected to further investigation.

In this paper , a novel feature based image steganalyzer has been proposed which takes several image quality measures namely invariant moments, statistical moments and Zernike moments as well as various other similarity measure parameters namely MSE, PSNR, RMSE, SSIM, Shannon's ENTROPY, DC Coefficient, DCT distance, NAE and Average Difference as features for the design of steganalyzer. Various image quality metric will act as a functional unit that converts its input signal into a measure that supposedly to be sensitive to the presence of a steganographic message. This steganalyzer searches for the measures that reflect the quality of distorted or degraded image signal vis-à-vis its original in an accurate, consistent and monotonic way. Such a measure, in the context of steganalysis, should respond to the presence of hidden message with minimum error, should work for a various embedding methods, and its reaction should be proportional to the length of the hidden message.

This paper has been organized as following sections: Section II describes some review works of image steganalysis. Section III describes the various methods for image feature selection. Section IV describes the design of steganalyzer. Experimental Results of the method has been discussed in Section V and Section VI analyses the results and VII draws the conclusion.

2. REVIEW OF RELATED WORKS ON IMAGE STEGANALYSIS METHOD

Research in *image steganalysis* has been spanned over the last two decades. Image steganalysis techniques can be grouped into two broad categories, namely, specific and universal approach. The specific steganalysis techniques are designed for a targeted embedding technique and worked by first analyzing the embedding operation and next step is to identify some features of the cover image that become modified as a result of the embedding process. The design of specific steganalysis techniques requires detailed knowledge of the steganographic embedding process and results a very accurate decisions when they are used against the particular steganographic technique.

Universal Approaches of image based steganalysis

A universal steganalytic [8] approach usually adopts a learning based strategy involving a training as well as a testing stage. The process has been illustrated in Figure 1. In this process, a feature extraction step is required which is used in both training and testing stage. This feature extraction step is used to map an input image from a high-dimensional image space to a low-dimensional feature space. The training stage results a trained classifier. Out of many effective classifiers, like Fisher linear discriminant (FLD), support vector machine (SVM), k-nearest neighbor (KNN), neural network (NN), etc., any one can be chosen. Decision boundaries are created by the classifier to separate the feature space into positive regions and negative regions with the help of the generated feature vectors extracted from the training images. In the testing stage, with the help of the trained classifier with a specific decision boundary, an image can be classified according to its feature vector's domination in the feature space. If the feature vector idenfies a

region where the classifier is labeled as positive, the testing image is classified as a positive class or the stego image. Otherwise, it is classified as a negative class or the cover image.In the following some typical universal steganalytic features has been described.

Image Quality Features (IQM) : Almost all the steganographic methods may cause degradation in more or less form of the carrier image. Objective image quality measures (IQMs) are the quantitative metrics based on the image features for examining this sort of distortion. The statistical evidence generated by steganographic methods may be identified based on a set of IQMs and can be used for detection also [9]. In order to search for some specific quality measures which are sensitive, consistent and monotonic to steganographic distortions, the analysis of variance (ANOVA) technique can also be exploited. The ranking of the goodness of the quantitative metrics is implemented according to the F-score generated from the ANOVA tests. The identified metrics can be used as feature sets to distinguish between cover and stego images.

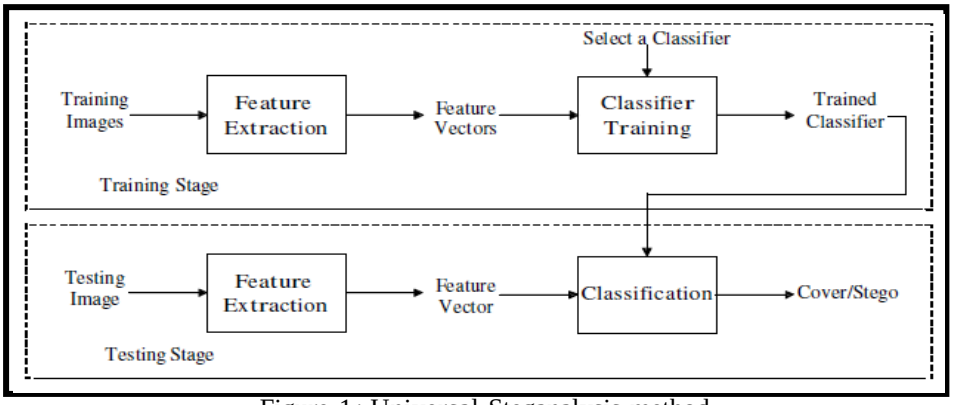


Figure 1: Universal Steganalysis method

Calibration Based Feature: Fridrich et al. [10] developed a feature-based classification technique incorporated with some calibration process to generate a blind steganalysis technique specific to JPEG images. Some parameters of the cover image may be approximately recovered with the help of the stego image as the side information through the calibration process which in turn increases the features sensitivity to the embedding modifications while suppressing image-to-image variations.

Moment Based Feature: The impact of image steganography can be regarded as introducing some noise in the cover image. Some statistics of the image may be changed with the introduction of noise. These changes reflected heavily on wavelet the domain. From this concept Lyu and Farid [11] uses the assumption that the PDF of the wavelet sub band coefficients and that of the prediction error of the sub band coefficients will change after data embedding. For example in a 3-level wavelet decomposition [12], the first four PDF moments, i.e., mean, variance, skewness, and kurtosis, of the sub band coefficients at each high-pass orientation (horizontal, vertical and diagonal direction) of each level are taken into consideration as one set of features. The same kinds of PDF moments of the difference between the logarithm of the sub band coefficients and the logarithm of the coefficients' cross-sub band linear predictions at each high-pass orientation of each level computed may be considered as another set of features. These two kinds of features yield satisfactory results at high embedding rate.

Correlation Based Feature: Local correlation pattern of an image may get disturbed after data embedding. In this case the inter-pixel dependency of a spatial image, and the intra-block or inter-block DCT coefficient dependency of a JPEG image correlation may be referred as correlation. Sullivan et al. [13] modeled the inter-pixel dependency by Markov chain and depicted it though a gray-level co-occurrence matrix (GLCM).

In this context, the working principle of the three most widely used universal techniques namely BSM (Binary Similarity Measures) [14], WBS (Wavelet Based Steganalysis) [15] and FBS (Feature-Based Steganalysis) [16] has been discussed. Binary Similarity Measures (BSMs) [14] has been developed by Avcibas et al. where distinguished features are obtained from the spatial domain representation of the image. The authors conjecture that correlation between the contiguous bit planes decreases after a message is embedded in the image. More specifically, the method looks at seventh and eight bit planes of an image and calculates three types of features, which include computed similarity differences, histogram and entropy related features, and a set of measures based on a neighborhood-weighting mask.

Lyu and Farid proposed a different approach for feature extraction from images known as WBS [15]. In their opinion most of the specific steganalysis techniques concentrate on first order statistics, i.e., histogram of DCT coefficients, but simple countermeasures could keep the first-order statistics intact, which

in turn makes the steganalysis technique inactive. They proposed building a model for natural images by using higher order statistics and shows that images with messages embedded in them deviate from this model.

