IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 16, 2022, accepted March 30, 2022, date of publication April 5, 2022, date of current version April 13, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3165031

Quantized Gaussian JPEG Steganography and Pool Steganalysis

MOHAMMED ALORAINI[1], (Member, IEEE), MEHDI SHARIFZADEH[2], (Member, IEEE), AND DAN SCHONFELD[3], (Senior Member, IEEE)

[1]Department of Electrical Engineering, College of Engineering, Qassim University, Unaizah 56452, Saudi Arabia
[2]Department of Information Retrieval Machine Learning Models, Google, Los Angeles, CA 90012, USA
[3]Department of Electrical and Computer Engineering, University of Illinois at Chicago, Chicago, IL 60607, USA

Corresponding author: Mohammed Aloraini (mo.aloraini@qu.edu.sa)

This work was supported by the Deanship of Scientific Research, Qassim University.

ABSTRACT Currently, algorithms for compressed image steganography mainly embed hidden message by minimizing the resulting distortion or statistical detectability. However, as a result of purely heuristic distortion definitions and numerically solvable equations in statistical models, there are no closed-form solutions for JPEG steganography. The absence of closed-form expression to model JPEG steganography is the main limitation on understanding single image and pool steganalysis behavior. In this study, building upon our previously proposed framework for spatial steganography, we develop a statistical framework for JPEG steganography in which the cover and the hidden message are modeled by multivariate Gaussian distribution. Based on this statistical model, we propose a novel quantized Gaussian JPEG steganography that is able to accomplish embedding using any costs defined in spatial or discrete cosine transform (DCT) domain as well as residual variances. We conduct our experiments using a popular database with different compression qualities to determine the effectiveness of the proposed model. The experimental results show that the proposed model improves the security of previous works and outperforms the state-of-the-art JPEG steganography algorithms. Furthermore, we extend the closed-form expression of single image steganalysis error to pool steganalysis for an omniscience optimal detector. We employ the derived expression to approximate the empirical results of pool steganalysis based on the empirical detection error of single image steganalysis. The practical advantage of the approximation is that even though it is derived based on the adopted statistical model, it is accurate regardless of payload, embedding domain, embedding method, and steganalysis feature as long as the pooling strategy is optimal. In addition to approximation of the error, we employ the proposed model to make predictions about the variance behavior of pool steganalysis error. We mathematically show that the variance increases as the pool size increases in small payloads. The same behavior is observed in experimental results which re-validates our analytical model. We conclude that although pooling improves detector's performance, it makes the detector less stable in low payloads and high pool sizes.

INDEX TERMS JPEG steganography, optimal detector, quantized gaussian embedding, pool steganalysis.

I. INTRODUCTION

Steganography is the art of embedding hidden message in a cover medium without getting detected by the warden [1]. The most common medium for steganography is digital image data due to having high redundancy which results in high capacity for embedding. In early works in digital image steganography both in spatial and compressed domains, non-adaptive methods were proposed and they treated all the pixels or DCT coefficients in the same manner. Examples of such methods in spatial domain are [2], [3] and in JPEG steganography are Jsteg [4], F5 [5], and nsF5 [6]. As a result of not taking pixel to pixel or coefficient to coefficient dependencies into consideration, all non-adaptive approaches have low security [6], [7]. Thus, for attaining a higher security performance, adaptive methods have been developed.

Content adaptive steganography methods embed more in textured regions rather than smooth regions of an image to

The associate editor coordinating the review of this manuscript and approving it for publication was Yassine Maleh.

minimize the produced distortion [8], [9]. Distortion minimization embedding is formulated to source coding with a fidelity criterion [10], and it is solved for a general case by syndrome trellis codes [11], [12]. This coding scheme employs a distortion measure or embedding cost for each cover element and executes embedding accordingly, e.g. higher embedding rate in low cost elements. Many methods are available for computing the embedding costs for image steganography for both spatial and JPEG domains. HILL [13] and SUNIWARD [14] are well-known examples for spatial domain steganography. CHAT-GAN [15] is another example that uses generative adversarial networks to embed hidden messages in spatial domain. For JPEG domain steganography, UED [16], UERD [17], IUERD [18], JUNIWARD [14], and GUED [1] are among most frequently used approaches. Even though some of these methods such as HILL, SUNIWARD, and JUNIWARD have the highest security, they are all based on heuristically defined distortions and therefore, there is no theoretical/statistical measure for their performances. This issue has been addressed in another type of image steganography, called statistical or model based.

