

RESEARCH ARTICLE

# Extreme learning machine based optimal embedding location finder for image steganography

Hayfaa Abdulzahra Atee<sup>1,2\*</sup>, Robiah Ahmad<sup>2</sup>, Norliza Mohd Noor<sup>2</sup>, Abdul Monem S. Rahma<sup>3</sup>, Yazan Aljeroudi<sup>4</sup>

**1** Foundation of Technical Education, Higher Education and Scientific Research, Baghdad, Iraq, **2** Department of Engineering, UTM Razak School of Engineering and Advanced Technology, UTM Kuala Lumpur, Kuala Lumpur, Malaysia, **3** Computer Science Department, University of Technology, Baghdad, Iraq, **4** Department of Mechanical Engineering, International Islamic University of Malaysia, Kuala Lumpur, Malaysia

☯ These authors contributed equally to this work.

✉ Current address: Department of Engineering, UTM Razak School of Engineering and Advanced Technology, UTM Kuala Lumpur, Kuala Lumpur, Malaysia

\* [haifaa\\_atee@yahoo.com](mailto:haifaa_atee@yahoo.com), [haifaaate@gmail.com](mailto:haifaaate@gmail.com)



## Abstract

In image steganography, determining the optimum location for embedding the secret message precisely with minimum distortion of the host medium remains a challenging issue. Yet, an effective approach for the selection of the best embedding location with least deformation is far from being achieved. To attain this goal, we propose a novel approach for image steganography with high-performance, where extreme learning machine (ELM) algorithm is modified to create a supervised mathematical model. This ELM is first trained on a part of an image or any host medium before being tested in the regression mode. This allowed us to choose the optimal location for embedding the message with best values of the predicted evaluation metrics. Contrast, homogeneity, and other texture features are used for training on a new metric. Furthermore, the developed ELM is exploited for counter over-fitting while training. The performance of the proposed steganography approach is evaluated by computing the correlation, structural similarity (SSIM) index, fusion matrices, and mean square error (MSE). The modified ELM is found to outperform the existing approaches in terms of imperceptibility. Excellent features of the experimental results demonstrate that the proposed steganographic approach is greatly proficient for preserving the visual information of an image. An improvement in the imperceptibility as much as 28% is achieved compared to the existing state of the art methods.

## OPEN ACCESS

**Citation:** Atee HA, Ahmad R, Noor NM, Rahma AMS, Aljeroudi Y (2017) Extreme learning machine based optimal embedding location finder for image steganography. PLoS ONE 12(2): e0170329. doi:10.1371/journal.pone.0170329

**Editor:** Zhaohong Deng, Jiangnan University, CHINA

**Received:** July 28, 2016

**Accepted:** January 3, 2017

**Published:** February 14, 2017

**Copyright:** © 2017 Atee et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Data Availability Statement:** All relevant data are within the paper and its Supporting Information files.

**Funding:** The authors received no specific funding for this work.

**Competing interests:** The authors have declared that no competing interests exist.

## Introduction

Over the decades, the ever-escalating advancements of communication technology allowed the free transferring and sharing of confidential information over the complex internet network. This free sharing of sensitive information in the form of data files, and video/audio recordings

posed severe security threats. The preservation of users' privacy is repeatedly threatened by the highly sophisticated and deceptive phishing attacks. Thus, absolute protection of sensitive data communication from unauthorized accesses or attacks is demanded.

Presently, the secured communication is achieved via mathematical models assisted cryptographic and steganographic techniques. Ironically, cryptography being the encryption of a plain-text for generating the cipher-text does not obscure the data existence. It rather makes the data incomprehensible to protect the secret message from attacks or unauthorized access. For absolutely secured information communication, the limitations of cryptography are surmounted by introducing a new technique called steganography. However, most of the conventional steganographic techniques suffer from high computational loads when selecting the best location for concealing the message in the host medium with minimal deformation. This shortcoming can be overcome by introducing the neural network (NN) based steganographic technique, where the NN uses a distributed representation to store the learning knowledge. Thus, accessing the concealed data without knowing the topology of the NN appears practically infeasible [1]. Although some researchers prefer models with interpretability power such as explicit mathematical or statistical models or even heuristically encoded models such as fuzzy models, it has been proved that black box type of models when learning is feasible have more capability of capturing complicated knowledge and proving functionality in real world type of systems [2][3][4]. Such black box models have dramatically proved high efficiency in the state of the art of speech recognition, visual object recognition and many other fields [5].

Using information-hiding protocols, the steganographic technique embeds the message into a cover medium to keep the hidden data from being detected. This cover medium may be an image, video, or audio file. Among various steganographic techniques image steganography (concealing data into an image) is most popular and widely used because it allows an easy exchange of vast amount of images via the internet [6]. On top, the image steganography assisted hidden data cannot be recognized through the visual inspection [7]. Lately, in the image steganography domain the heuristic searching optimization became attractive [5]. Despite much research achieving an efficient steganographic algorithm for finding the best embedding location with reduced computational time expenses remains challenging.

Depending on embedded locations, the image steganographic algorithms are categorized into spatial[8][9] and frequency domain embedding. The later one is also called transform-domain embedding [10]–[13]. In the spatial domain, the least significant bit (LSB) based steganography [8][9] is the most extensively used method [14], where the carrier or cover image LSB is applied to conceal the secret message. Conversely, in the least significant bit replacement (LSBR) based steganography, the hidden secret message can be uncovered by the existing steganalysis methods [15][16]. Thus, it is weak against visual and statistical attacks. The least significant bit matching (LSBM) method also called  $\pm$  embedding method provides better security than LSBR. However, it is incompatible for most of the model-preserving steganographic techniques [17]. Despite their high capacity the spatial-domain techniques are not robust against image-processing operations, noise attacks, lossy compression, and filtering. Furthermore, they offset the statistical properties of the image due to the sole usage of the BMP format.

As aforementioned, in frequency-domain steganography the secret data are concealed in the significant parts of the cover image. This domain is comprised of several transforms such as discrete cosine transforms (DCT), discrete wavelet transforms (DWT), and discrete Fourier transforms (DFT). These transforms are used as media for hiding a message into an image [18]. Although both DWT and DCT have relatively smaller capacities but the former one is superior in terms of robustness against image-processing operations, statistical and noise attacks as well as distortion [19]. Thus, the steganographic techniques in the frequency-domain possess better immunity to attacks than the one spatial-domain. The limitations

involving the spatial-domain techniques are overcome using frequency-domain. Numerous researches are performed with DWT [10], [12]. The presence of rounding error in the inverse DFT make it disadvantageous for steganographic applications [20]. Table 1 presents a brief summary of embedding the secret information in spatial or frequency domain.

Some researchers have combined the spatial and frequency domains. The [21][22] introduced a framework for optimizing the adaptive distortion function to achieve minimal statistical detectability. The [23] improved the detection percentage and classified the images as stego or clean. Furthermore, spatial or frequency domain techniques are integrated with other techniques including artificial NN (ANN), genetic algorithm (GA), or both to attain enhanced steganographic performances. Spatial-domain based GAs are used [1], [24] to minimize the distortion and. GA and ANN are used [25] to accelerate the training speed. Frequency-domain ANN is used [26] to augment the embedding capacity. Spatial domain based ANN is utilized [27] to realize good approximation capacity, faster convergence, and a more stable performance surface. This type of ANN is also used [28] to increase the approximation capacity and minimize distortion.

The ANN is also used with steganography for message embedding [25], where the secret message is assumed to represent an image. This allowed the steganographer to change the message data freely provided the visual information is preserved. However, this assumption is not applied to the text messages. Meanwhile, ANN is also used for digital watermarking to authenticate the image [29], in which concealing a secret message is not required [30]. ANN is employed for the capacity maximization [28], steganographic content detection [31–33], identification of the embedded data in an image when applied to steganalysis or as a classifier and determination of the lower and upper bounds of embedding capacity [34]. Likewise, GAs are used in steganography for diverse purposes. GA is used to model the steganography problem [24] for search and optimization. Besides, for optimization with minimum distortion the GAs are utilized, where a stego image closer to the cover image is obtained [1], [35]. The [11] presents DCT with Markov as a detection and classifier for images. Table 2 summarizes different embedding techniques with combined spatial and frequency domains.

Lately, the learning ability of NNs is exploited to expand the optimization potential of conventional data-hiding techniques. In steganography, ANN is used either for the classification of the stego image or for the detection of the embedded data in an image. We intend to reduce the distortion in a stego image as much as possible by appropriately selecting the location in the image for messages embedment. Theoretically, an ELM demonstrates a good

**Table 1. The embedding domain for the existing state of the art methods.**

Author(s)	Domain and Technique	Pros	Cons
Banerjee, Bhattacharyya, and Sanyal 2013	Spatial-LSB	Capable of extracting the secret message without the cover image	Capacity issue has not addressed
Pevny, Filler, and Bas 2010	Spatial-LSB	Allows the embedder to conceal seven times longer message with same security	Applied theoretically and did not test by real data such as text or images
Wu, Hsien C. et al. 2009	Spatial-LSB	High payload in cover image	Unsatisfied image quality
Luo, Huang, and Huang 2010	Spatial-LSB	The visual quality and security have been improved significantly compared to conventional LSB	Did not tested against image processing or statistical analysis
Islam and Gupta 2014	Spatial-LSBM	Better security than LSBR	Conflicting for most of the model-preserving steganographic techniques
Abdelwahab and Hassaan 2008	Frequency-DWT	Does not require the original cover image to extract the embedded secret image.	Did not tested for text into image.
Prabakaran and Bhavani 2012	Frequency-DWT	Hiding a large-size secret image into a small-size cover image.	The quality of stego-image is not satisfied.

doi:10.1371/journal.pone.0170329.t001

**Table 2. The combined spatial and frequency domains with different embedding techniques for the existing state of the art methods.**

Author(s)	Domain and Technique	Pros	Cons
Tomás Filler and Fridrich 2011	Frequency-DCT and Spatial	Strong against many types of steganalysis	High complexity
Tom Filler, Judas, and Fridrich 2011	Frequency-DCT and spatial	The methods are not limited to binary embedding and allow the embedder to choose the amplitude of embedding changes dynamically based on the cover-image content.	Focus on payload aspects rather than embedding
Pathak and Selvakumar, 2014	Frequency-DCT and Spatial	It is used as a classifier and embedding.	This method omitted some features of images.
Iranpour and Rahmati 2014	Spatial and GA	Enhancing the security by minimize the distortion.	Omitted the optimum number of blocks as well as their sizes.
EI-Emam and AL-Zubidy 2013	Frequency GA and ANN	Allowed the steganographer to change the message data freely provided the visual information is preserved.	Omits the text steganography.
Tsai et al. 2009	Frequency and ANN	Augment the embedding capacity and supports true-color secret image with size constraint on shares.	Hiding small image into large image.
Husien and Badi 2014	Spatial and ANN	Good approximation capacity, faster convergence, and more stable performance surface.	Did not present numerical comparisons with other works.
Ghaleb Al-jbara, Mat Kiah, and Jalab 2012	Spatial-LSB and ANN	Increases the approximation capacity.	PSNR and MSE are not satisfied and did not tested against image processing.
EI-Alfy 2013	Spatial domain-PVD and ANN	99% rates of detection have been achieved.	Applied only in transformed domain.
Pratt, Konda, and Chu 2008	Spatial-LSB, and ANN	It is especially challenging when the embedding rate is low, such as below 10 percent of all embedded data.	It is used as a steganalysis and not as embedding. Some error rates have been addressed in extracting the embedded data.
Nazeri and Kanan 2014	Spatial domain and GA	It is modeling the steganography problem as a search and optimization problem.	Did not tested against image processing or any statistical analysis attack.
Roy and Laha 2015	Spatial- LSB and GA	High security and robustness.	The image quality (PSNR) is not satisfied.
Cho, Seongho, Byung-Ho Cha, Martin Gawecki, and C.-C. Jay Kuo 2013	Frequency-DCT and Markov	Tested in terms of spatial and frequency domains	Using as a classifier not as embedding

doi:10.1371/journal.pone.0170329.t002

generalization performance and universal approximation at extremely fast learning speeds. It can be used for either classification or regression purposes [36]. Inspired by such notable advantages, we propose an ELM-based supervised mathematical model called Optimal Embedding Location Finder (OELF) for image steganography. In addition, a novel fusion metric (*fusion1*) is introduced for the training in the regression mode to realize the best performance metric for steganography. Another novel fusion metric (*fusion2*) is developed for evaluating the results. To the best of our knowledge, for the first time we use the machine learning to determine the best location with least sensitive area for embedding.