Fridrich proposed a new approach of image steganalysis FBS [16] in which a set of distinguishing features are obtained from DCT and spatial domains. As the main component of the proposed approach, is used to estimate statistics of the original image, before embedding, estimation is simply done by decompressing the JPEG image and then cropping its spatial representation by four lines of pixels in both horizontal and vertical directions. Afterward, the image is JPEG recompressed with the original quantization table. The difference between statistics obtained from the given JPEG image and its original estimated version are obtained through a set of functions that operate on both spatial and DCT domains.

3. IMAGE FEATURE SELECTION

Because the dimensionality of image data is normally huge, it is unrealistic to use the image data directly for steganalysis. A feasible approach is to extract certain amount of data from the image and use them to represent the image itself for steganalysis. In other words, they are features characterizing the image. Different tasks decide the different relation of features with respect to image. For example in the area of facial recognition, the features should reflect the shape of target faces in an image, i.e. the main content of the image. Minor distortions should not affect the final decision. However, in steganalysis, the main content of an image is not an issue to be considered since human eyes cannot tell the difference between an original image and its stego-version. On the contrary, those minor distortions introduced during data hiding stand up as the first priority. Therefore, the features for steganalysis should reflect those minor distortions associated with data hiding.

Moments based Image Feature

To construct the features of both cover and stego or suspicious images several moments of the image series has been computed. In mathematics, a moment is, loosely speaking, a quantitative measure of the shape of a set of points. The "second moment", for example, is widely used and measures the "width" of a set of points in one dimension or in higher dimensions measures the shape of a cloud of points as it could be fit by an ellipsoid. Other moments describe other aspects of a distribution such as how the distribution is skewed from its mean, or peaked. There are two ways of viewing moments [17], one based on statistics and one based on arbitrary functions such as $f(x)$ or $f(x, y)$.

Statistical view

Moments are the statistical expectation of certain power functions of a random variable. The most common moment is the mean which is just the expected value of a random variable as given in 1.

$$\mu = \mathrm{E}[X] = \int_{-\infty}^{\infty} x\, f(x)\, dx \quad (1)$$

where $f(x)$ is the probability density function of continuous random variable X. More generally, moments of order $p = 0, 1, 2, \ldots$ can be calculated as $m_p = \mathrm{E}[X^p]$.These are sometimes referred to as the raw moments. There are other kinds of moments that are often useful. One of these is the central moments $\mu_p = \mathrm{E}[(X-\mu)^p]$.The best known central moment is the second, which is known as the variance given in 2.

$$\sigma^2 = \int (x-\mu)^2\, f(x)\, dx = m_2 - \mu_1^2 \quad (2)$$

Two less common statistical measures, skewness and kurtosis, are based on the third and fourth central moments. The use of expectation assumes that the pdf is known. Moments are easily extended to two or more dimensions as shown in 3.

$$m_{pq} = \mathrm{E}[X^p Y^q] = \iint x^p\, y^q\, f(x, y)\, dx\, dy \quad (3)$$

Here $f(x, y)$ is the joint pdf.

Estimation

However, moments are easy to estimate from a set of measurements, x_i. The p-th moment is estimated as given in 4 and 5.

$$m_p = \frac{1}{N} \sum_{i=1}^{N} x_i^p \quad (4)$$

(Often 1/*N* is left out of the definition) and the *p*-th central moment is estimated as

$$\mu_p = \frac{1}{N}\sum_i (x_i - \bar{x})^p \quad (5)$$

$\bar{x}$ is the average of the measurements, which is the usual estimate of the mean. The second central moment gives the variance of a set of data $s^2 = \mu_2$.For multidimensional distributions, the first and second order moments give estimates of the mean vector and covariance matrix. The order of moments in two dimensions is given by $p+q$, so for moments above 0, there is more than one of a given order. For example, m_{20}, m_{11}, and m_{02} are the three moments of order 2.

Non-statistical view

This view is not based on probability and expected values, but most of the same ideas still hold. For any arbitrary function $f(x)$, one may compute moments using the equation 6 or for a 2-D function using 7.

$$m_p = \int_{-\infty}^{\infty} x^p \, f(x)\,dx \quad (6) \qquad m_{pq} = \iint x^p \, y^q \, f(x,y)\,dx\,dy \quad (7)$$

In order to find the mean value of $f(x)$, one must use m_1/m_0, since $f(x)$ is not normalized to area 1 like the pdf. Likewise, for higher order moments it is common to normalize these moments by dividing by m_0 (or m_{00}). This allows one to compute moments which depend only on the shape and not the magnitude of $f(x)$. The result of normalizing moments gives measures which contain information about the shape or distribution (not probability dist.) of $f(x)$.

Digital approximation

For digitized data (including images) we must replace the integral with a summation over the domain covered by the data. The 2-D approximation is written in 8.

$$m_{pq} = \sum_{i=1}^{M}\sum_{j=1}^{N} f(x_i, y_j) x_i^p \, y_j^q = \sum_{i=1}^{M}\sum_{j=1}^{N} f(i,j) i^p \, j^q \quad (8)$$

If $f(x, y)$ is a binary image function of an object, the area is m_{00}, the x and y centroids are 9 and 10.

$$\bar{x} = m_{10}/m_{00} \quad (9) \qquad \bar{y} = m_{01}/m_{00}. \quad (10)$$

Invariance

In many applications such as shape recognition, it is useful to generate shape features which are independent of parameters which cannot be controlled in an image. Such features are called invariant features. There are several types of invariance.For binary connected components, this can be achieved simply by using the central moments, μ_{pq}. If an object is not at a fixed distance from a fixed focal length camera, then the sizes of objects will not be fixed. In this case size invariance is needed. This can be achieved by normalizing the moments as given in 11.

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}}, \text{ where } \gamma = ½(p+q)+1. \quad (11)$$

M.K. Hu [29] derived a transformation of the normalized central moments to make the resulting moments rotation invariant as given in 12.

$$\begin{aligned}
p+q &= 2 \\
\phi_1 &= \eta_{20} + \eta_{02} \\
\phi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
p+q &= 3 \\
\phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (\eta_{03} - 3\eta_{21})^2 \\
\phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{03} + \eta_{21})^2 \\
\phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\
&\quad + (\eta_{03} - 3\eta_{21})(\eta_{03} + \eta_{21})[(\eta_{03} + \eta_{21})^2 - 3(\eta_{12} + \eta_{30})^2] \\
\phi_6 &= (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
&\quad + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{03} + \eta_{21}) \\
\phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\
&\quad + (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[(\eta_{03} + \eta_{21})^2 - 3(\eta_{30} + \eta_{12})^2]
\end{aligned} \quad (12)$$

Zernike moments

For an image (or other) function $f(r,\theta)$, the Zernike moments are given in equation 13.

$$A_{nl} = < f, V_{nl} >$$
$$\approx \frac{n+1}{\pi} \iint_{r \le 1} f(r,\theta) V_{nl}^{*}(r,\theta)\, r\, dr\, d\theta$$
$$\approx \frac{n+1}{\pi} \Delta x \Delta y \sum_{i} \sum_{j} f(x_i, y_j) V_{nl}^{*}(x_i, y_j) \quad (13)$$