Statistical or model based image steganography methods mathematically model the cover image and perform embedding while minimizing a distance measure between the cover and the stego image. Examples of such approaches in spatial domain are HUGO [19], MG [20], MVGG [21], and MiPOD [22]. Denemark and Fridrich introduced the only statistical based method in JPEG domain called J-MiPOD based on MiPOD statistical model and also proposed algorithms for steganography with pre-cover for both spatial and JPEG domains (SI-MiPOD and SI-J-MiPOD) [23]. In all of the mentioned statistical based approaches, the optimization problem, defined as minimizing distance between cover and stego images while embedding, results in numerically solvable equations. Thus, there are no closed-form expressions for the embedding probabilities and detection error. Having such an expression, specially for the detection error, would be beneficial in understanding and estimating image steganography behavior as well as batch steganography and pool steganalysis.

In our previous work, we developed a statistical framework for spatial steganography which resulted in closed-form expressions for embedding probabilities and detection error while achieving state-of-the-art empirical performance [24]. In this work, we extend our model to JPEG domain and propose a statistical framework for JPEG steganography which results in closed-form expressions for detection error and embedding probabilities. Our proposed framework is able to employ any embedding costs defined in the spatial or JPEG domain, and also any residual variance estimator for JPEG steganography. In addition, it can be utilized to model single image and pool steganalysis.

Pool steganalysis is the extension of steganalysis problem in which the warden knows multiple objects share the same source and therefore, pools evidence from all of the objects to achieve a higher detection performance. This problem was introduced by Ker and it is the dual of batch steganography problem in which the steganographer embeds a payload in multiple cover objects [25]. Both problems are major research problems in steganography [26]. Previous studies have proposed methods for ranking multiple sources according to their ''guiltiness'' [27], [28]. However, a more general question still remains; which source is guilty? This question was studied under the assumption of an omniscience detector and it was shown that for finding the guilty source, the average pooling strategy performance is close to optimal for a very wide range of hidden message distribution strategies [29]. In another study, the problem of sequential steganalysis is discussed and a method is proposed for finding the first stego message in a sequence of objects [30]. Cogranne *et al.* formulated the problem in spatial domain and demonstrated that knowledge of the steganographer's strategy increases the performance of pool steganalysis [31]. In contrast to these studies, Zakaria *et al.* assumed that steganalyzer does not know the payload spreading strategy and proposed a pooling methods that performs close to an omniscience steganalyzer for all the state-of-the-art payload spreading strategies [32]. In all the mentioned works, there is no statistical analysis for modeling pool steganalysis of steganography with state-of-the-art payload spreading strategies in real images.

In this study, we derive the detection error for single image steganalysis mathematically based on the adopted statistical model. We show that the detection error formula is valid for embedding in spatial domain or any linear transformation domain. This allows us to derive a unified closed-form formulation for the optimal pool steganalysis strategy and it's error for steganography in any domain. Here, we assume steganalyzer is omniscience and payload is spread among all of the images uniformly or using the state-of-the-art batch steganography method [24]. To show the relevance of the results, we employ the derived closed-form expression for pool steganalysis error to approximate the empirical detection error of JPEG steganography and it's variance for various pool sizes. We demonstrate that our proposed approximation is precise considering the error of empirical steganalysis. As a result, one can approximate the pool steganalysis results instead of running time consuming and cumbersome experiments.

In this work, our contribution is threefold:

1) We develop a statistical model for JPEG cover and stego images. Based on that, we extend our previous embedding model for spatial steganography to JPEG steganography and derive the closed-form detection error for such an embedder against an optimal hypothesis detector [24]. The embedding model is generalized in the sense that it is able to utilize any embedding cost or variance estimator defined in spatial domain or JPEG domain, and it results in superior security comparing to the state-of-the-art approaches.
2) We extend the closed-form expression of single image steganalysis detection error to pool steganalysis for an omniscience optimal warden. We employ the derived

expression to approximate empirical results of pool steganalysis computed by an ensemble classifier steganalyzer based on the empirical detection error of single image steganalysis [33]. Although the approximation is derived based on our proposed embedding model, it is precise for all the payloads, embedding domains, embedding methods, and steganalysis features as long as the pooling strategy is optimal. It also holds for single image steganography and batch steganography using the state-of-the-art batching strategy, i.e. *AdaBIM* [24].

3) We approximate the variance of such a pool steganalyzer and show that it increases as the pool size increases in small payloads employing the proposed detection error approximation. Small payloads are more interesting as they are more applicable than high payloads which are easily detectable. Therefore, we conclude that although pooling makes the detector more reliable as it decreases detection error, it makes detector less reliable in the sense that it increases the variance. In other words, pooling makes the steganalyzer less stable. We observed the exact same behavior in empirical results as well which confirms the correctness of the approximation.

This paper is organized as follows. The statistical models for cover and stego message are presented in Sec. II. Based on the proposed Gaussian model, a framework for quantized Gaussian JPEG steganography is introduced in Sec. III. The results are then extended to pool steganalysis in Sec. IV. In Sec. V, we provide the empirical results. Sec. VI summarizes and concludes this work.