This paper is organized as follows. Section 2 depicts the proposed OELF model. Section 3 highlights the detail mathematical background of steganography. Section 4 describes the proposed methodology. Section 5 explains the experimental results with various attributes. Section 6 concludes the paper with further outlook.

### Optimal Embedding Location Finder (OELF) model

Most traditional steganographic methods embed the message into an image by ignoring the significance of the image’s spatial features. Nevertheless, the identification of best embedding location is critically decided by the message homogeneity and other texture features [37] of the blocks. A location having least image distortion is considered to be the optimum one. To

protect the embedding process from a steganalysis, any form of distortion in the image must be minimized after the payload is inserted. Furthermore, the cover image and stego image must be approximately identical both visually and statistically. The selected area and the embedding method are the primary factors that affect the distortion. Based on OELF model an ELM is proposed for finding the best embedding location. It is worth noting that ELM is beneficial due to its universal approximation capacity which allows rapid training with good over-fitting avoidance than other classical NN based approaches [36]. Thus, a modified ELM is used to train a single-hidden-layer NN with a varying number of neurons. Appendix A provides a short depiction of ELM.

### Background of steganography modeling

As mentioned earlier, OELF locates the most suitable window for embedding the secret message into the image without affecting its visual features. Initially, the image is partitioned into (8 × 8) block pixels and one bit of the message is inserted into each block. Depending on the message size, the image is then partitioned into overlapping square windows to embed the message. The features of contrast (*C*), energy (*Enr*), homogeneity (*H*), entropy (*Ent*), correlation (*Corr*), standard deviation (*Std*), and the mean (*M*) of each square window are calculated using:

$$C = \sum_{i,j} |i - j|^2 p(i, j) \tag{1}$$

$$Enr = \sum_{i,j} p(i, j)^2 \tag{2}$$

$$H = \sum_{i,j} \frac{1}{1 - (i - j)^2} p(i, j) \tag{3}$$

$$Ent = -\sum_{i,j} p(i, j) \log(p(i, j)) \tag{4}$$

$$Corr = \frac{cov(Cover\_image, Stego\_image)}{\|Cover\_image\| \|Stego\_image\|} \tag{5}$$

where *i* and *j* are the horizontal and vertical pixel coordinates, respectively, and *p* is the pixel value.

$$cov(x, y) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))(y_i - E(y)) \tag{6}$$

where *N* is the number of the window pixels.

$$Std = \sigma_{xy} = \sqrt{cov(x, y)} \tag{7}$$

$$M = E(x) = \frac{1}{N} \sum_{i=1}^N x_i; E(y) = \frac{1}{N} \sum_{i=1}^N y_i \tag{8}$$

After calculating the window features and embedding the message in the corresponding window, the resultant imperceptibility is represented using one of three metrics including correlation, MSE, and SSIM. The expression for MSE and SSIM yields:

$$MSE = \frac{1}{N \times M} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} [Cover\_image(i, j) - Stego\_image(i, j)]^2 \tag{9}$$

where  $N$  and  $M$  are the length and width of the image, respectively.

$$SSIM = \frac{(2\mu_x\mu_y + C_1)(\sigma_{xy}^2 + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{10}$$

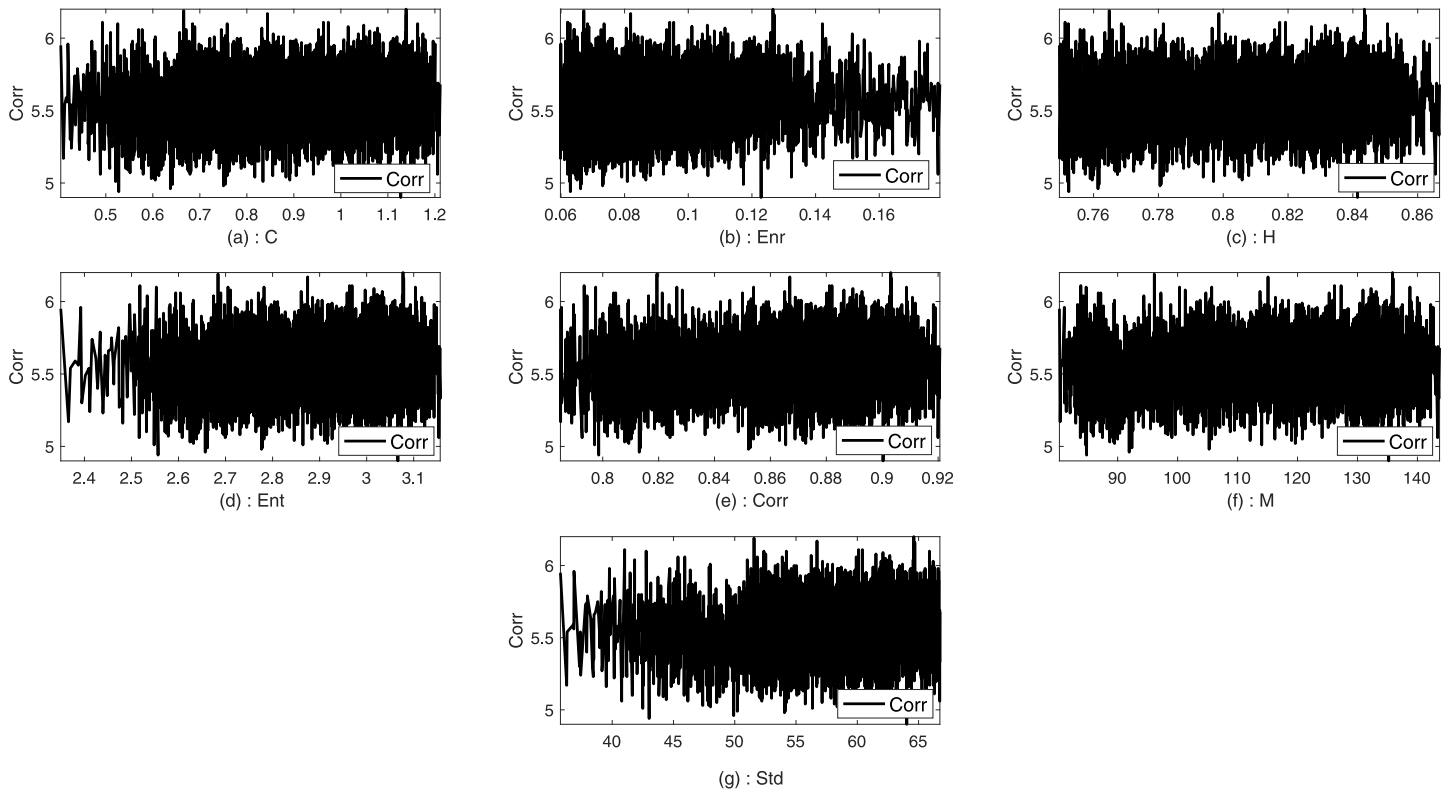
where  $\mu_x$  and  $\mu_y$  are the local mean,  $\sigma_x$  and  $\sigma_y$  are the standard deviation,  $\sigma_{xy}$  is the cross-covariance,  $C_1$  and  $C_2$  are constants.

### Methodology

The following subsections describe the detailed methodology including the input (host or cover image), the message to be embedded in the image, the output (stego image) and the evaluation metrics of imperceptibility.