Where $x = r\cos(\theta)$ and $y = r \sin(\theta)$, and the function f must be rescaled so that it is contained in the unit circle. There is more than one way to compute Zernike moments. One can pre compute and store in a table all the values of $V_{nl}(x_i, y_j)$, but this requires a fixed number of samples in the image and the table may be quite large. A more flexible way, which does not need to store the values in a table, is to first compute the centered ordinary (geometric) moments of $f(x_i, y_j)$, then apply a transformation to the moments.

$$m_{pq} = \Delta x \Delta y \sum_{i} \sum_{j} f(x_i, y_j)(x_i - x_c)^p (y_j - y_c)^q \quad (14)$$

where x_c and y_c are the centers which may be obtained from the average (i.e., the centroid), median, or midrange of function f over its finite extent, whichever is the most robust for the application. Since the Zernike moments are defined only within a unit circle, either f must be rescaled to fit inside the unit circle, or better, the geometric moments can be normalized by the radius α of a bounding circle with center at (x_c, y_c) which contains all of f:

$$M_{pq} = \frac{m_{pq}}{\alpha^{p+q+2}} \quad (15)$$

The complex Zernike moments are as in 16.

$$A_{nl} = \frac{n+1}{\pi} \sum_{\substack{k=l \\ n-k=\text{even}}}^{n} \sum_{j=0}^{q} \sum_{m=0}^{l} (-i)^m \binom{q}{j} \binom{l}{m} B_{nlk} M_{k-2j-m,\,2j+m} \quad (16)$$

where $0 \le l \le n$, $q = (k - l)/2$.

Other Similarity based Image feature

Besides the moments based image features some other image similarity measure parameters are also taken as features for designed the steganalysis. They are namely MSE, PSNR, SSIM, Shannon's Entropy, DC Coefficient, DCT Distance, NAE and Average Difference. The peak signal-to-noise ratio (PSNR) is the ratio between a signal's maximum power and the power of the signal's noise. In statistics, the mean squared error (MSE) of an estimator is one of many ways to quantify the difference between values implied by an estimator and the true values of the quantity being estimated.

Assume a cover image C (i,j) that contains N by N pixels and a stego image S(i,j) where S is generated by embedding / mapping the message bit stream. Mean squared error (MSE) of the stego image as follows:

$$MSE = \frac{1}{[N \times N]} \sum_{i=1}^{N} \sum_{j=1}^{N} [\, C(i,j) - S(i,j)]^2 \quad (17)$$

The PSNR is computed using the following formulae:

$$\text{PSNR} = 10 \log_{10} 255^2 / \text{MSE db.} \quad (18)$$

The structural similarity (SSIM) [18] index is a method for measuring the similarity between two images. SSIM is designed to improve on traditional methods like peak signal-to-noise ratio (PSNR) and mean squared error (MSE), which have proved to be inconsistent with human eye perception. The SSIM metric is calculated on various windows of an image. The measure between two images x and y of common size N x N is:

$$SSIM(x,y)=\frac{(2\mu_x\mu_y+c_1)(2\sigma_{xy}+c_2)}{(\mu_x^2+\mu_y^2+c_1)(\sigma_x^2+\sigma_y^2+c_2)} \quad (19)$$

Where μ_x the average of x , μ_y is the average of y, σ_x^2 the variance of x, σ_y^2 the variance of y ,σ_{xy} the covariance of x and y,$c_1=(k_1L)^2$, $c_2=(k_2L)^2$ two variables to stabilize the division with weak denominator. L is the dynamic range of the pixel-values and $k_1=0.01$ and $k_2=0.03$ by default.

Entropy is a measure of the uncertainty associated with a random variable. In this context, the term usually refers to the Shannon Entropy [19], which quantifies the expected value of the information contained in a message, usually in units such as bits. Shannon denoted the entropy *H* of discrete random variable *X* with possible values $\{x_1, \ldots, x_n\}$ as,

$$H(X)=\mathrm{E}(I(X)). \quad (20)$$

Here E is the expected value, and *I* is the information content of *X*. *I*(*X*) is itself a random variable. If *p* denotes the probability mass function of *X* then the entropy can explicitly be written as

$$H(X)=\sum_{i=1}^{n}p(x_i)\,I(x_i)=\sum_{i=1}^{n}p(x_i)\log_b\frac{1}{p(x_i)}=-\sum_{i=1}^{n}p(x_i)\log_b p(x_i) \quad (21)$$

where *b* is the base of the logarithm used.

The discrete cosine transform (DCT) represents an image as a sum of sinusoids of varying magnitudes and frequencies. The DCT has the property that, for a typical image, most of the visually significant information about the image is concentrated in just a few coefficients of the DCT. The two-dimensional DCT of an M-by-N matrix A is defined as follows:

$$B_{pq}=\alpha_p\alpha_q\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}A_{mn}\cos\frac{\pi(2m+1)p}{2M}\cos\frac{\pi(2n+1)q}{2N},\quad \begin{matrix}0\le p\le M-1\\0\le q\le N-1\end{matrix}$$

$$\alpha_p=\begin{cases}1/\sqrt{M}, & p=0\\ \sqrt{2/M}, & 1\le p\le M-1\end{cases}\qquad \alpha_q=\begin{cases}1/\sqrt{N}, & q=0\\ \sqrt{2/N}, & 1\le q\le N-1\end{cases} \quad (22)$$

The values B_{pq} are called the *DCT coefficients* of A. The DCT is an invertible transform, and its inverse is given by

$$A_{mn}=\sum_{p=0}^{M-1}\sum_{q=0}^{N-1}\alpha_p\alpha_qB_{pq}\cos\frac{\pi(2m+1)p}{2M}\cos\frac{\pi(2n+1)q}{2N},\quad \begin{matrix}0\le m\le M-1\\0\le n\le N-1\end{matrix}$$

$$\alpha_p=\begin{cases}1/\sqrt{M}, & p=0\\ \sqrt{2/M}, & 1\le p\le M-1\end{cases}\qquad \alpha_q=\begin{cases}1/\sqrt{N}, & q=0\\ \sqrt{2/N}, & 1\le q\le N-1\end{cases} \quad (23)$$

The inverse DCT equation can be interpreted as meaning that any M-by-N matrix A can be written as a sum of *MN* functions of the form

$$\alpha_p\alpha_q\cos\frac{\pi(2m+1)p}{2M}\cos\frac{\pi(2n+1)q}{2N},\quad \begin{matrix}0\le p\le M-1\\0\le q\le N-1\end{matrix} \quad (24)$$

These functions are called the *basis functions* of the DCT. The DCT coefficients B_{pq}, then, can be regarded as the *weights* applied to each basis function. Horizontal frequencies increase from left to right, and vertical frequencies increase from top to bottom. The constant-valued basis function at the upper left is often called the *DC basis function*, and the corresponding DCT coefficient B_{00} is often called the *DC coefficient*.