II. STATISTICAL MODELS

In this section, we describe the statistical model for cover image in spatial domain and subsequently, we derive probability distribution of DCT coefficients of cover. Also, we derive the statistical model of the stego image in DCT domain by embedding a Gaussian message in each coefficient.

A. COVER MODEL

We show an 8-bit gray-scale image in spatial domain by $P = [P_1, \ldots, P_{n'}]$, where n' is the number of blocks, and P_b is the b^{th} block of 8×8 pixels, $P_b = [p_{bij}]_{8\times 8}$. Note that total number of pixels shown by n is $n = n' \times 64$. All the pixels, p_{bij}, are quatized to $\{0, 1, \ldots, 255\}$. Lets assume $\hat{p}_{bij}$ is an unbiased estimation of the pixel based on its neighbors. We model the estimation errors, defined as $e_{bij} = p_{bij} - \hat{p}_{bij}$, as independent Gaussian random variables, $\mathcal{N}(0, \omega^2_{bij})$. This model is based on the assumption of fine quantization which is given by $\omega_{bij} \gg 1$, since the quantization step is 1. For a detailed explanation of this model, refer to [22]. Suppose the scaled DCT coefficients of the cover image are similarly shown as $F = [F_1, \ldots, F_{n'}]$, where $F_b = [f_{bkl}]_{8\times 8}$ and each coefficient, f_{bkl}, is given by

$$f_{bkl} = \frac{1}{q_{kl}} \sum_{i,j=0}^{7} \mathrm{w}(k, l, i, j) p_{bij} \; \forall k, l \in \{0, 1, \ldots, 7\} \quad (1)$$

where q_{kl} is the kl^{th} element of JPEG quantization matrix and $\mathrm{w}(k, l, i, j)$ is defined as

$$\mathrm{w}(k, l, i, j) = \frac{\mathrm{c}(k)\mathrm{c}(l)}{4} \cos\frac{\pi k(2i+1)}{16} \cos\frac{\pi l(2j+1)}{16} \quad (2)$$

where $\mathrm{c}(x)$ is given by

$$\mathrm{c}(x) = \begin{cases} 1/\sqrt{2} & \text{if } x = 0 \\ 1 & o.w. \end{cases} \quad (3)$$

By using Eq. (1) and the estimation in spatial domain, $\hat{p}_{bij}$, we can estimate the scaled DCT coefficients as well. The estimation, $\hat{f}_{bij}$, is

$$\hat{f}_{bkl} = \frac{1}{q_{kl}} \sum_{i,j=0}^{7} \mathrm{w}(k, l, i, j) \hat{p}_{bij} \quad (4)$$

and the residual of the estimation, $x_{bkl} = f_{bkl} - \hat{f}_{bkl}$, is

$$x_{bkl} = \frac{1}{q_{kl}} \sum_{i,j=0}^{7} \mathrm{w}(k, l, i, j) e_{bij} \quad (5)$$

which is a linear combination of zero mean Gaussian random variables. Therefore, the distribution of scaled DCT coefficient residual is

$$p_{x_{bkl}}(k) = \frac{1}{\sigma_{bkl}\sqrt{2\pi}} \exp\left(\frac{-k^2}{2\sigma^2_{bkl}}\right) \quad (6)$$

where σ_{bkl}, based on all e_{bij} being independent, is given by

$$\sigma^2_{bkl} = \frac{1}{q^2_{kl}} \sum_{i,j=0}^{7} \mathrm{w}^2(k, l, i, j) \omega^2_{bij} \quad (7)$$

where ω^2_{bij} is the residual variance of ij^{th} pixel of the b^{th} block in the raw image. The conclusion of DCT residuals having Gaussian distribution, shown in Eq. (6), is drawn based on the fact that DCT is a linear transformation. Thus, the conclusion is valid for any linear transformation of image. Note that Eq. (6) is the probability distribution of scaled DCT coefficient residual or estimation error not the coefficient's distribution. It is well known in the literature that the probability distribution of the scaled DCT coefficient of an image is Laplacian [34], [35]. The Gaussian distribution of the residuals or in other words noise in the JPEG domain can alternatively be derived based on the previous studies on DCT coefficients of JPEG images. By analysing JPEG errors, it has been shown that the summation of all the quantization, rounding, and truncation errors has a Gaussian distribution [36]. In a later work on uncovering JPEG compression history, Li *et al.* have shown that distribution of the error in JPEG domain depends on the number of compression cycles and quantization matrix elements and it has a Gaussian distribution or a quantized-Gaussian distribution [37].