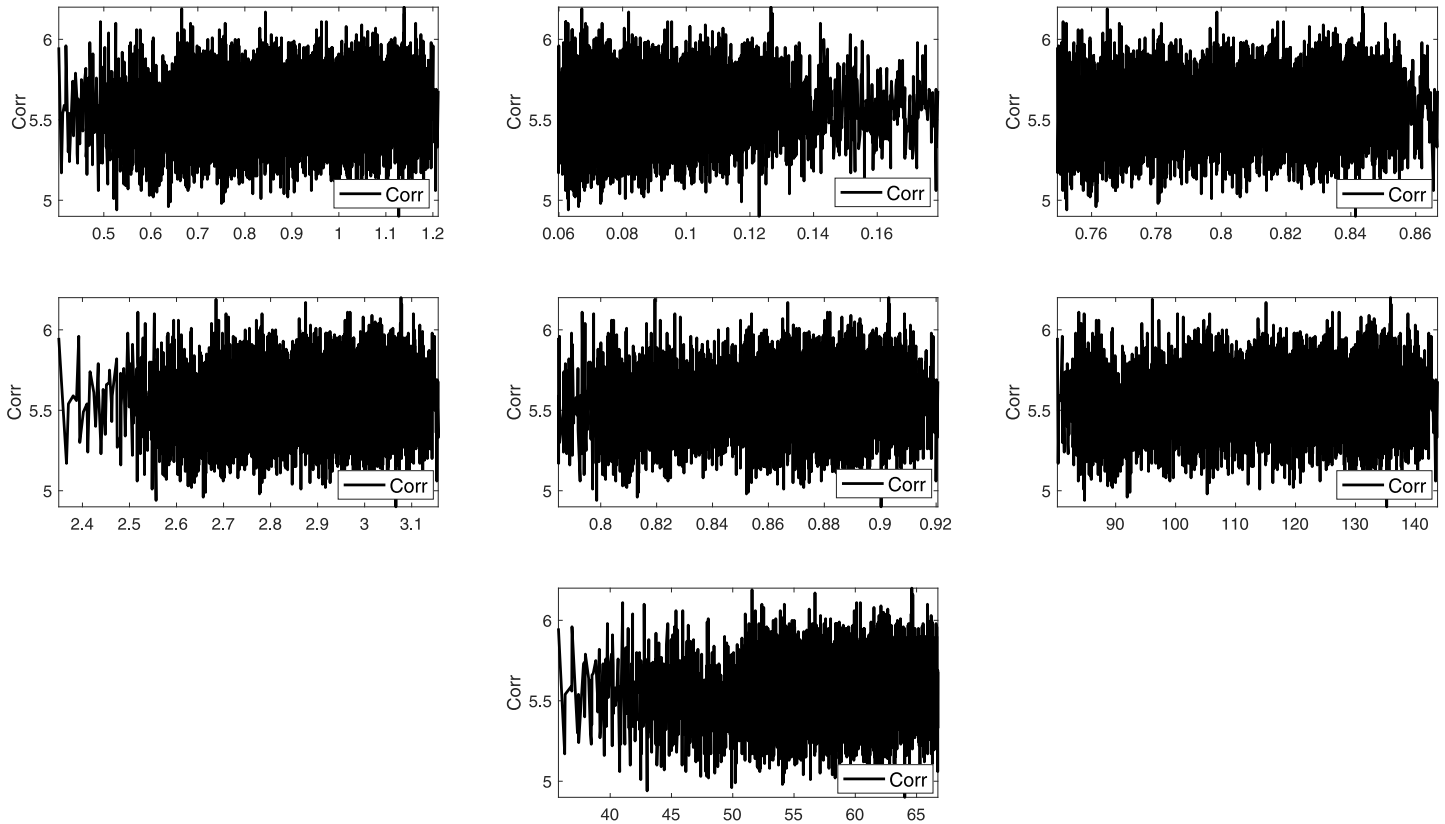
#### Input and output determination

Two images such as Lena and Sails from the standard database are used to analyze the trends between the imperceptibility and the texture features of the image. Imperceptibility is measured in terms of correlation, MSE, and SSIM between two corresponding square windows for the host and stego images with respect to the extracted features. Figs 1–6 show the trends of the imperceptibility of the Lena and Sails images after the message is embedded into a square window regarding the corresponding texture features in this window. It is evident that all the features (contrast, energy, homogeneity, entropy, correlation, entropy, and Std) are strongly



**Fig 1. Relationship of the correlation metric to the texture features (a) contrast, (b) energy, (c) homogeneity, (d) entropy, (e) correlation, (f) mean, and (g) standard deviation for Lena image.**

doi:10.1371/journal.pone.0170329.g001



**Fig 2. Relationship of the MSE metric to the texture features (a) contrast, (b) energy, (c) homogeneity, (d) entropy, (e) correlation, (f) mean, and (g) standard deviation for Lena image.**

doi:10.1371/journal.pone.0170329.g002

correlated. The occurrence of less variability in the imperceptibility correlation with respect to the set of features implies their equivalent usage in the machine learning model.

Table 3 summarizes the trends of imperceptibility to texture feature.

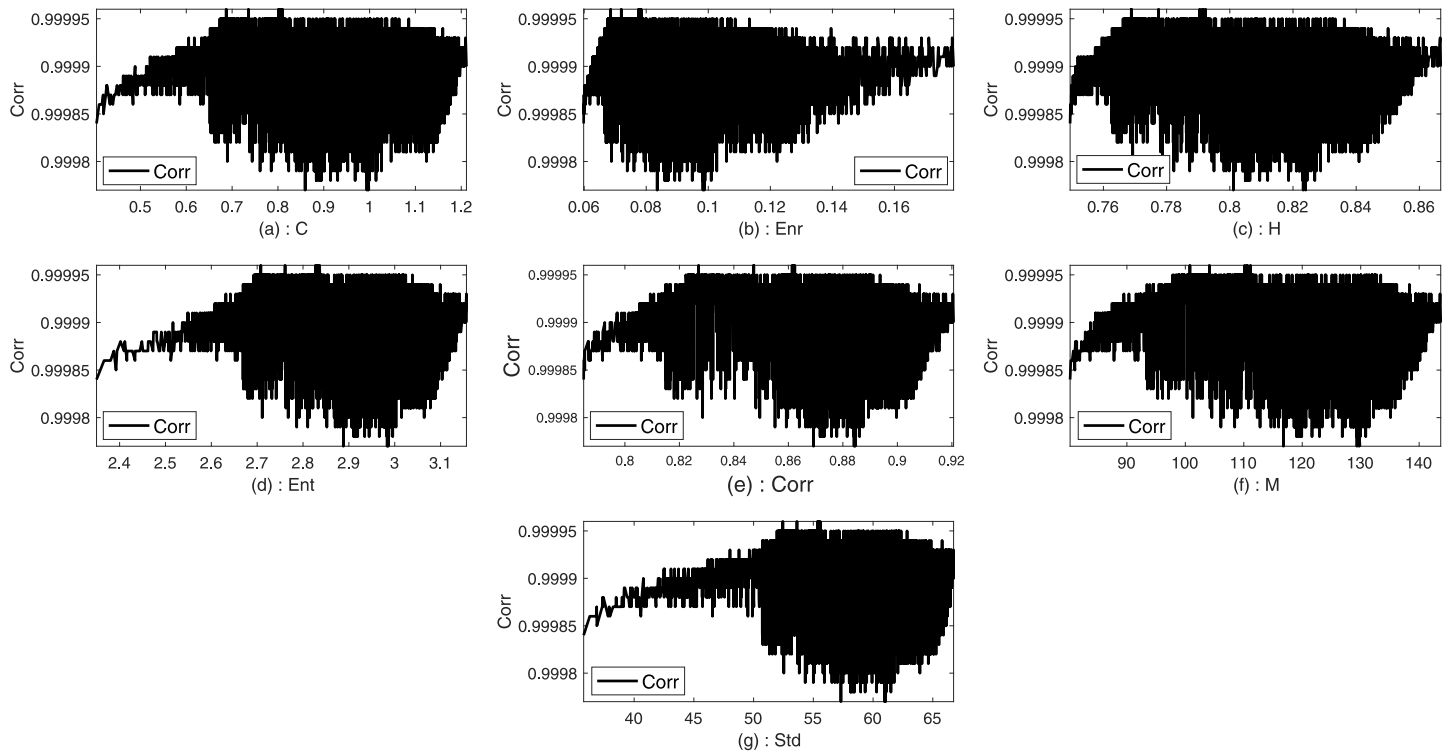
A detail analysis of such trends between the imperceptibility and the texture features of the image allowed us to determine the possible causality among them. Thus, the machine learning is designed with an optimized embedder or steganographer.

### Model design

The following steps are adopted to develop the proposed model:

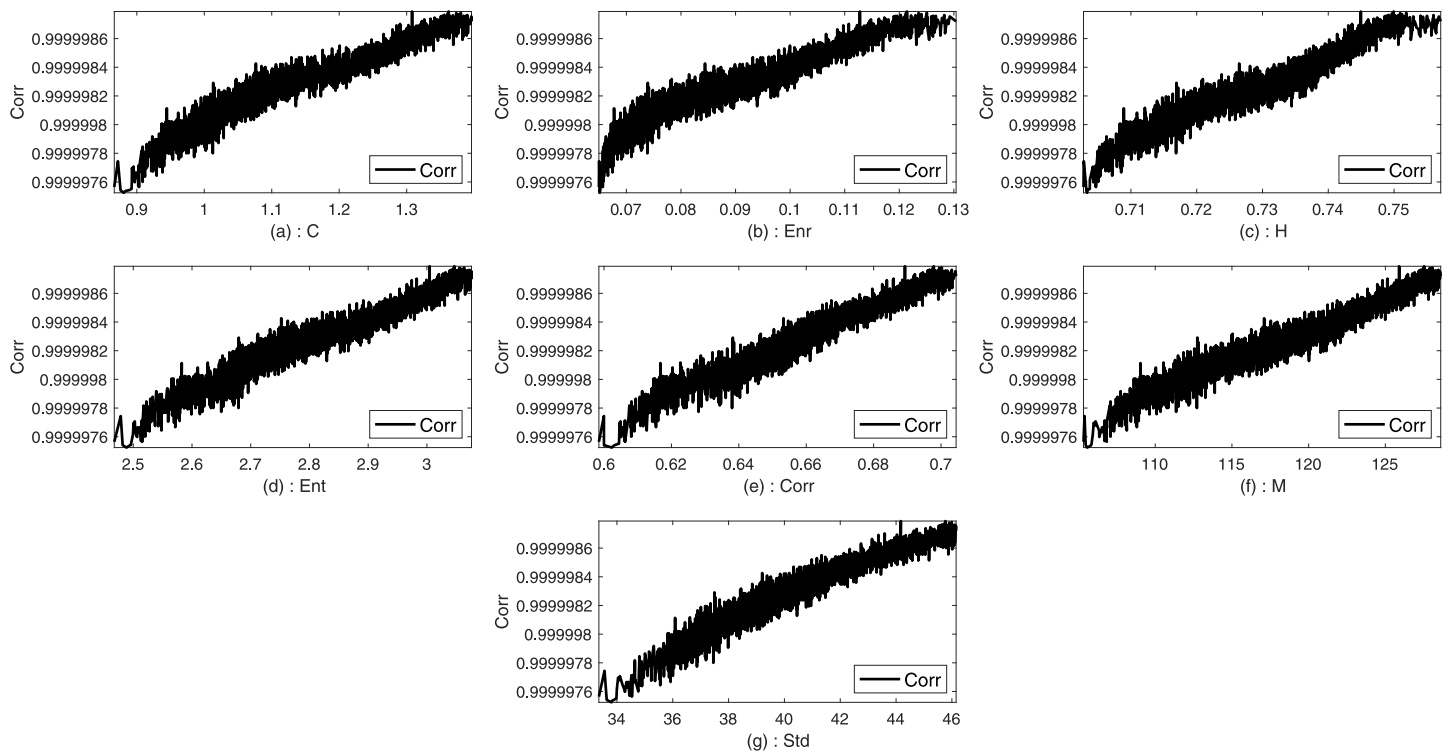
1. Partitioning of the  $(N \times M)$  host image into  $(K \times L)$  pixel non-overlapping sub-blocks, where  $(K = L = 8)$ .
2. Determination of the number of blocks needed to embed the message according to the message bits' size  $m$ .
3. Determination of the minimum square window size (SWS) from the image that contains the required blocks. The SWS is calculated using:

$$SWS = 8 \lceil \sqrt{m} \rceil \times 8 \lceil \sqrt{m} \rceil \text{ Pixels} \tag{11}$$



**Fig 3. Relationship of the SSIM metric to the texture features (a) contrast, (b) energy, (c) homogeneity, (d) entropy, (e) correlation, (f) mean, and (g) standard deviation for Lena image.**