A lower value of Average Difference (AD) gives a "cleaner" image as more noise is reduced and it is computed using Eq. (25).

$$\text{Average Difference (AD)} = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}[f(i,j)-f'(i,j)] \quad (25)$$

where original image is *f(i,j)* and on the decompressed image is *f'(i,j).*

Normalized absolute error (NAE) computed by Eq. (26) is a measure of how far is the decompressed image from the original image with the value of zero being the perfect fit. [28]. Large value of NAE indicates poor quality of the image [28].

$$(\text{NAE}) = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\left|\,[f(i,j).f'(i,j)]\,\right|}{\sum_{i=1}^{M}\sum_{j=1}^{N}\left|\,f(i,j)\,\right|} \quad (26)$$

4. DESIGN OF STEGANALYZER

The Steganalysis technique proposed here to test the presence of the hidden message is the combination of statistical moments and invariant moments based analysis along with several other similarity based image features on the cover data and stego data series for the estimation of the presence of the secret message as well as the predictive size of the hidden message. Steganalysis approach has been designed here based on the above mentioned fact considering cover image data as the independent data series and stego image data as the dependent series data. From the experimental results it can be shown that with the introduction of secret message/increasing length of the secret message moments parameters also changes. This is the basis of proposed steganalyzer that aims to classify image signal as original and suspicious. In order to classify the signals as "cover" or "stego" based on the selected image quality features, authors tested and compared three types of classifiers, namely, k- nearest neighbor classifier with k=1, support vector machines and back propagation neural network classifier. Block diagram of the steganalyzer has been shown in figure 2.

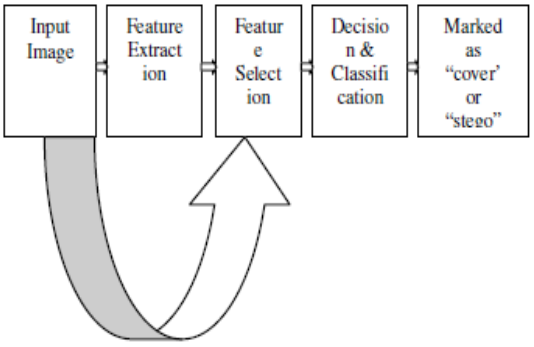


Figure 2: Block Diagram of the MSM steganalysis system

a) *Support Vector Machine Classifier*

In machine learning, **support vector machines** (**SVMs**, also **support vector networks**) [20, 21] are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. The basic SVM takes a set of input data and predicts, for each given input, which of two possible classes forms the input, making it a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other.The support vector method is based on an efficient multidimensional function optimization [20], which tries to minimize the empirical risk, which is the training set error. For the training feature data (f_i, g_i), $i = 1,..,N$, $g_i \in [-1,1]$, the feature vector F lies on a hyperplane given by $w^T F + b = 0$, where w is the normal to the hyperplane. The set of vectors is said to be optimally separated if it is separated without error and the distance between the closest vectors to the hyperplane is maximal. A separating hyperplane in canonical form, for the i'th feature vector and label, must satisfy the following constraints:

$$g_i\left[\left(wF_i\right)+b\right] \geq 1, \quad i = 1,2,....N \quad (27)$$

The distance *d(w,b;F)* of a feature vector *F* from the hyperplane *(w,b)* is,

$$d(w,b;F) = \frac{\left|w^T F + b\right|}{\left\|w\right\|} \qquad (28)$$

b) K Nearest Neighbor (KNN) Classifier

In pattern recognition, the *k*-nearest neighbor algorithm (*k*NN) is a method for classifying objects based on closest training examples in the feature space. *KNN* is a type of instance-based learning, or lazy learning where the function is only approximated locally and all computation is deferred until classification. The *k*-nearest neighbor algorithm is amongst the simplest of all machine learning algorithms: an object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its *k* nearest neighbors (*k* is a positive integer, typically small). If $k = 1$, then the object is simply assigned to the class of its nearest neighbor. The same method can be used for regression, by simply assigning the property value for the object to be the average of the values of its *k* nearest neighbors. It can be useful to weight the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. The *k*-nearest neighbor algorithm is sensitive to the local structure of the data. Nearest neighbor rules in effect compute the decision boundary in an implicit manner. It is also possible to compute the decision boundary itself explicitly, and to do so in an efficient manner so that the computational complexity is a function of the boundary complexity.

c) Back Propagation Feed Forward Neural Network Classifier

Any successful pattern classification methodology [22] depends heavily on the particular choice of the features used by that classifier .The Back-Propagation is the best known and widely used learning algorithm in training multilayer feed forward neural networks. The feed forward neural net refer to the network consisting of a set of sensory units (source nodes) that constitute the input layer, one or more hidden layers of computation nodes, and an output layer of computation nodes. The input signal propagates through the network in a forward direction, from left to right and on a layer-by-layer basis. Back propagation is a multi-layer feed forward, supervised learning network based on gradient descent learning rule. This BPN provides a computationally efficient method for changing the weights in feed forward network, with differentiable activation function units, to learn a training set of input-output data. A typical back propagation network of input layer, one hidden layer and output layer is shown in figure 3.

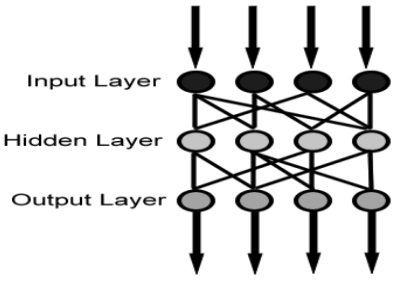


Figure 3: Feed Forward BPN

The proposed neural network based classifier is a two-layer feed-forward network, with sigmoid hidden and output neurons can classify vectors arbitrarily well, given enough neurons in its hidden layer. The network will be trained with scaled conjugate gradient back propagation.

d) Proposed Algorithm for Classification:

- Input: 50 image of various dimension of different types for training and 30 image of various dimension of different types for testing
- Select the cover images and extract various features
- Embed secret message based on five steganography tools (COX[23], MB[24], PQ[25], F5[26] and S–Tool[27])
- Repeat step 2 for stego images also.
- Store the results
- Create a sample data set based on the results.
- Create training data set for identifying cover/stego (cover=1, stego=0)

- Create training data set for identifying type of image.
- Create training data set for identifying the dimension of image.
- Test for classification and group them.
- Evaluate the performance of classifier based on ROC and Confusion Matrix
- The area under the ROC curve, also known as AUR, is calculated as the accuracy of the designed classifier

5. EXPERIMENTAL RESULTS

The steganalyzer has been designed based on a training set and using various image steganographic tools. The steganographic tools used here COX [23], MB [24], PQ [25], F5 [26] and S–Tool [27]. In the experiments 50 input image were used for training and 30 images for testing. These experiments are performed using a large data set of JPEG and BMP images obtained from publicly available websites. The image data set is categorized with respect to different features of the image to determine their potential impact on steganalysis performance. After embedding secret message into the cover image with the various embedding rate rates of 0.01, 0.02,….,0.1 with 0.01 in increments and from 0.1,0.2,…upto 1 with 0.1 increments ,various stego images has been created. From table 1 and table 2 it can be seen that with the introduction of small message all the statistical moments' value and invariant moments values changes. Table 3 shows the changes of different image similarity features at different embedding rate.