Now, we prove that given the independence of the estimation errors in spatial domain, the errors are independent in DCT domain as well. Based on Eq. (5), and $\mathrm{E}[e_{bij}e_{bi'j'}] = 0$ for two distinct pixels, the covariance of the errors in the same block is

$$\mathrm{E}[x_{bkl}x_{bk'l'}] = \frac{1}{q_{kl}q_{k'l'}}\sum_{i,j=0}^{7}\mathrm{w}(k,l,i,j)\mathrm{w}(k',l',i,j)\omega_{bij}^2 \quad (8)$$

We can assume that ω_{bij}^2 is constant in each block, which is reasonable as in real image ω_{bij}^2 is highly correlated with the energy of the b^{th} block and it has small variation in each block of 8×8 pixels. At the end of this paragraph, we show that this assumption results in a diagonal covariance matrix which elements are shown in Eq. (8). But in general, the covariance matrix is not necessarily diagonal. It can be diagonalized/whitened using eigen-decomposition because it is a real symmetric matrix. Suppose the eigen-decomposition of error covariance matrix of b^{th} block is $\mathrm{U}_b\Gamma_b\mathrm{U}_b^T$. Then, the hidden message can be computed using the method which is explained in Sec. III based on the whitened error covariance, Γ_b. Then the computed message is multiplied by U_b, quantized and embedded into DCT coefficients. This method is explained thoroughly in Sec. V-C where we show that it results in slightly better performance only in high payloads comparing to skipping the whitening step. It also drastically increases the time complexity which is discussed in Sec. V-D. Note that dependant hidden message elements cannot be embedded in dependent cover elements by syndrome trellis codes in practice because of violating the additive distortion assumption of such coding method, although there has been some studies on using STC for non-additive distortion coding for steganography in special cases such as [38]. As a result, for the rest of this study, we assume that ω_{bij}^2 is constant in each block, unless mentioned otherwise. Therefore, we can move the ω_{bij}^2 out of the summation in Eq. (8). Given that $\sum_{i,j=0}^{7}\mathrm{w}(k,l,i,j)\mathrm{w}(k',l',i,j) \approx 0$ unless $k = k'$ and $l = l'$, the covariance of the errors are

$$\mathrm{E}[x_{bkl}x_{bk'l'}] = \begin{cases} 1 & \text{if } k = k' \text{ and } l = l' \\ 0 & o.w. \end{cases} \quad (9)$$

Thus all x_{bkl} are independent zero mean Gaussian random variables with variances shown in Eq. (7). Note that the residual variances, ω_{bij}^2, can be calculated using any variance estimator such as the ones proposed in [21], [22]. In the following two subsections, we discuss the cases where the cost of embedding in spatial domain and DCT domain is given.

1) EMBEDDING COST IN SPATIAL DOMAIN

For the proposed Gaussian embedding model, any embedding cost in spatial domain, e.g. costs defined in [13], [14], can also be used as a proxy to calculate ω_{bij}^2. As we have shown in our previous work, $\omega_{bij}^2 \approx 1/\eta_{bij}^2$ where η_{bij} is the cost of changing the ij^{th} pixel of the b^{th} block by 1 in the raw image. Therefore, based on Eq. (7), the DCT residual variances are derived as follows in case of having spatial domain embedding costs, i.e. η_{bij}, instead of residual variances, ω_{bij}^2.

$$\sigma_{bkl}^2 = \frac{1}{q_{kl}^2}\sum_{i,j=0}^{7}\mathrm{w}^2(k,l,i,j)\frac{1}{\eta_{bij}^2} \quad (10)$$

2) EMBEDDING COST IN DCT DOMAIN

In case of having the cost of embedding in each DCT coefficient as η_{bij}, which is the cost of changing the scaled ij^{th} DCT coefficient of the b^{th} block, the DCT residual variance is given by

$$\sigma_{bkl}^2 = \frac{1}{\eta_{bij}^2} \quad (11)$$

based on our previous work where we showed the reciprocal of the squared embedding cost can be used as a proxy for calculating residual variance [24]. In Eq. (11), η_{bij} can be computed by any of the methods proposed in [16], [17].

As a result of Equations (7), (10), and (11), the embedding model is universal and it works with embedding costs or residual variances calculated in the spatial domain, or the embedding costs calculated in the DCT domain.

This statistical cover model is violated in practice in smooth or saturated blocks because of assuming unbounded DCT coefficients and $\sigma_{bkl} \gg 1$. However, our proposed method will avoid embedding in those regions anyway which is covered thoroughly in Sec. III.