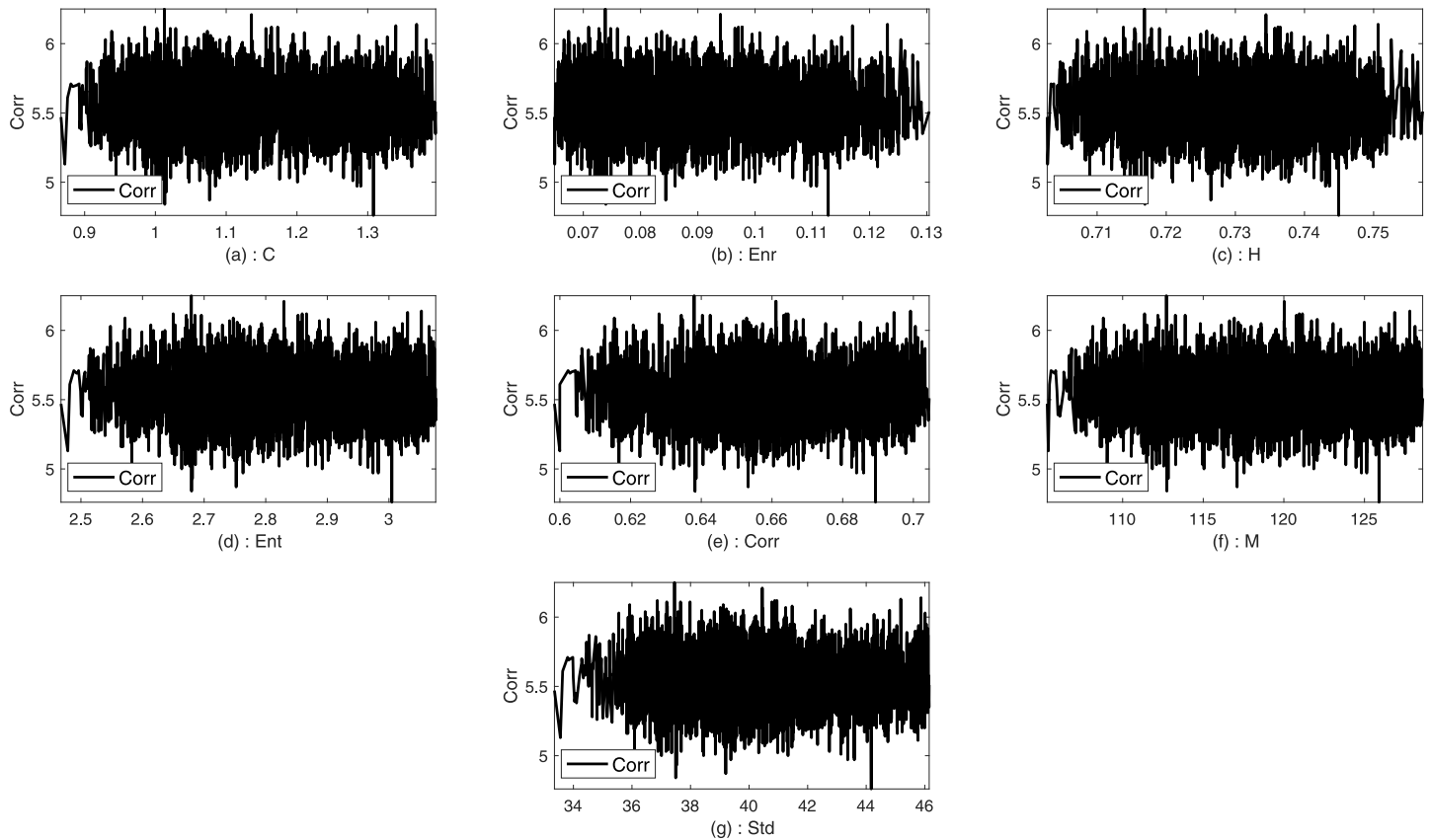
doi:10.1371/journal.pone.0170329.g003



**Fig 4. Relationship of the correlation metric to the texture features (a) contrast, (b) energy, (c) homogeneity, (d) entropy, (e) correlation, (f) mean, and (g) standard deviation for Sails image.**

doi:10.1371/journal.pone.0170329.g004





**Fig 5. Relationship of the MSE metric to the texture features (a) contrast, (b) energy, (c) homogeneity, (d) entropy, (e) correlation, (f) mean, and (g) standard deviation for Sails image.**

doi:10.1371/journal.pone.0170329.g005

4. Creation of raw data set of the square windows with a scanning resolution of 4 pixels NOS. The size of the data set is:

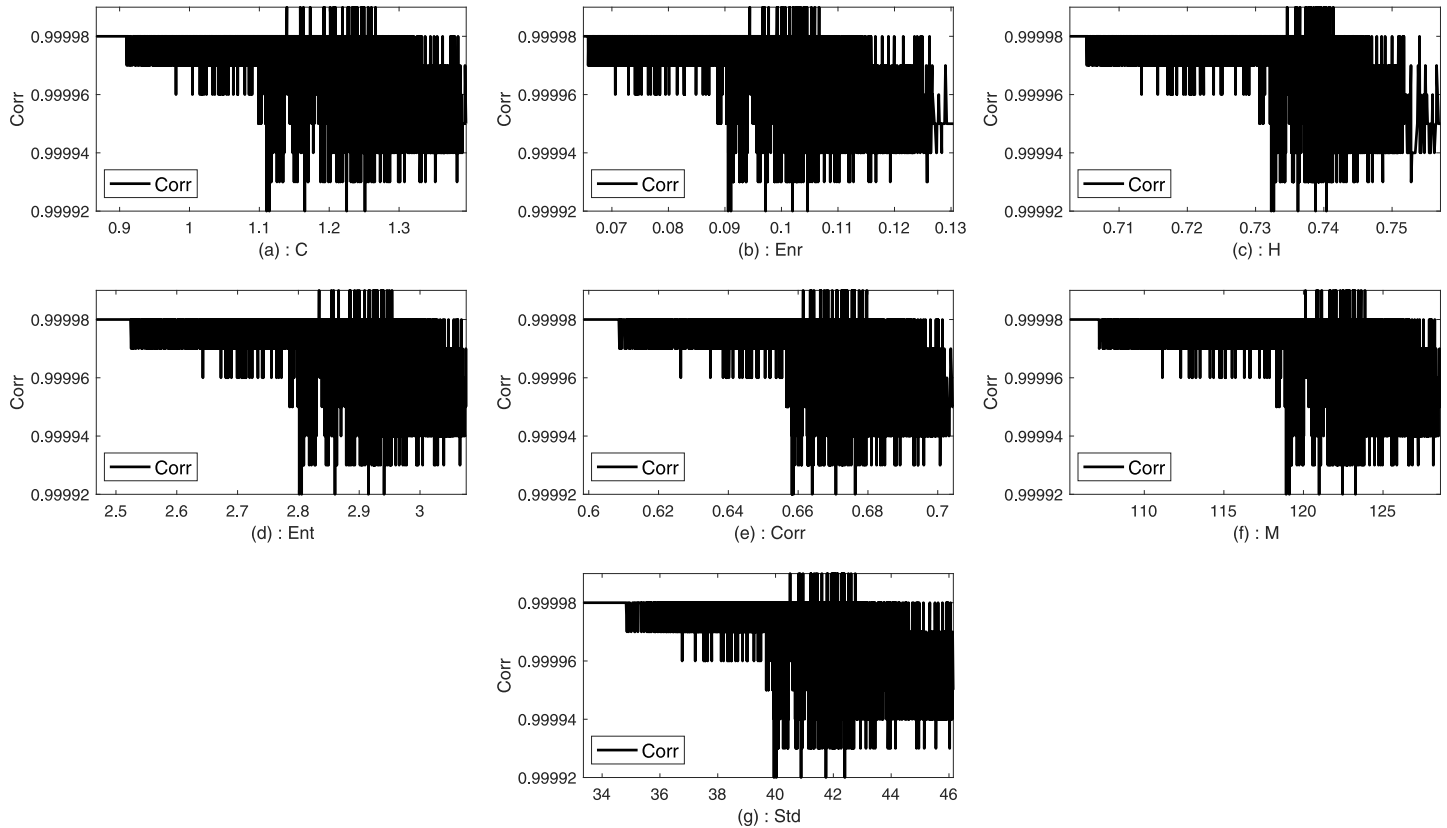
$$NOS = \left( \left( \frac{N - SWS}{4} \right) + 1 \right) \times \left( \left( \frac{M - SWS}{4} \right) + 1 \right) \quad (12)$$

where  $N$  and  $M$  are the length and width of the image, respectively, and  $SWS$  is the square window size.

### Data set preparation

Fig 7 illustrates the schematic framework for the creation of the learning data set and the feature domain prior to the ELM training and testing. The texture feature extraction, metric calculation and embedding are performed for building the learning data set. It is customary to explain briefly the embedding and the feature extraction procedure.

**Wavelet transform based embedding.** As aforementioned, the message must be embedded into its corresponding square window for each square window in the data set. The learning data are extracted from the raw data set using the embedding process and the calculation of the resultant visual imperceptibility metrics. To achieve this goal, the following steps are executed:



**Fig 6. Relationship of the SSIM metric to the texture features (a) contrast, (b) energy, (c) homogeneity, (d) entropy, (e) correlation, (f) mean, and (g) standard deviation for Sails image.**