Table 1: Statistical Moments value of LENA 256x256 image at various embedding rate through S Tools

Insertion Rate (in %)	M1 |log|	M2 |log|	M3 |log|	M4 |log|	M5 |log|	M6 |log|	M7 |log|
0	6.4007	18.0374	26.1739	23.7762	49.3133	33.1483	48.9478
0.01	6.4006724	18.2037424	26.1741527	23.7762401	49.3133377	33.1483041	48.9479886
0.02	6.4006726	18.0373638	26.1738598	23.7762499	49.3133206	33.1482813	48.9478018
0.03	6.4006735	18.0373292	26.1739867	23.7762522	49.3133577	33.1482925	48.9478829
0.04	6.4006733	18.0373777	26.1739444	23.7762411	49.3133096	33.1482796	48.9478501
0.05	6.4006758	18.0374041	26.1740587	23.7762407	49.3134087	33.1483066	48.9478862
0.06	6.4006735	18.0374087	26.1738934	23.7762388	49.3131800	33.1483253	48.9478695
0.07	6.4006744	18.0374572	26.1739268	23.7762390	49.3132679	33.1482946	48.9478524
0.08	6.4006709	18.0374338	26.1737240	23.7762437	49.3131671	33.1483130	48.9477612
0.09	6.4006727	18.0374601	26.1739061	23.7762153	49.3132502	33.1482570	48.9477933
0.10	6.4006756	18.0373690	26.1738632	23.7762271	49.3133050	33.1482584	48.9477609
0.20	6.4006724	18.0373701	26.1737740	23.7762666	49.3132398	33.1483197	48.9478141
0.30	6.4006733	18.0373535	26.1740001	23.7762765	49.3132788	33.1483281	48.9479849
0.40	6.4006694	18.0373976	26.1737089	23.7762452	49.3131917	33.1482927	48.9477415
0.50	6.4006719	18.0373917	26.1739877	23.7762540	49.3133146	33.1483072	48.9478897
0.60	6.4006728	18.0373028	26.1739318	23.7762656	49.3134580	33.1482713	48.9478237
0.70	6.4006721	18.0374249	26.1741652	23.7762464	49.3132645	33.1483077	48.9480472
0.80	6.4006736	18.0374491	26.1738598	23.7762729	49.3131166	33.1483644	48.9479510
0.90	6.4006706	18.0374337	26.1742273	23.7762085	49.3133331	33.1482711	48.9479759
1.00	6.4006737	18.0373215	26.1736047	23.7762245	49.3131023	33.1482114	48.9476613

Table 2: Invariant Moments value of LENA 256x256 image at various embedding rate through S Tools

Insertion Rate (in %)	$\Phi_{2\|log\|}$	$\Phi_{3\|log\|}$	$\Phi_{4\|log\|}$	$\Phi_{5\|log\|}$	$\Phi_{6\|log\|}$	$\Phi_{7\|log\|}$
0	7.9137	10.4202	16.6387	20.1753	25.6352	29.6747
0.01	7.9136733	10.4203305	16.6386813	20.1753320	25.6351858	29.6747977
0.02	7.9136724	10.4203089	16.6386754	20.1752950	25.6351707	29.6747570

0.03	7.9136875	10.4202197	16.6387057	20.1752426	25.6352075	29.6747216
0.04	7.9136721	10.4202720	16.6386783	20.1752701	25.6351762	29.6747343
0.05	7.9131368	10.4202809	16.6386908	20.1752929	25.6351988	29.6747702
0.06	7.9136723	10.4202928	16.6386804	20.1752916	25.6351823	29.6747613
0.07	7.9136768	10.4202892	16.6386955	20.1753077	25.6352104	29.6747911
0.08	7.9136851	10.4202976	16.6387056	20.1753029	25.6352198	29.6747800
0.09	7.9136711	10.4202620	16.6386791	20.1752863	25.6351856	29.6747750
0.10	7.9136782	10.4202526	16.6386829	20.1752586	25.6351801	29.6747297
0.20	7.9136748	10.4202957	16.6386897	20.1753008	25.6352048	29.6747864
0.30	7.9136710	10.4202177	16.6386811	20.1752235	25.6351829	29.6747002
0.40	7.9136683	10.4202616	16.6386618	20.1752372	25.6351480	29.6746936
0.50	7.9136681	10.4202905	16.6386751	20.1753088	25.6351908	29.6748062
0.60	7.9136766	10.4202081	16.6386843	20.1752087	25.6351817	29.6746794
0.70	7.9136601	10.4203450	16.6386508	20.1753096	25.6351411	29.6747541
0.80	7.9136503	10.4202434	16.6386433	20.1752409	25.6351356	29.6747053
0.90	7.9136602	10.4202493	16.6386525	20.1752305	25.6351357	29.6746813
1.00	7.9136636	10.4203550	16.6386705	20.1753162	25.6351722	29.6747690

Table 3: Different features of LENA 256x256 image at various embedding rate through S Tools