B. STEGO MODEL

We show hidden message by $M = [M_1, \ldots, M_n]$, where M_b is the b^{th} block of 8×8 message elements, $M_b = [m_{bij}]_{8\times8}$. In contrast to all the previous works in which hidden message elements are modeled as discrete random variables, we model them, m_{bij}, as Gaussian random variables with variances β_{bij} distributed according to

$$p_{m_{bij}}(k) = \frac{1}{\beta_{bij}\sqrt{2\pi}}\exp\left(\frac{-k^2}{2\beta_{bij}^2}\right) \quad (12)$$

The scaled DCT coefficients of the stego image is the summation of the cover coefficients with hidden message elements, i.e. $S = F + M$. Hence, the kl^{th} scaled DCT coefficient residual of the b^{th} block is $y_{bkl} = x_{bkl} + m_{bkl}$, and based on Eq. (6) and Eq. (12), its probability distribution is derived as

$$p_{y_{bkl}}(k) \propto \frac{1}{\sqrt{2\pi(\sigma_{bkl}^2 + \beta_{bkl}^2)}}\exp\left(\frac{-k^2}{2(\sigma_{bkl}^2 + \beta_{bkl}^2)}\right) \quad (13)$$

in which we assume unbounded quantization levels and $\sqrt{\sigma_{bkl}^2 + \beta_{bkl}^2} \gg 1$. In the next section, we find $B = [B_1, \ldots, B_n]$, where B_b is the b^{th} block of 8×8 message elements variances, $B_b = [\beta_{bij}]_{8\times8}$, that maximizes the security for a payload limited sender.

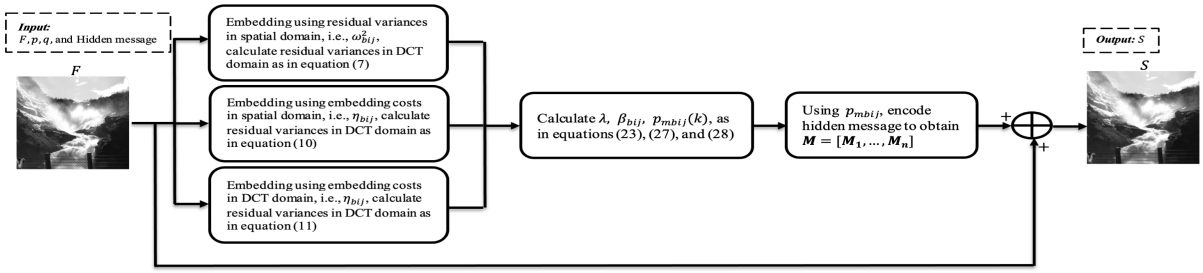

FIGURE 1. A flowchart of the proposed JPEG Gaussian embedding model.

III. METHODOLOGY

In this section, we describe our proposed model as illustrated in Fig. 1. We first discuss the problem of JPEG steganography in a single image which is formulated into the following constrained optimization.

$$\begin{cases} \arg\max\limits_{B} \mathrm{P_E}(B) \\ \sum\limits_{b=1}^{n'} \sum\limits_{i,j=0}^{7} \mathrm{H}(p_{m_{bij}}) = \nu p \end{cases} \tag{14}$$

where ν is the number of non-zero AC DCT coefficients, $\mathrm{P_E}$ is the detection error of the steganalyzer derived in the following section, $\mathrm{H}(p_{m_{bij}})$ is the entropy of a random variable with probability distribution $p_{m_{bij}}$ in natural unit of information (nats) and p is the relative payload in nats per non-zero AC coefficients.

Assume the worst case scenario in which the steganalyzer is omniscience and knows all the cover and hidden message probability distributions, i.e. $p_{x_{bij}}$ and $p_{m_{bij}}$. To compute the detection error of this steganalyzer, i.e. $\mathrm{P_E}(B)$, suppose that it employs a likelihood ratio test (LRT) to decide whether the received image is a cover or it conveys a hidden message, shown by null hypothesis ($\mathcal{H}_0$) and alternative hypothesis ($\mathcal{H}_1$) respectively.

Suppose $R = [R_1, \ldots, R_n]$ are the residuals of received image's DCT coefficients where R_b is the b^{th} block of 8×8 residuals, i.e. $R_b = [r_{bij}]_{8\times 8}$, and they are statistically independent. Therefore, the likelihood ratio for all the DCT coefficients can be simplified as $\prod_{b=1}^{n'} \prod_{i,j=0}^{7} \Lambda_{bij}$ where Λ_{bij} is the likelihood ratio for the ij^{th} residual of b^{th} block. Given Eq. (6) and Eq. (13), Λ_{bij} is

$$\begin{aligned} \Lambda_{bij} &= \frac{p_{y_{bij}}(r_{bij})}{p_{x_{bij}}(r_{bij})} \\ &= \sqrt{\frac{\sigma^2_{bij}}{\sigma^2_{bij}+\beta^2_{bij}}} \exp\left(\frac{-r^2_{bij}}{2} \frac{-\beta^2_{bij}}{\sigma^2_{bij}(\sigma^2_{bij}+\beta^2_{bij})}\right) \end{aligned} \tag{15}$$

Thus the natural logarithm of Λ_{bij} is given by

$$\ln \Lambda_{bij} = \ln \sqrt{\frac{\sigma^2_{bij}}{\sigma^2_{bij}+\beta^2_{bij}}} + \frac{\beta^2_{bij}}{2\sigma^2_{bij}(\sigma^2_{bij}+\beta^2_{bij})} r^2_{bij} \tag{16}$$