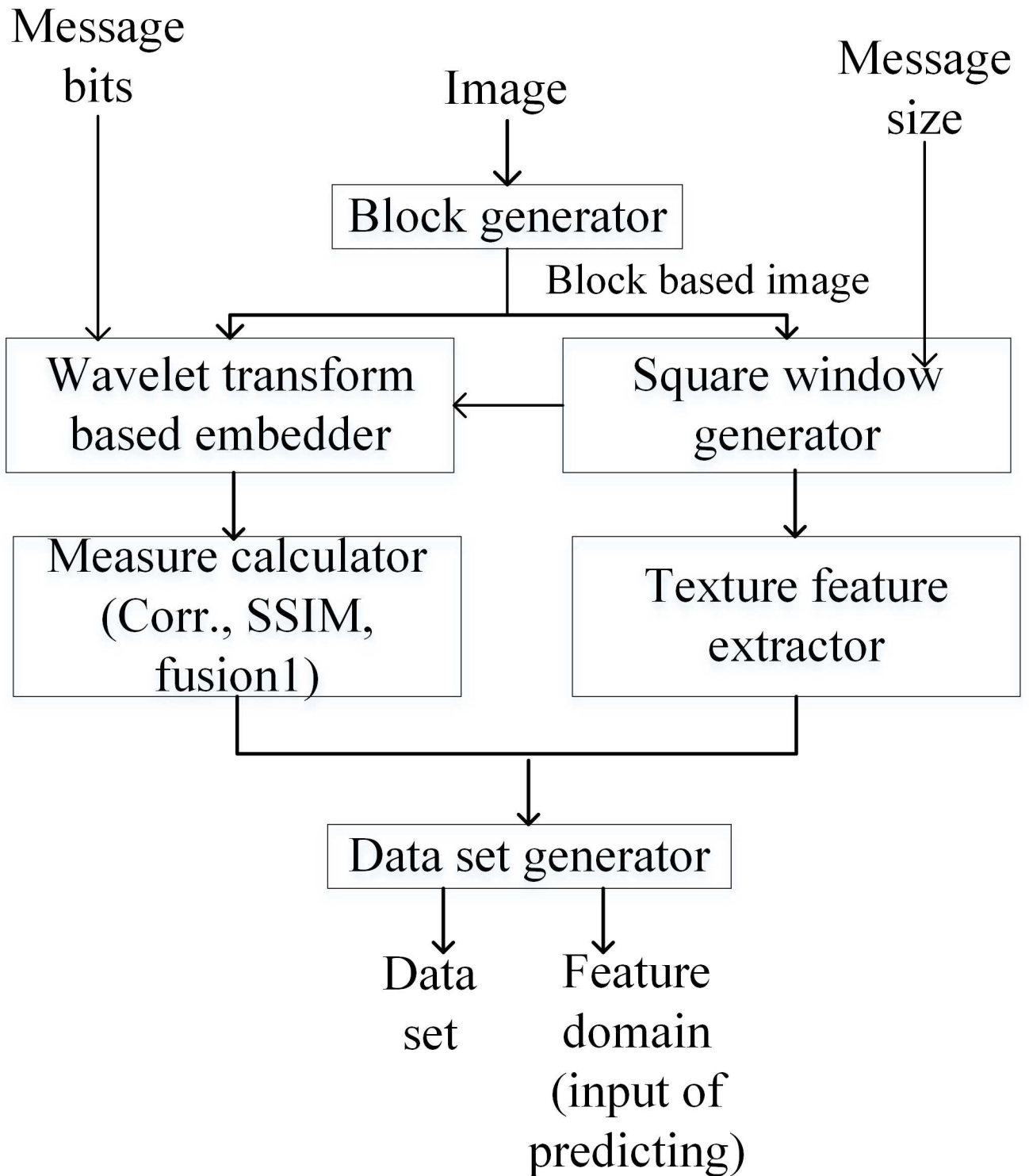
doi:10.1371/journal.pone.0170329.g006

1. For message bit one, the value is quantized to the nearest even number with the index (8, 8) in the corresponding block. Otherwise, it is quantized to the nearest odd number.
2. The wavelet transform for each sub-block is computed by inverting the wavelet.
3. The wavelet is transformed to its corresponding spatial domain.
4. The embedding process is repeated until the final bit of the message is embedded.

**Table 3. Trends of the imperceptibility to the texture feature for the Lena and Sails images.**

Features	Measures		
	Correlation	MSE	SSIM
<b>Contrast</b>	Positive	No trend	Negative
<b>Energy</b>	Positive	No trend	Negative
<b>Homogeneity</b>	Positive	No trend	Negative
<b>Correlation</b>	Positive	No trend	Negative
<b>Mean</b>	Positive	No trend	Negative
<b>Standard deviation</b>	Positive	No trend	Negative
<b>Entropy</b>	Positive	No trend	Negative

doi:10.1371/journal.pone.0170329.t003



**Fig 7. Construction of data set and feature domain.**

doi:10.1371/journal.pone.0170329.g007

- For each square window, the corresponding visual metrics are calculated. These metrics include correlation, MSE, SSIM, and fusion1. The expression for *fusion1* yields:

$$fusion1 = correlation \times SSIM \tag{13}$$

**Texture feature extraction.** The texture features are extracted using the following steps:

- The co-occurrence matrix is built for each square window whose sub-blocks are used for embedding the message bits.
- The feature function (contrast, energy, homogeneity, entropy, correlation, and standard deviation) of the co-occurrence matrix is computed for each square window. The expression for *features* yields

$$features = (C, Enr, H, Ent, Corr, M, Std) \tag{14}$$

## Extreme Learning Machine (ELM)

**ELM training.** The prepared data is represented the matrix form

$$X = (f_{1j}, f_{2j}, \dots, f_{7j}; y_{ij}), \text{ with } j = 1, \dots, n$$

where *n* is the number of square windows,  $f_{1j}, f_{2j}, \dots, f_{7j}$  are the extracted features,  $y_{ij}$  is the corresponding output metrics, and  $i = 1, 2, 3, 4$  correspond to the Corr, MSE, SSIM, and fusion1, respectively.

A neural network of  $\tilde{n}$  hidden neurons is built and trained on a part of *X* to predict  $y_i$ . Furthermore, the training and the testing phases are validated using the RMSE before applying the ELM-based model. Now we turn our attention in determining the optimal training percentage and the optimum number of neuron.

**RMSE for training and testing.** The OELF being a supervised model the authentication of the training and testing phases are necessary. They play a decisive role in the proposed model. In the present case, OELF is trained to predict the visual imperceptibility metrics (Corr and SSIM) and the fusion1 metric. The RSMEs of the proposed OELF model for the training and testing phase are computed to evaluate its predictability performance. Table 4 summarizes the RSME values of the square window for each of the similarity metrics. The computed RSMEs for all the metrics in both the training phase and testing phase with different images are discerned to be approximately zero, indicating the suitability of the proposed model.

**Developed ELM training.** A number of issues need to be addressed when using ELM. First, an appropriate training–testing ratio has to be determined accurately to avoid over-fitting for using a high training percentage and under-fitting for using a low training percentage. Second, the ELM does not provide the user with the exact number of neurons to be selected for building the network structure. Moreover, the performance of the model depends on the accurate determination neurons number, where a large (small) number of neurons lead to over (under) fitting [38] [2].

The used data set is partitioned into 50% training and 50% testing. Next, the number of neurons is increased from 50 to 200 at a step of 5. In each case, the data set is partitioned into 80% for training and 20% for validation. Validation is performed on a part of the training data set because in the normal functioning mode of the system the testing data set is unavailable. The number of neurons in the hidden layer corresponding to the best validation accuracy is then selected. Once the optimal number of neurons is selected, the search for the best training–testing ratio is performed by assigning a fixed testing data set size. Allocation of fixed

**Table 4. RMSEs for the training phase and testing phase for different images.**

Images	Measure	RMSE (Training phase)	RMSE (Testing phase)
Lena	Corr	0.0000002592	0.0000002604
	MSE	0.000183980	0.0001953800
	SSIM	0.0000060068	0.0000063730
	Fusion1	0.0000059813	0.0000063790
Sails	Corr	0.0000013995	0.0000014011
	MSE	0.000179340	0.0001922900
	SSIM	0.0000097757	0.0000088329
	Fusion1	0.0010000000	0.0011000000
Baboon	Corr	0.0000010623	0.0000010641
	MSE	0.000193010	0.0002089800
	SSIM	0.0000041315	0.0000042289
	Fusion1	0.0000022554	0.0000023760
4.2.01	Corr	0.0000000833	0.0000000874
	MSE	0.000193390	0.0002117400
	SSIM	0.0000891080	0.0000968680
	Fusion1	0.0000891170	0.0000968790

doi:10.1371/journal.pone.0170329.t004

percentage of the data for testing is required to avoid the bias in the RSME with increasing testing data set. Afterward, the percentage of the training data is increased from 10% to 60% for validating each case using the validation part composed of 20% of the training data set. From the total data set, 50% is found to be best for training. Table 5 summarizes the training data set (%) dependent accuracy levels for the Lena, Sails and Baboon images. Figs 8–11 displays the training data set percentages dependent variation in the Corr, MSE, SSIM, and fusion1 values between the host and stego images (Lena, Sails, and Baboon).

**Table 5. Accuracy levels of the different training data set percentages for the Lena, Sails and Baboon images.**

Images	Training (%)	Sample No.	Corr	MSE	SSIM	fusion1
Lena	10	504	0.000000034633	0.00019974	0.0000096058	0.0000096302
	20	1008	0.000000031163	0.00019103	0.0000066814	0.0000066951
	30	1512	0.000000032858	0.00019834	0.0000069203	0.0000069401
	40	2016	0.000000030532	0.00019407	0.0000066675	0.0000066905
	50	2520	0.000000028931	0.00019340	0.0000064897	0.0000065042
	60	3025	0.000000032376	0.00019964	0.0000064997	0.0000065511
Sails	10	504	0.000000061209	0.00020495	0.0000107230	0.0000094358
	20	1008	0.000000060383	0.00019897	0.0000073274	0.0000071024
	30	1512	0.000000058905	0.00019439	0.0000074816	0.0000074046
	40	2016	0.000000055183	0.00017993	0.0000062795	0.0000062419
	50	2520	0.000000054812	0.00018018	0.0000067782	0.0000068851
	60	3025	0.000000056117	0.00018278	0.0000064654	0.0000065537
Baboon	10	504	0.000000064792	0.00020644	0.0000026758	0.0000021155
	20	1008	0.000000061951	0.00020515	0.0000019249	0.0000019248
	30	1512	0.000000061739	0.00020448	0.0000018580	0.0000019298
	40	2016	0.000000060180	0.00020039	0.0000020036	0.0000020270
	50	2520	0.000000059464	0.00019712	0.0000018142	0.0000018343
	60	3025	0.000000059567	0.00019783	0.0000019505	0.0000019876

doi:10.1371/journal.pone.0170329.t005

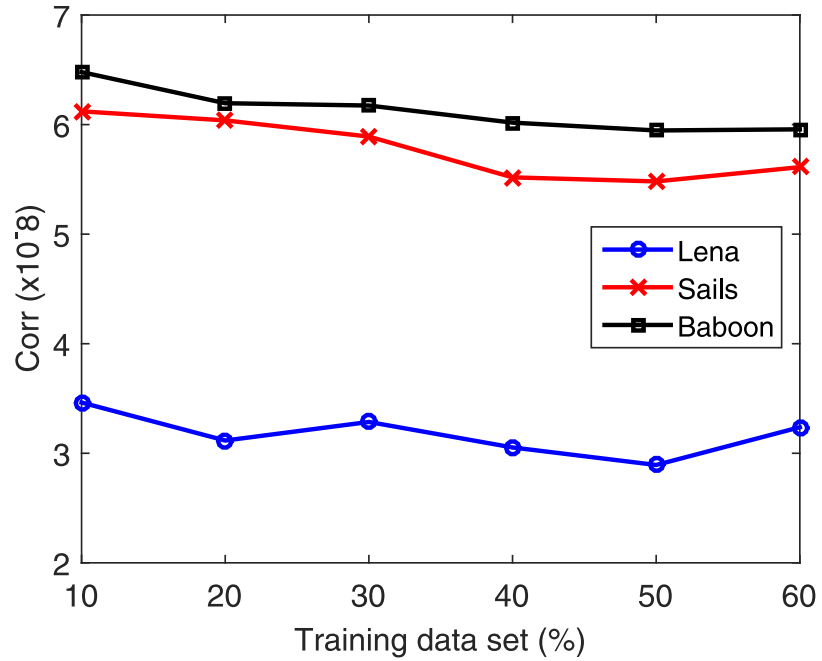


Fig 8. Training data set percentage dependent variation of Corr for the Lena, Sails, and Baboon images.