Insertion Rate (in %)	Zernike 1	Zernike 2	MSE	RMSE	PSNR	SSIM	Shannon's ENTROPY	DC COEFF	DCT DIS	NAE	AVG DIF
0	10.1903	8.4111	0	0	99	1	7.5683	10.1371	10000	10000	10000
0.01	10.1902404000	8.4111570500	0.0017000000	0.0410000000	75.8817000000	0.9999951700	7.5682536700	10.1370514000	10.5970685600	1.7009E-05	3.4570E-03
0.02	10.1902404000	8.4112281400	0.0019000000	0.0438000000	75.2919000000	0.9999944500	7.5682265200	10.1370514000	10.2891996900	1.9483E-05	3.2015E-02
0.03	10.1902762000	8.4111860900	0.0024000000	0.0489000000	74.3366000000	0.9999926400	7.5682834100	10.1370537000	10.2867349800	2.4277E-05	-2.2888E-04
0.04	10.1902763000	8.4111289800	0.0021000000	0.0454000000	74.9923000000	0.9999942200	7.5682519200	10.1370513000	10.2172765000	2.0875E-05	1.5259E-05
0.05	10.1902404000	8.4111644100	0.0022000000	0.0467000000	74.7422000000	0.9999937800	7.5682943900	10.1370528000	10.4537599400	2.2112E-05	-1.3733E-04
0.06	10.1902404000	8.4110864800	0.0024000000	0.0493000000	74.2816000000	0.9999931200	7.5682437700	10.1370513000	10.3839656400	2.4586E-05	1.5259E-05
0.07	10.1902285000	8.4110301500	0.0022000000	0.0474000000	74.6224000000	0.9999940100	7.5682928400	10.1350519000	10.5064501400	2.2731E-05	-4.5776E-05
0.08	10.1902285000	8.4111431600	0.0027000000	0.0520000000	73.8159000000	0.9999926500	7.5682840600	10.1370491000	10.0740907700	2.7370E-05	2.2888E-05
0.09	10.1902763000	8.4112282800	0.0025000000	0.0500000000	74.1472000000	0.9999932700	7.5682757100	10.1375110000	10.3581451100	2.5359E-05	3.0518E-05
0.10	10.1902524000	8.4111573500	0.0024000000	0.0493000000	74.2816000000	0.9999932900	7.5682592300	10.1370534000	10.1465097100	2.4586E-05	-1.9836E-04
0.20	10.1902285000	8.4110722500	0.0027000000	0.0520000000	73.8159000000	0.9999917900	7.5682653200	10.1370522000	10.3628808000	2.7370E-05	-7.6294E-05
0.30	10.1902404000	8.4111925800	0.0030000000	0.0551000000	73.3071000000	0.9999924700	7.5683116400	10.1370522000	10.0065393900	3.0771E-05	-7.6294E-05
0.40	10.1902643000	8.4111221200	0.0036000000	0.0599000000	72.5849000000	0.9999897700	7.5682991100	10.1370500000	10.0162118900	3.6338E-05	1.3733E-04
0.50	10.1902285000	8.4111149400	0.0037000000	0.0605000000	72.4935000000	0.9999901900	7.5683030300	10.1370527000	10.1351865100	3.7111E-05	-1.2207E-04
0.60	10.1902643000	8.4112141700	0.0038000000	0.0619000000	72.2989000000	0.9999893400	7.5683187000	10.1370537000	10.1445091700	3.8812E-05	-2.2888E-04
0.70	10.1902285000	8.4110936900	0.0043000000	0.0652000000	71.8396000000	0.9999882500	7.5682542300	10.1370528000	9.8488986250	4.3142E-05	-1.3733E-05
0.80	10.1902285000	8.4110085800	0.0047000000	0.0689000000	71.3680000000	0.9999871400	7.5682580400	10.1370513000	9.9950717780	4.8090E-05	1.5259E-05
0.90	10.1902404000	8.4111080100	0.0050000000	0.0709000000	71.1236000000	0.9999855900	7.5682781800	10.1370494000	9.9509185390	5.0873E-05	1.9836E-04
1.00	10.1902763000	8.4111649300	0.0054000000	0.0732000000	70.8425000000	0.9999852800	7.5682587800	10.1370522000	9.9676781380	5.4275E-05	-7.6294E-05

In order to calculate the performance of three classifiers at different embedding rate and to show the relationship between the false-positive rate and the detection rate, authors have also calculated the receiver operating characteristics (ROC) curves of steganographic data embedding for the five different embedding methods. The ROC curves are calculated for the classifiers by first designing a classifier and then testing the data unseen to the classifier against the trained classifier. The testing results also consists of true positive (TP), true negative (TN), false positive (FP), and false negative (FN). The testing accuracy is calculated by (TP+TN) / (TP+TN+FP+FN). Figure 4 and 5 respectively shows the ROC analysis of KNN and SVM classifier at different embedding rate for S Tools. Performance comparison of different classifier at various embedding rate has been shown in table 4. A comparative study amongst various existing image steganalysis method with feature based steganalysis method has been shown in table 5.

Table 4: Performance measure of MSM image steganalyzer (AUR) at various embedding rate

Emb Rate (in %)	S-tool[27]			F5[26]			PQ[25]			MB[24]			COX[23]		
	KNN	SVM	BPN	KNN	SVM	BPN	KNN	SVM	BPN	KNN	SVM	BPN	KNN	SVM	BPN
0.01	75.00%	81.57%	46.63%	58.33%	53.13%	48.93%	62.55%	50.00%	36.67%	68.75%	50.00%	63.33%	63.33%	50.00%	56.67%
0.02	89.67%	90.62%	77.94%	58.33%	53.13%	53.39%	75.50%	62.67%	72.35%	68.75%	50.00%	72.30%	78.57%	62.50%	66.66%
0.03	90.62%	90.62%	77.94%	68.75%	65.05%	67.73%	75.50%	71.43%	72.35%	75.00%	62.75%	74.30%	78.57%	62.50%	74.66%
0.04	90.62%	90.62%	77.94%	69.05%	65.07%	67.74%	75.50%	74.60%	72.35%	75.00%	62.75%	74.30%	78.57%	62.50%	74.66%
0.05	90.62%	90.62%	77.94%	69.05%	65.14%	78.03%	75.50%	80.00%	78.90%	75.00%	62.75%	74.30%	78.57%	62.50%	74.66%
0.1	90.62%	90.62%	91.67%	71.90%	66.38%	78.03%	77.56%	87.50%	93.77%	75.65%	62.75%	78.00%	78.96%	62.50%	79.67%
0.2	90.62%	94.48%	91.67%	76.67%	66.66%	80.30%	77.56%	90.00%	93.77%	75.66%	62.75%	78.03%	79.34%	62.50%	80.03%
0.3	90.62%	97.06%	93.46%	78.76%	66.66%	81.52%	84.38%	92.56%	93.77%	75.66%	63.77%	78.16%	80.02%	66.66%	81.98%
0.4	91.67%	97.06%	93.85%	78.76%	66.34%	81.63%	84.38%	92.87%	93.77%	76.13%	63.95%	78.26%	80.51%	67.29%	82.38%
0.6	93.22%	97.06%	94.06%	78.76%	69.02%	82.0%	84.38%	93.76%	93.89%	77.13%	64.19%	79.66%	81.33%	69.05%	87.50%

6. DISCUSSIONS

Experimental results demonstrate that the proposed moments and similarity measure feature based steganalysis (MSM) method performs well for different image steganography tools as compared to various other existing methods. It clearly indicates that the information-hiding modifies the characteristics of the various moments of the images. From the ROC analysis it can be seen that the KNN and BPN classifier performs well for the presence of a very little amount of secret message of MSM method. This steganalyzer uses three classifiers for identifying the cover and stego ones which produces the superiority of this method compared to the other existing ones. The steganalysis performance in detecting S-Tools [27] based image steganograms is much better than the detection of the image steganograms produced by using other steganography tools. By employing some more features the steganalysis performance could be improved.

7. CONCLUSION

In this paper, an image based steganalysis technique is proposed and tested which is based on moments and other similarity measure feature based image distortion measurement. The de-noised version of the image object has been selected as an estimate of the cover-object. Next step is to use statistical, invariant and other similarity measure features to measure the distortion which is in turn used for designing the classifiers to determine the presence of hidden information in an image. The design of the image steganalyzer based on three classifiers is useful for find out the presence of very small amount of hidden information.

Results from simulations with numerous image series showed that the proposed steganalysis algorithm provides significantly higher detection rates than existing ones.