In Eq. (16), r_{bij} is Gaussian random variable. Thus, $\ln \Lambda_{bij}$ is a constant term plus a Gamma distributed term with shape (k_{bij}) and scale (θ_{bij}) parameters, i.e. $\Gamma(k_{bij}, \theta_{bij})$. In both cases of $\mathcal{H}_0$ and $\mathcal{H}_1$, the shape parameter is equal to 0.5, i.e. $k_{bij} = 0.5$. However, the scale parameter, θ_{bij}, depends on the variance of r_{bij}, and it is given by

$$\theta_{bij} = \begin{cases} \beta^2_{bij}/(\sigma^2_{bij}+\beta^2_{bij}) & \text{if } \mathcal{H}_0 \text{ is true.} \\ {\beta_{bij}}^2/\sigma^2_{bij} & \text{if } \mathcal{H}_1 \text{ is true.} \end{cases} \tag{17}$$

Based on our previous work [24], for large enough number of DCT coefficients (or pixels), the following approximation for probability distribution of $\sum_{b=1}^{n'} \sum_{i,j=0}^{7} \ln \Lambda_{bij}$ holds.

$$\sum_{b=1}^{n'} \sum_{i,j=0}^{7} \ln(\Lambda_{bij}) \xrightarrow{\mathrm{d}} \begin{cases} \mathcal{N}(\frac{-1}{4}\alpha,\ \frac{1}{2}\alpha) & \text{if } \mathcal{H}_0 \text{ is true} \\ \mathcal{N}(\frac{+1}{4}\alpha,\ \frac{1}{2}\alpha) & \text{if } \mathcal{H}_1 \text{ is true} \end{cases} \tag{18}$$

$$\alpha = \sum_{b=1}^{n'} \sum_{i,j=0}^{7} \left(\frac{\beta^2_{bij}}{\sigma^2_{bij}}\right)^2 \tag{19}$$

Eq. (18) shows that embedding hidden message in scaled DCT coefficients changes variance of detectors output, however the mean stays the same. This behaviour is similar to the one explained by shift hypothesis for embedding in spatial domain [25].

A steganalyzer utilizing a LRT compares the likelihood ratio with a decision threshold to figure out if there is hidden message in an image or not. The natural logarithm of the LRT is given by

$$\sum_{b=1}^{n'} \sum_{i,j=0}^{7} \ln(\Lambda_{bij}) \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \text{decision threshold} \tag{20}$$

It has been previously shown that for the given LRT, using minimax, one of the most common optimal decision criteria, the decision threshold equal to 0 results in the lowest expected risk over all possible priors [24]. As a result, based on Eq. (18) and Eq. (19), the detection error for the optimal detector is given by

$$\hat{\mathrm{P}}_{\mathrm{E}} = \phi\left(\frac{-\alpha/4}{\sqrt{\alpha/2}}\right) = \phi\left(-\sqrt{\frac{\alpha}{8}}\right) \tag{21}$$

where ϕ is the cumulative density function of standard normal distribution. $\hat{\mathrm{P}}_{\mathrm{E}}$ shown in Eq. (21) is monotonically decreasing as α increases. Thus, to achieve a more secure steganography method, we can minimize α instead of maximizing the error of the steganalyzer. The same conclusion can be made employing other common optimal decision rules such

as Bayes and Neyman–Pearson. Consequently, the problem shown in Eq. (14) can be simplified as

$$\begin{cases} \arg\min_{B} \alpha \equiv \arg\min_{B} \sum_{b=1}^{n'} \sum_{i,j=0}^{7} \left(\frac{\beta_{bij}^2}{\sigma_{bij}^2}\right)^2 \\ \sum_{i=1}^{n'} \sum_{i,j=0}^{7} \mathrm{H}(p_{m_{bij}}) = vp \end{cases} \tag{22}$$

The solution of Eq. (22) using Lagrangian multiplier method is given by

$$\beta_{bij}^* = \frac{\sqrt[4]{\lambda(p)}}{\sqrt{2}} \sigma_{bij} \tag{23}$$

where λ is the Lagrangian multiplier determined by the payload constraint in Eq. (22) as a function of the relative payload, p, and it is derived as follows

$$\lambda(p) = \frac{\mathrm{e}^{4p}}{\left(\sqrt[v]{\prod_{b=1}^{n'} \prod_{i,j=0}^{7} \pi \mathrm{e} \sigma_{bij}^2}\right)^2} \tag{24}$$