doi:10.1371/journal.pone.0170329.g008

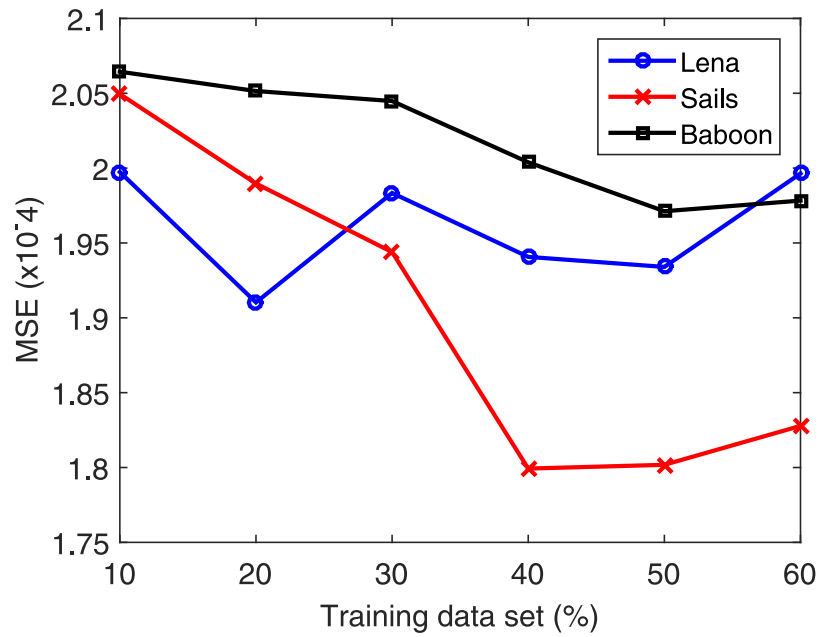


Fig 9. Training data set percentage dependent variation of MSE for the Lena, Sails, and Baboon images.

doi:10.1371/journal.pone.0170329.g009

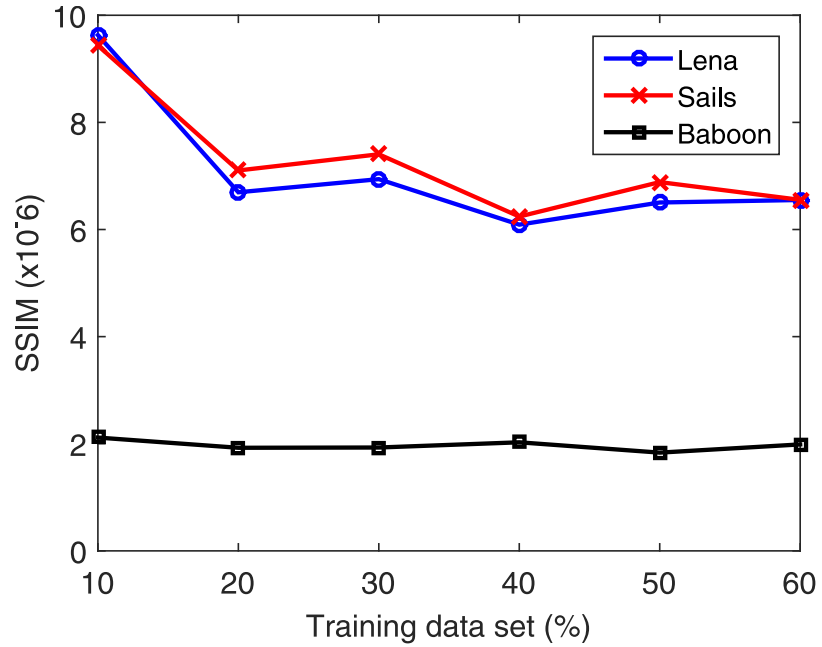


Fig 10. Training data set percentage dependent variation of SSIM for the Lena, Sails, and Baboon images.

doi:10.1371/journal.pone.0170329.g010

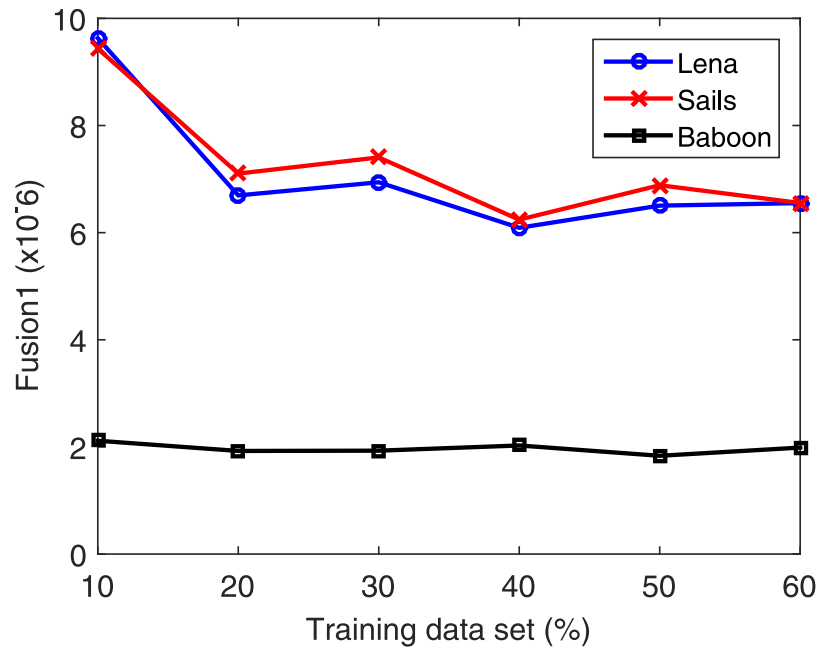
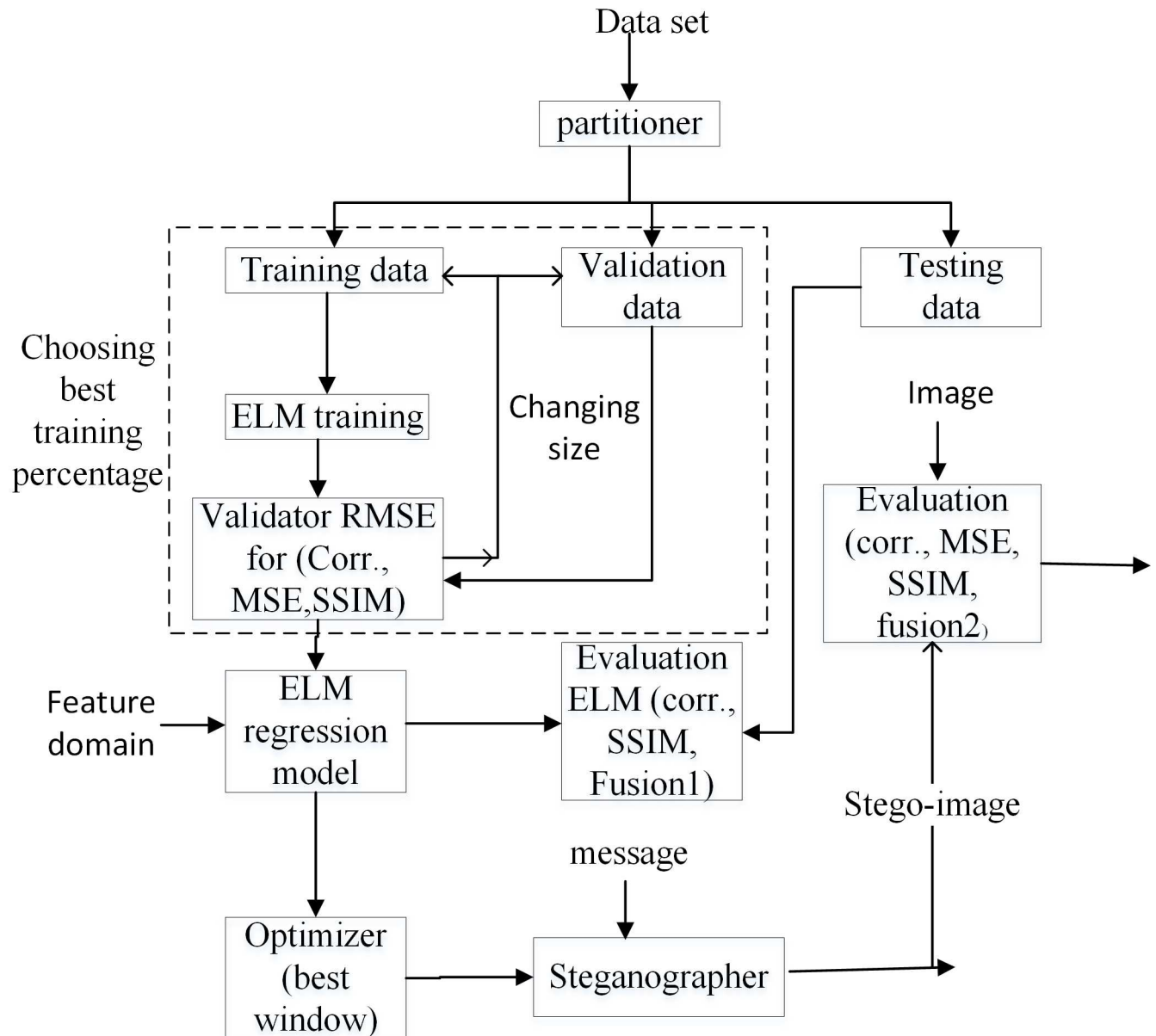


Fig 11. Training data set percentage dependent variation of fusion1 for the Lena, Sails, and Baboon images.

doi:10.1371/journal.pone.0170329.g011



**Fig 12. General framework of the proposed OELF model.**

doi:10.1371/journal.pone.0170329.g012

### Design and optimization of ELM

Fig 12 depicts schematically the framework of the proposed OELF model, which is achieved using the following steps:

1. The data set is partitioned into 50% for training as well as validation and 50% for testing.
2. The ELM regression model is designed based on the training data set (Appendix A) which is partitioned into 80% for training and 20% for validation.
3. The ELM regression model is further used to predict the best square window in terms of the fusion2 metric.



- The embedding process is performed to insert the secret message into the identified optimum square window for generating the stego image.

Using the ELM training the message is embedded into each square window and all visual imperceptibility metrics are determined via fusion2 metric given by:

$$fusion2 = \frac{Corr \times SSIM}{MSE} \tag{15}$$

### Experiments and results

Experiments are conducted on Intel® Core™ i7-2670QM CPU @ 2.20 GHz 6 GB RAM computer with 64-bit operating system. The proposed OELF model is evaluated using 24 gray scale images of size (512 × 512) pixels. Total 5041 square windows are obtained, in which square windows of (232 × 232) are used. The message of size 100 bytes is utilized for embedment. Fig 13 illustrates the tested images before (left panel of each image) and after (right panel of each image) embedding.

Table 6 enlists the RMSEs of the ELM prediction for the visual imperceptibility metrics of the host and stego images for 50% training data set.

The experimental results obtained using the proposed OELF model are compared (Table 7 and Fig 14) with the art-of-the existing methods [24], [39] in terms of the fusion2 metric. OELF approach is found to outperform the other methods [24], [39] in terms of imperceptivity and fusion2 measure which are nearly 28% and 114%, respectively. Thus, OELF is demonstrated to be a useful steganography technique for embedding text in images



Fig 13. Achieved host (left) and stego (right) images.

doi:10.1371/journal.pone.0170329.g013

Table 6. RMSE values obtained using the ELM model for various images.