Table 5: Comparison of MSM image steganalyzer with other existing methods

Insertion Rate (in %)	LSB[27]						F5[26]						MB[24]						PQ[25]					
	BSM[14]	WBS[15]	FBS[16]	MSM(KNN)	MSM (SVM)	MSM (BPN)	BSM[14]	WBS[15]	FBS[16]	MSM(KNN)	MSM (SVM)	MSM (BPN)	BSM[14]	WBS[15]	FBS[16]	MSM(KNN)	MSM (SVM)	MSM (BPN)	BSM[14]	WBS[15]	FBS[16]	MSM(KNN)	MSM (SVM)	MSM (BPN)
0.05	68.42%	56.91%	97.30%	90.62%	90.62%	77.94%	51.52%	51.76%	71.32%	69.05%	65.14%	78.03%	50.11%	50.14%	53.35%	75.00%	62.75%	74.30%	75.36%	76.61	85.09	78.57%	62.50%	78.90%
0.1	78.28%	65.69%	99.35%	90.62%	90.62%	91.67%	50.56%	52.58%	77.12%	71.9%	66.38%	78.03%	50.85%	50.85%	57.06%	75.65%	62.75%	78.00%	75.5%	76.59%	85.55%	78.96%	62.50%	93.77%
0.2	87.30%	75.5%	99.71%	90.62%	94.48%	91.67%	51.76%	54.97%	85.59%	76.67%	66.66%	80.30%	51.53%	53.41%	64.65%	75.66%	62.75%	78.03%	75.5%	75.92%	85.79%	79.34%	62.50%	93.77%
0.4	92.50%	87.06%	99.80%	91.67%	97.06%	93.46%	53.86%	61.46%	93.27%	78.76%	66.66%	81.52%	53.62%	56.79%	79.09%	76.13%	63.95%	78.16%	76.9%	79.36%	86.96%	80.51%	67.29%	93.77%
0.6	93.27%	90.86%	99.80%	93.22%	97.06%	93.85%	NA	NA	NA	78.76%	66.34%	81.63%	56.4%	61.61%	87.29%	77.13%	64.19%	78.26%	NA	NA	NA	81.33%	69.05%	93.77%

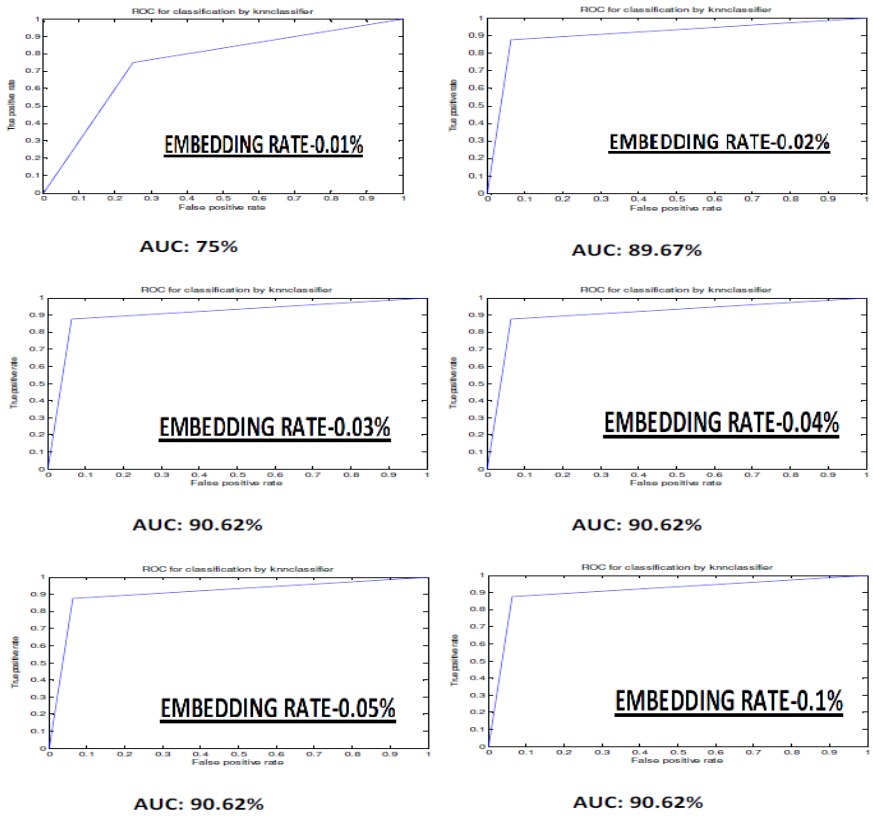


Figure 4: ROC curve of KNN classifier based steganalysis of S-Tools at different embedding rate

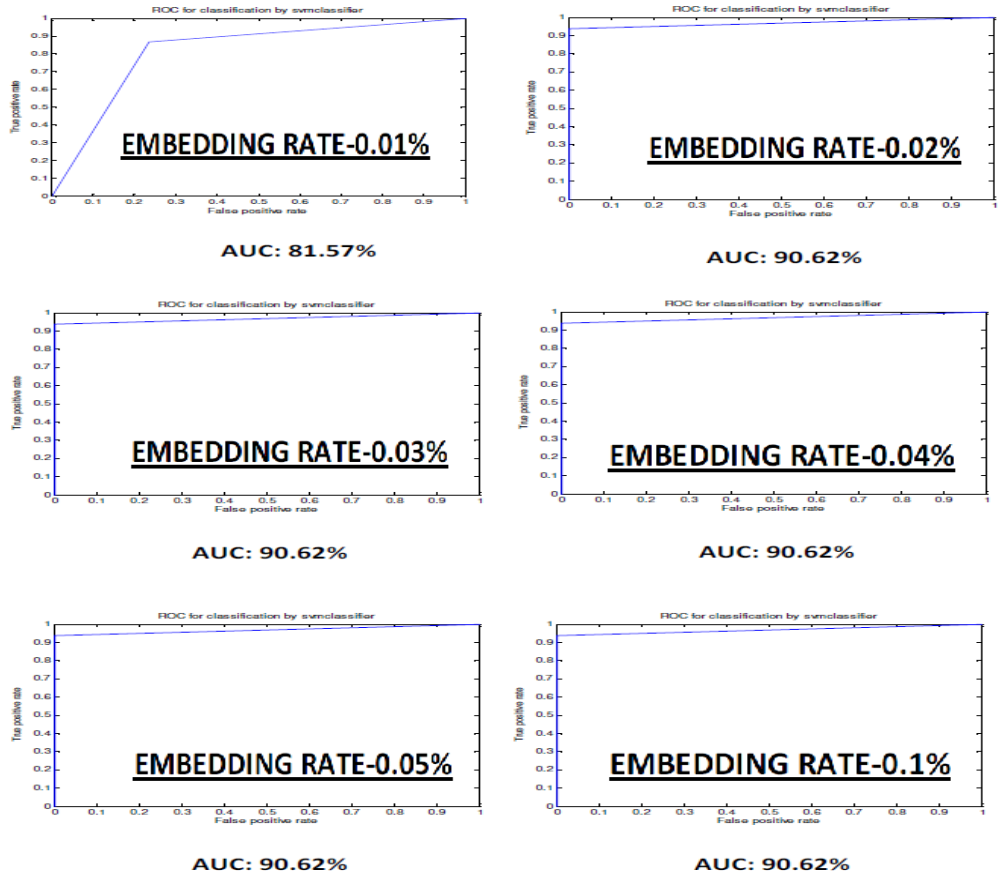


Figure 5: ROC curve of SVM classifier based steganalysis of S-Tools at different embedding rate