Therefore

$$\alpha = \sum_{b=1}^{n'} \sum_{i,j=0}^{7} \left(\frac{\beta_{bij}^2}{\sigma_{bij}^2}\right)^2 = 64\, n' \frac{\lambda(p)}{4} = \frac{n\lambda(p)}{4} \tag{25}$$

$$\hat{\mathrm{P}}_{\mathrm{E}} = \phi\left(-\sqrt{\frac{\alpha}{8}}\right) = \phi\left(-\sqrt{\frac{n\lambda(p)}{32}}\right) \tag{26}$$

where $n = n' \times 64$ is the total number of pixels or DCT coefficients. The closed-form expression for detection error of steganalysis shown in Eq. (26) is derived based on the Gaussian distribution of cover elements residuals and hidden message elements. In addition, the Gaussian distribution is drawn from the fact that DCT is a linear transformation. As a result, the closed-form expression for detection error of steganalysis shown in Eq. (26) is valid for embedding using the proposed adopted model in raw image, or any linear transformation of image such as DCT. Based on this generalized error formulation, in the next section, we develop an statistical model for pool steganalysis which is valid for steganography in raw image data or any linear transformation of image data.

Eq. (23) shows that the message variance is proportional to the DCT coefficients residual variance. As a result, we embed more nats by adding a Gaussian with higher variance in noisy coefficients comparing to coefficients with small variance.

Now that the problem is solved in the continuous domain, we translate the problem, shown in Eq. (22), to discrete domain by quantizing hidden message to $\mathcal{Q} = \{-q, \ldots, -1, 0, 1, \ldots, +q\}$, as follows

$$\begin{cases} \beta_{bij}^* = \frac{\sqrt[4]{\lambda(p)}}{\sqrt{2}} \sigma_{bij} \quad \forall b, i, j \\ -\sum_{b=1}^{n'} \sum_{i,j=0}^{7} \sum_{k=-q}^{q} \left(p_{m_{bij}}(k) \ln p_{m_{bij}}(k)\right) = vp \end{cases} \tag{27}$$

$$p_{m_{bij}}(k) = \frac{\phi\left(\frac{k+0.5}{\beta_{bij}}\right) - \phi\left(\frac{k-0.5}{\beta_{bij}}\right)}{\phi\left(\frac{q+0.5}{\beta_{bij}}\right) - \phi\left(\frac{-q-0.5}{\beta_{bij}}\right)} \tag{28}$$

Algorithm 1 Pseudo-Code of the JPEG Gaussian Embedding Model

Input: F = Cover Image Scaled DCT Coefficients, p = Payload, q, Hidden Message
Output: S = Stego Image Scaled DCT Coefficients

if using residual variances in spatial domain, ω_{bij}^2, for embedding **then**
    derive residual variances in DCT domain by Eq. (7).
else if using embedding costs in spatial domain, η_{bij}, for embedding **then**
    derive residual variances in DCT domain by Eq. (10).
else if using embedding costs in DCT domain, η_{bij}, for embedding **then**
    derive residual variances in DCT domain by Eq. (11).
end if
Find λ by solving the system of equations shown in Eq. (27) using Newton–Raphson method.
Calculate all β_{bij} values for all b, i, and j by Eq. (23).
Determine all $p_{m_{bij}}(k)$ values for all b, i, j, and k by Eq. (28).
Encode hidden message according to the determined change rates, $p_{m_{bij}}$, to get $M = [M_1, \ldots, M_n]$.
Compute the stego image scaled DCT coefficients by $S = F + M$.

Eq. (28) is a truncated Gaussian random variable indicating the probability of changing the ij^{th} coefficient of b^{th} block by k. We utilize the Newton—Raphson method to find the Lagrangian multiplier, $\lambda(p)$, which determines all hidden message variances, i.e. β_{bij}, and distributions, i.e. $p_{m_{bij}}$. To be able to take advantage of practical embedding methods such as syndrome-trellis codes (STCs) [12] for real world implementation of the proposed embedding model, the cost of changing each coefficient is required. We show cost of changing the ij^{th} coefficient of b^{th} block by k by $\rho_{bij}(k)$. Assuming symmetric costs, i.e. $\rho_{bij}(k) = \rho_{bij}(-k)$, there are $64 \times n \times q$ variables and equations having Gibbs form given by

$$p_{m_{bij}}(k) = \mathrm{e}^{-\rho_{bij}(k)} / \sum_{d=-q}^{q} \mathrm{e}^{-\rho_{bij}(d)}, \tag{29}$$

$\forall b \in \{1, \ldots, n\}$, $\forall i, j \in \{0, \ldots, 7\}$, $\forall k \in \{1, \ldots, q\}$. Computing these costs, allows us to utilize STCs for the actual embedding when $q = 1$ and multi-layered STCs for $q > 1$ [12]. However in this manuscript, similar to conceptual studies in steganography, we disregard the coding process and change the coefficients according to the change rates shown in Eq. (28). A summary of our proposed method is shown in Algorithm 1.