Image	Corr	MSE	SSIM	fusion2
Lena	0.00000028931	0.00019340	0.0000064897	6.2242
Sails	0.00000054812	0.00018018	0.0000067782	5.9077
Baboon	0.00000059464	0.00019712	0.0000018142	6.3427
4.2.01	0.00000072910	0.00021492	0.0000897420	6.7471
Barbara	0.00000046185	0.00020671	0.0000038981	6.5776
Boat	0.00000046384	0.00018874	0.0000014097	6.1217
Boy	0.00000036993	0.00018259	0.0002314300	5.8925
Bridge	0.00000037995	0.00018965	0.0000006518	6.0393
Cameraman	0.00000034447	0.00019519	0.0000087432	6.3871
Car	0.00000050114	0.00019526	0.0003784400	6.1066
Couple	0.00000074732	0.00017851	0.0000070962	5.7333
Elaine	0.00000048013	0.00018282	0.0000011586	5.9634
Fruits	0.00000091042	0.00018445	0.0000388410	5.9403
Fry mire	0.00000029372	0.00020057	0.0010000000	6.0890
Gold hill	0.00000068636	0.00020280	0.0000048690	6.4077
Lake	0.00000028259	0.00019234	0.0000295420	6.0766
Serrano	0.00000027705	0.00021271	0.0012000000	6.1314
Sport team	0.00000018767	0.00019272	0.0018000000	5.9935
Tulips	0.00000041771	0.00020646	0.0000247010	6.7391
Watch	0.000000170190	0.00021892	0.0012000000	6.0348
Zelda	0.00000076098	0.00019957	0.0000061628	6.4019
Pepper	0.00000042533	0.00019457	0.0000098827	6.2634
F16	0.00000050038	0.00019940	0.0001149300	6.3402
Tiffany	0.000000138370	0.00018762	0.0001004100	5.9857

doi:10.1371/journal.pone.0170329.t006

with minimum level of distortion. Furthermore, it requires only a small training part of the host image features.

### Conclusion

Based on ELM, we proposed a novel OELF model to achieve high-performance image steganography. In this approach, a modified ELM algorithm is used to establish the supervised mathematical model for determining the optimum embedding image location with minimal distortion. The ELM is trained on an image part (or any host medium) and tested in the regression mode to select the best location for embedding the message. This allowed in achieving the best values of the predicted evaluation metrics. The training is performed based on a set of the extracted texture, statistical features, and their corresponding visual imperceptibility metrics using a part of the image. The trained model is further used for the performance optimization. The proposed model is demonstrated to outperform the existing state-of-the-art models. The excellent features of the results suggest that the present model may constitute a basis for the development of secured image steganography algorithm. It is worth to look at the robustness of the proposed method against various statistical attacks by incorporating a wider range of features. Also, it is good to further develop the model to have more degree of freedom in terms of the region finding by defining the region analytically instead of explicit geometrical definition (block region). Other worthy development is to create an index for ranking the solution based on Pareto efficiency.

Table 7. Comparison of the OELF model results with other existing models.

Image	Proposed OELF model				Kanan and Nazeri [24]				Miao Qi et al. [39]			
	MSE	Corr.	SSIM	Fusion2	MSE	Corr.	SSIM	Fusion2	MSE	Corr.	SSIM	Fusion2
Lena	0.001133	0.999999	0.999989	881.8879	0.001384	0.999999	0.999996	722.1569	0.012939	0.999998	0.999910	77.2759
Sails	0.001130	0.999999	0.999996	884.8709	0.001388	0.999999	0.999994	720.1871	0.012329	0.999995	0.999957	81.1051
Baboon	0.001126	0.999999	0.999997	887.8688	0.001372	0.999999	0.999998	728.1764	0.012512	0.999996	0.999994	79.9213
4.2.01	0.001199	0.999999	0.999948	833.4831	0.001266	0.999999	0.999981	789.5758	0.013305	0.999997	0.999937	75.1510
Barbara	0.001205	0.999999	0.999994	829.5649	0.001380	0.999999	0.999995	724.1509	0.011779	0.999997	0.999972	84.8887
Boat	0.001109	0.999999	0.999996	901.4161	0.001407	0.999999	0.999995	710.4141	0.012878	0.999997	0.999929	77.6436
Boy	0.001181	0.999999	0.999971	846.2843	0.001277	0.999999	0.999992	782.5134	0.012756	0.999998	0.999992	78.3917
Bridge	0.001064	0.999999	0.999998	939.5825	0.001399	0.999999	0.999994	714.2845	0.011908	0.999998	0.999988	84.0194
Camera-man	0.001099	0.999999	0.999995	909.6258	0.001361	0.999999	0.999994	734.2930	0.013244	0.999998	0.999928	75.4968
Car	0.001080	0.999999	0.999961	925.5033	0.001194	0.999999	0.999982	837.5063	0.010925	0.999997	0.999973	91.5281
Couple	0.001117	0.999999	0.999995	894.6849	0.001380	0.999999	0.999995	724.1515	0.012207	0.999996	0.999996	81.9194
Elaine	0.001239	0.999999	0.999995	806.5932	0.001377	0.999999	0.999995	726.1570	0.012451	0.999997	0.999947	80.3092
Fruits	0.001182	0.999999	0.999966	845.5976	0.001380	0.999999	0.999987	724.1452	0.014648	0.999997	0.999955	68.2634
Fry-mire	0.001273	0.999999	0.999600	784.9895	0.001296	0.999999	0.999995	771.0079	0.010742	0.999999	0.999995	93.0905
Gold-hill	0.001148	0.999999	0.999995	870.9060	0.001419	0.999999	0.999992	704.6828	0.012390	0.999997	0.999937	80.7041
Lake	0.001176	0.999999	0.999990	860.3891	0.001411	0.999999	0.999990	708.4908	0.012878	0.999998	0.999990	77.6484
Serrano	0.001162	0.999999	0.999817	860.3891	0.001296	0.999999	0.999993	771.0064	0.012512	0.999998	0.999994	79.9214
Sport-team	0.001155	0.999999	0.999542	864.7656	0.001304	0.999999	0.999993	766.4982	0.012023	0.999998	0.999976	83.1654
Tulips	0.001062	0.999999	0.999992	941.2638	0.001384	0.999999	0.999991	722.1538	0.011413	0.999998	0.999968	87.6121
Watch	0.001139	0.999999	0.999730	877.2379	0.001135	0.999999	0.999987	888.6120	0.012390	0.999996	0.999991	80.7084
Zelda	0.001165	0.999999	0.999986	858.0696	0.001388	0.999999	0.999983	720.1634	0.012268	0.999996	0.999910	81.5048
Pepper	0.001156	0.999999	0.999985	864.4361	0.001396	0.999999	0.999990	716.2335	0.012390	0.999997	0.999981	80.7077
F16	0.001115	0.999999	0.999976	896.1971	0.001380	0.999999	0.999990	724.1478	0.011291	0.999997	0.999960	88.5585
Tiffany	0.001131	0.999999	0.999960	884.0927	0.001396	0.999999	0.999985	716.2293	0.134440	0.999996	0.999991	74.3775

doi:10.1371/journal.pone.0170329.t007

### Appendix A

For ELM training, the used data is combined with  $n$  arbitrary distinct square windows  $(x_j, t_j)$  with  $j = 1, \dots, n$ ,  $x_j = (x_{j1}, x_{j2}, \dots, x_{jn})$  denotes the input vector and  $t_j$  denotes the target. It is possible to model the standard Single Hidden Layer Feed Forward Network (SLFN) with an activation function  $g(x)$  and  $\tilde{n}$  hidden layer neurons via:

$$\sum_{i=1}^{\tilde{N}} \beta_i g(w_i x_j + b_i) = t_j \tag{A.1}$$

where  $j = 1, \dots, n$ ,  $w_i = (a_{i1}, a_{i2}, \dots, a_{in})^T$ ,  $b_i$  is the threshold (biases) of the  $i^{\text{th}}$  hidden node, and  $\beta_i$  is the weight connecting the  $i^{\text{th}}$  hidden node and the output.

The above equation is compact form yields:

$$I\beta = T \tag{A.2}$$

where  $I = I(a_1, a_2, \dots, a_{\tilde{n}}, x_1, x_2, \dots, x_N, b_1, b_2, \dots, b_{\tilde{n}})$

$$I = \begin{bmatrix} g(a_1 x_1 + b_1) & \cdots & g(a_{\tilde{n}} x_1 + b_{\tilde{n}}) \\ \vdots & \ddots & \vdots \\ g(a_1 x_n + b_1) & \cdots & g(a_{\tilde{n}} x_n + b_{\tilde{n}}) \end{bmatrix} \tag{A.3}$$

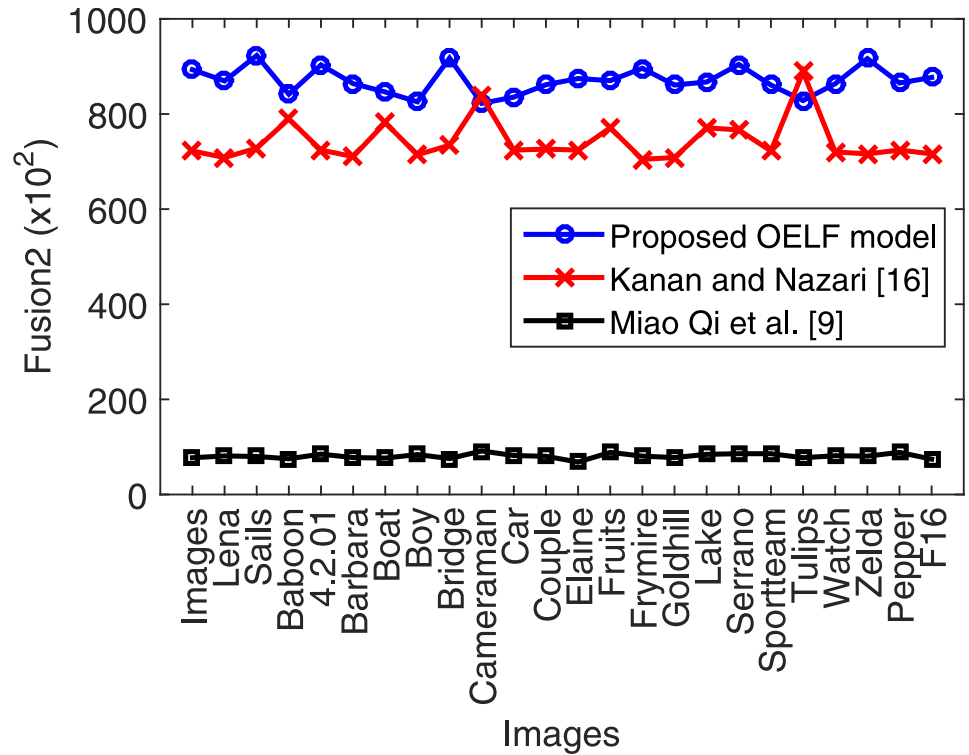


Fig 14. Performance of the proposed imperceptibility metric (*fusion2*).

doi:10.1371/journal.pone.0170329.g014

with

$$\beta = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_{\tilde{n}} \end{pmatrix} \quad T = \begin{pmatrix} t_1 \\ \vdots \\ t_n \end{pmatrix} \tag{A.4}$$

where  $I$  is called the hidden layer output matrix of the neural network and  $T$  is the target vector.