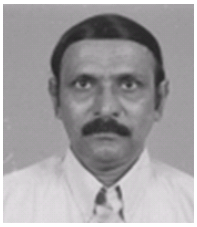
REFERENCES

[1] http://www.dia.unisa.it/~ads/corso-security/www/CORSO-0203/steganografia/LINKS%20LOCALI/Attacks.pdf
[2] Andreas Wetfied and Andreas Pfitzmann. Attacks on steganographic systems. In Proceedings of the Third Intl. Workshop on Information Hiding, Springer-Verlag., pages 61-76, 1999.
[3] www.informatik.htw-dresden.de/~fritzsch/VWA/Source/EzStego.java.
[4] Derek Upham. Jsteg, http://zooid.org/~paul/crypto/jsteg/.
[5] X. Y. Luo, D. S.Wang, P.Wang, and F. L. Liu. A review on blind detection for image steganography. Signal Processing, 88(9):2138-2157, 2008.
[6] N. Johnson and S. Jajodia. "Exploring Steganography: Seeing the Unseen", IEEE Computer, vol. 31, no.2, pp. 26-34, 1998.
[7] N. Provos and P. Honeyman. "Hide and Seek: An Introduction to Steganography", IEEE: Security & Privacy, vol. 1, pp. 32-44, 2003.
[8] X. Y. Luo, D. S.Wang, P.Wang, and F. L. Liu. A review on blind detection for image steganography. Signal Processing, 88(9):2138-2157, 2008.
[9] I Avcibas, N. Memon, and B. Sankur. Steganalysis using image quality metrics. IEEE Transactions on Image Processing, 12(2):221-229, 2003.
[10] J. Fridrich. Feature-based steganalysis for jpeg images and its implications for future design of steganographic schemes. In Proceedings of the 6th Information Hiding Workshop, volume 3200 of LNCS, pages 67-81. Springer, 2004.
[11] Lyu Siwei and H. Farid. Detecting hidden message using higher-order statistics and support vector machines. In Proceedings of the 5th Information Hiding Workshop, volume 2578 of LNCS, pages 131-142. Springer, 2002.
[12] Y. Q. Shi, C. Chen, and W. Chen. A Markov process based approach to effective attacking jpeg steganography. In Proceedings of the 8th Information Hiding Workshop, volume 4437 of LNCS, pages 249-264. Springer, 2006.
[13] K. Sullivan, U. Madhow, S. Chandrasekaran, and B. S. Manjunath. Steganalysis for markov cover data with applications to images. IEEE Transactions on Information Forensics and Security, 1(2):275-287, 2006.
[14] Avcibas, N. Memon, and B. sankur, Image steganalysis with binary similarity measures.," IEEE International Conference on Image Processing, Rochester, New York. , September 2002.

[15] H. Farid, "Detecting hidden messages using higher-order statistical models," in Proc. IEEE International Conference on Image Processing (ICIP '02), vol. 2, pp. 905-908, Rochester, NY, USA, September 2002.
[16] J. Fridrich, Feature-based steganalysis for jpeg images and its implications for future design of steganographic schemes," Proc. 6th Information Hiding Workshop, Toronto, Canada, May 23-25 , 2004
[17] MOMENTS IN IMAGE PROCESSING Bob Bailey Nov. 2002
[18] IEEE Alan Conrad Bovik Fellow IEEE Hamid Rahim Sheikh Student Member IEEE Zhou Wang, Member and IEEE. Eero P. Simoncelli, Senior Member. Image quality assessment: From error visibility to structural similarity. IEEE TRANSACTIONS ON IMAGE PROCESSING, 3, 2004.
[19] Claude E. Shannon. A mathematical theory of communication. The Bell System Technical Journal. 27:379-423.
[20] P. Pudil, J. Novovicova and J. Kittler, "Floating Search Methods in Feature Selection," Pattern Recognition Letters,vol. 15, no. 11, pp. 1119 - 1125, November 1994.
[21] Vapnik, V., The Nature of Statistical Learning Theory. Springer, New York, 1995.
[22] http://www.emilstefanov.net/Projects/NeuralNetworks.aspx.
[23] I.J. Cox, J. Kilian, T. Leighton and T. Shamoon: Secure Spread Spectrum Watermarking for Multimedia. IEEE Trans. on Image Processing, Vol.6 (1997) 1673-1687
[24] P. Sallee, Model-based steganography," International Workshop on Digital Watermarking, Seoul, Korea.2003.
[25] J. Fridrich, M. Goljan, and D. Soukal, Perturbed quantization steganography with wet paper codes," ACM Multimedia Workshop, Magdeburg, Germany, September 20-21 , 2004.
[26] Wetfeld, F5a steganographic algorithm: High capacity despite better steganalysis," 4th International Workshop on Information Hiding. , 2001.
[27] A.Brown,S-ToolsVersion4.0,Copyright©1996,http://members.tripod.com/steganography/stego/s-tools4
[28] THE PERFORMANCE OF FRACTAL IMAGE COMPRESSION ON DIFFERENT IMAGING MODALITIES USING OBJECTIVE QUALITY MEASURES by SUMATHI POOBAL and G.RAVINDRAN at International Journal of Engineering Science and Technology (IJEST) Vol. 3 No. 1 Jan 2011.
[29] M. K. Hu, "Visual pattern recognition by moment invariants," IRE Trans. Information Theory, vol. 8, pp. 179–187, 1962.

BIOGRAPHY OF AUTHORS

Souvik Bhattacharyya received his B.E. degree in Computer Science and Technology from B.E. College, Shibpur, India, presently known as Bengal Engineering and Science University (BESU) and M.Tech degree in Computer Science and Engineering from National Institute of Technology, Durgapur, India. Currently he is working as an Assistant Professor in Computer Science and Engineering Department at University Institute of Technology, The University of Burdwan. Presently he is pursuing his PhD from NIT Durgapur. He has a very good no of research publication in his credit. His areas of interest are Natural Language Processing, Network Security and Image Processing.



Gautam Sanyal has received his B.E and M.Tech degree National Institute of Technology (NIT), Durgapur, India. He has received Ph.D (Engg.) from Jadavpur University, Kolkata, India, in the area of Robot Vision. He possesses an experience of more than 25 years in the field of teaching and research. He has published nearly 72 papers in International and National Journals / Conferences. Three Ph.Ds (Engg) have already been awarded under his guidance. At present he is guiding six Ph.Ds scholars in the field of Steganography, Cellular Network, High Performance Computing and Computer Vision. He has guided over 10 PG and 100 UG thesis. His research interests include Natural Language Processing, Stochastic modeling of network traffic, High Performance Computing, Computer Vision. He is presently working as a Professor in the department of Computer Science and Engineering and also holding the post of Dean (Students' Welfare) at National Institute of Technology, Durgapur, India. He is a member of IEEE also.