One can prove that if the activation function is differentiable then the required number of the hidden layer neurons is lower than the data size or  $\tilde{n} < n$ . The training of the neural network is achieved via the following steps:

1. Assigning random weights ( $w_i$ ) and biases ( $b_i$ ).
2. Calculating the hidden layer output matrix.
3. Computing the output weights ( $\beta$ ) via:

$$\beta = I^T T \tag{A.5}$$

where  $I^T$  is the Moore-Penrose generalized inverse of hidden layer output matrix.

### Supporting information

**S1 File. Original data for Figs 1–6,8–11 and 14.**  
(XLSX)

## Author Contributions

- Conceptualization:** HAA.
- Data curation:** HAA RA NMN YA.
- Formal analysis:** HAA RA NMN AMSR.
- Funding acquisition:** HAA.
- Investigation:** HAA.
- Methodology:** HAA.
- Project administration:** HAA RA NMN.
- Resources:** HAA RA NMN.
- Software:** HAA YA.
- Supervision:** RA NMN AMSR.
- Validation:** HAA RA NMN AMSR.
- Visualization:** HAA.
- Writing – original draft:** HAA.
- Writing – review & editing:** HAA RA NMN AMSR.

## References

- Iranpour M, Rahmati M. An efficient steganographic framework based on dynamic blocking and genetic algorithm. *Multimed Tools Appl* [Internet]. 2014; 74(24):11429–50. Available from: <http://link.springer.com/10.1007/s11042-014-2237-2>
- Dwd Q, Jeroudi Y Al, Ali MA, Latief M, Akmeliawati R. Online Sequential Extreme Learning Machine Algorithm Based Activity Recognition Using Inertial Data. In: *Control Conference (ASCC), 2015 10th Asian*. Kota Kinabalu: IEEE Conference Publications; 2015.
- Deng Z, Choi KS, Cao L, Wang S. T2fela: Type-2 fuzzy extreme learning algorithm for fast training of interval type-2 TSK fuzzy logic system. *IEEE Trans Neural Networks Learn Syst*. 2014; 25(4):664–76.
- Wang S, Jiang Y, Chung F-L, Qian P. Feedforward kernel neural networks, generalized least learning machine, and its deep learning with application to image classification. *Appl Soft Comput* [Internet]. 2015; 37:125–41. Available from: <http://www.sciencedirect.com/science/article/pii/S1568494615004809>
- Lecun Y, Bengio Y, Hinton G. Deep learning. *Nature*. 2015; 521(1):436–44.
- Atawneh S, Almomani A, Sumari P. Steganography in digital images: Common approaches and tools. *IETE Tech Rev (Institution Electron Telecommun Eng India)*. 2013; 30(March 2015):344–58.
- Ma XY, Lin J-J. HVS-based imperceptibility evaluation for steganography. *Lect Notes Inst Comput Sci Soc Telecommun Eng*. 2009; 18 LNICST(2):152–61.
- Banerjee I, Bhattacharyya S, Sanyal G. Hiding & Analyzing Data in Image Using Extended PMM. In: *International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA) 2013* [Internet]. 2013 [cited 2015 May 20]. p. 157–66. Available from: <http://www.sciencedirect.com/science/article/pii/S2212017313005021>
- Pevný T, Filler T, Bas P. Using high-dimensional image models to perform highly undetectable steganography. *Inf hiding* [Internet]. 2010;161–77. Available from: [http://link.springer.com/chapter/10.1007/978-3-642-16435-4\\_13%5Cnhttp://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=5411758](http://link.springer.com/chapter/10.1007/978-3-642-16435-4_13%5Cnhttp://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5411758)
- Abdelwahab AA, Hassaan LA. A discrete wavelet transform based technique for image data hiding. In: *2008 National Radio Science Conference*. 2008. p. 1–9.
- Cho S, Cha B-H, Gawecki M, Jay Kuo C-C. Block-based image steganalysis: Algorithm and performance evaluation. *J Vis Commun Image Represent* [Internet]. 2013 Oct [cited 2015 Jun 7]; 24(7):846–56. Available from: <http://www.sciencedirect.com/science/article/pii/S1047320313000886>

12. Prabakaran G, Bhavani R. A modified secure digital image steganography based on Discrete Wavelet Transform. In: International Conference on Computing, Electronics and Electrical Technologies [ICCEET] [Internet]. IEEE Conference Publications; 2012. p. 1096–100. [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=6203811](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6203811)
13. Wu HC, Lee CC, Tsai CS, Chu YP, Chen HR. A high capacity reversible data hiding scheme with edge prediction and difference expansion. *J Syst Softw* [Internet]. 2009; 82(12):1966–73. Available from: <http://dx.doi.org/10.1016/j.jss.2009.06.056>
14. Luo W, Huang F, Huang J. Edge Adaptive Image Steganography Based on LSB Matching Revisited. *IEEE Trans Inf Forensics Secur* [Internet]. 2010 Jun; 5(2):201–14. Available from: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5411758>
15. Barni M. Steganography in Digital Media: Principles, Algorithms, and Applications [Book Reviews]. *IEEE Signal Process Mag*. 2011; 28(September):142–4.
16. Ker A, Böhme R. Revisiting weighted stego-image steganalysis. In: ESecurity, Forensics, Steganography, and Watermarking of Multimedia Contents X [Internet]. 2008. <http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=812649>
17. Islam S, Gupta P. Intelligent computing theory. In: Huang D-S, Bevilacqua V, Premaratne Prashan, editors. 10th International conference, ICIC. Taiyuan, China; 2014. p. 803–11.
18. Almohammad A. Steganography-Based Secret and Reliable Communications : Improving Steganographic Capacity and Imperceptibility by. Doctor thesis. Brunel University; 2010.
19. Cheddad A, Condell J, Curran K, Mc Kevitt P. Digital image steganography: Survey and analysis of current methods. *Signal Processing* [Internet]. 2010 Mar [cited 2015 Jan 1]; 90(3):727–52. Available from: <http://www.sciencedirect.com/science/article/pii/S0165168409003648>
20. Paulson LD. New system fights steganography. IEEE, editor. News briefs IEEE Comput Soc [Internet]. 2006; 39(8):25–7. Available from: [http://journals2.scholarsportal.info/details?uri=/00189162/v39i0008/25\\_nsf.xml](http://journals2.scholarsportal.info/details?uri=/00189162/v39i0008/25_nsf.xml)
21. Filler T, Fridrich J. Design of Adaptive Steganographic Schemes for Digital Images. In: Media Watermarking, Security, and Forensics III [Internet]. San Francisco Airport, California, USA; 2011. <http://dx.doi.org/10.1117/12.890762>
22. Filler T, Judas J, Fridrich J. Minimizing additive distortion in steganography using syndrome-trellis codes. *IEEE Trans Inf Forensics Secur*. 2011; 6(3 PART 2):920–35.
23. Pathak P, Selvakumar S. Blind Image Steganalysis of JPEG images using feature extraction through the process of dilation. *Digit Investig* [Internet]. 2014; 11(1):67–77. Available from: <http://www.sciencedirect.com/science/article/pii/S1742287614000024>
24. Kanan HR, Nazeri B. A novel image steganography scheme with high embedding capacity and tunable visual image quality based on a genetic algorithm. *Expert Syst Appl*. 2014; 41(14):6123–6130.
25. El-Emam NN, AL-Zubidy RAS. New steganography algorithm to conceal a large amount of secret message using hybrid adaptive neural networks with modified adaptive genetic algorithm. *J Syst Softw* [Internet]. 2013 Jun [cited 2015 Feb 9]; 86(6):1465–81. Available from: <http://www.sciencedirect.com/science/article/pii/S0164121212003317>
26. Tsai D-S, Horng G, Chen T-H, Huang Y-T. A novel secret image sharing scheme for true-color images with size constraint. *Inf Sci (Ny)* [Internet]. 2009 Sep [cited 2015 Jun 7]; 179(19):3247–54. Available from: <http://www.sciencedirect.com/science/article/pii/S0020025509002412>
27. Husien S, Badi H. Artificial neural network for steganography. *Sprin*, editor. *Neural Comput Appl* [Internet]. 2014 Aug 28 [cited 2014 Nov 30]; 26(1):111–6. Available from: <http://link.springer.com/10.1007/s00521-014-1702-1>
28. Ghaleb Al-Jbara HA, Mat Kiah LB, Jalab HA. Increased capacity of image based steganography using artificial neural network. *AIP Conf Proc*. 2012; 1482:20–5.
29. Najafi HL. A neural network approach to digital data hiding based on the perceptual masking model of the human vision system. *Int J Intell Comput Cybern*. 2010; 3(3):391–409.
30. Elshoura SM, Megherbi DB. A secure high capacity full-gray-scale-level multi-image information hiding and secret image authentication scheme via Tchebichef moments. *Signal Process Image Commun* [Internet]. 2013 May [cited 2015 Jun 7]; 28(5):531–52. Available from: <http://www.sciencedirect.com/science/article/pii/S0923596512002263>
31. El-Alfy E-SM. Detecting Pixel-Value Differencing Steganography Using Levenberg-Marquardt Neural Network. In: Proceedings of the 2013 IEEE Symposium on Computational Intelligence and Data Mining. 2013. p. 160–5.
32. Oplatkova Z, Holoska J, Zelinka I, Senkerik R. Steganography detection by means of neural network. *Int J Innov Comput Appl*. 2013; 5(3):184–90.

33. Pratt M a., Konda S, Chu C-HH. Texture-based image steganalysis by artificial neural networks. *Int J Intell Comput Cybern*. 2008; 1(4):549–62.
34. Zhang F, Pan Z, Cao K, Zheng F, Wu F. The upper and lower bounds of the information-hiding capacity of digital images. *Inf Sci (Ny)* [Internet]. 2008 Jul [cited 2015 Jun 7]; 178(14):2950–9. Available from: <http://www.sciencedirect.com/science/article/pii/S0020025508000984>
35. Roy R, Laha S. Optimization of Stego Image Retaining Secret Information Using Genetic Algorithm with 8-connected PSNR. *Procedia Comput Sci* [Internet]. 2015; 60:468–77. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S1877050915022954>
36. Huang G-B, Zhu Q-Y, Siew C-K. Extreme learning machine: Theory and applications. *Neurocomputing*. 2006; 70(1–3):489–501.
37. Haralick R, Shanmugan K, Dinstein I. Textural features for image classification. *IEEE Trans Syst Man Cybern* [Internet]. 1973; 3(6):610–21. Available from: <http://dceanalysis.bigr.nl/Haralick73-Texturalfeaturesforimageclassification.pdf>
38. Luo J, Vong C-M, Wong P-K. Sparse Bayesian Extreme Learning Machine for Multi-classification. *IEEE Trans NEURAL NETWORKS Learn Syst*. 2014; 25(4):836–43.
39. Qi M, Lu Y, Du N, Zhang Y, Wang C, Kong J. A novel image hiding approach based on correlation analysis for secure multimodal biometrics. *J Netw Comput Appl* [Internet]. 2010 May [cited 2015 Jun 7]; 33(3):247–57. Available from: <http://www.sciencedirect.com/science/article/pii/S1084804509001404>