IV. POOL STEGANALYSIS

In previous section, we have derived the closed-form solution for JPEG steganography against optimal single image staganalysis and its error. In this section, we discuss the case

where the steganalyzer also knows the source of a pool of images. Then, the detection error is derived for an arbitrary sized pool of images, in which the images are all stego or cover. The notation is the same as before except that we show the image number using a superscript in parenthesis, e.g. $\lambda^{(i)}$ is the Lagrangian multiplier for the i^{th} image. In addition, we show the detection error for pool size l by $\hat{\mathrm{P}}_{\mathrm{E}}(l)$ when it is theoretically estimated and by $\mathrm{P}_{\mathrm{E}}(l)$ when it is empirically computed. The following theorem explains how to derive $\hat{\mathrm{P}}_{\mathrm{E}}(l)$ and what would be its error's behavior.

Theorem 1: Statistical Model for Pool Steganalysis Detector's Error and Variance

Suppose that l images are sent from the same source and in the case of being stego images, they carry the same amount of hidden message or embedding has been done using the state-of-the-art batch steganography method [24]. An omniscience optimal detector should examine the images together and decide based on the summation of all the images detection statistics. The error of such optimal detector can be approximated by

$$\hat{P}_E(l) \approx \phi\left(\phi^{-1}\left(\hat{P}_E(1)\right)\sqrt{l}\right) \tag{30}$$

The standard deviation of $\hat{P}_E(l)$, i.e. $\hat{\sigma}_l$, as a function of the standard deviation of $\hat{P}_E(1)$, i.e. $\hat{\sigma}_1$, is given by

$$\hat{\sigma}_l \approx \sqrt{l}\exp\left(-\frac{1}{2}\left(\phi^{-1}\left(\hat{P}_E(1)\right)\right)^2(l-1)\right)\hat{\sigma}_1 \tag{31}$$

which is an increasing function of the pool size (l) until $l = l_0$ and a decreasing function afterwards, where l_0 is written as

$$l_0 = \left(\phi^{-1}\left(\hat{P}_E(1)\right)\right)^{-2} \tag{32}$$

See Appendix A for the proof. Given that Theorem 1 is true for steganography in raw image data or any linear transformation of image data, its true for JPEG steganography as well. The beauty of this approximation is that utilizing it, one can run only one experiment employing an ensemble classifier steganalyzer [33] to find $\hat{\mathrm{P}}_{\mathrm{E}}(1)$, and plug the result in Eq. (30) to find $\hat{\mathrm{P}}_{\mathrm{E}}(l)$ for any l. In Sec. V-E, we show that although this approximation is based on the Gaussian embedding model and optimal pool steganalysis, it works for any embedding method, as long as the same steganalyzer is employed for all the pool sizes using the explained optimal pooling strategy.

Another conclusion that can be drawn form Theorem 1 is that although pool steganalysis gives better results comparing to single image steganalysis, it suffers from an increasing variance as the pool size increases for some payloads. To the best of authors knowledge this phenomenon has never been discussed nor been formulized in the literature. The variance increases until pool size reaches l_0, shown in Eq. 32 and Fig. 6, and it decreases afterwards. In Sec. V-E, we observe that this statistical model and its results are aligned with the empirical results.

V. EXPERIMENTS AND DISCUSSION

Throughout this paper, we use the BOSSbase 1.01 database containing $10k$ gray-scale 512×512 pixels images [39]. All the images are compressed to JPEG with two quality factors, 75 and 95. Performance evaluations are done using an ensemble of classifiers with 10-fold cross validation trained on steganalysis features extracted from $5k$ images chosen randomly as training/validation set and tested on features extracted from the rest $5k$ images [33]. We utilize two different state-of-the-art JPEG steganalysis feature vectors DCTR [40], and GFR [41] with 8000 and 17000 elements respectively. Performances are reported by the classifier average detection error defined as the mean of false alarm and missed detection rates in payloads ranging from 0.05 to 1, i.e. $p \in \{0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.75, 1\}$, bits per non zero AC coefficient (bpnzac).

To find out if a performance improvement is statistically significant, we employ the significance level of 0.05. For all the performance evaluations in this article, sample sizes are 10, and the standard deviations of samples are in the range of 0.001 to 0.005. In the worst case scenario of comparing two performances both having standard deviation of 0.005, if the difference between them is greater than 0.0047, it is statistically significant.

For all the experiments, we employ six different JPEG steganography methods. The first three are the three state-of-the-art JPEG steganography methods, i.e. UERD [17], JUNIWARD [14], and GUED [1], with their optimal parameters for achieving best security. The next two methods are based on the mentioned methods, UERD and JUNIWARD, but utilizing our proposed quantized Gaussian embedding, we show them by G-UERD and G-JUNIWARD respectively. In addition to these five methods, we also experiment G-JHILL which employs the proposed quantized Gaussian embedding model using spatial domain embedding cost computed by HILL algorithm as shown in Sec II-A1. HILL algorithm is used with a 3×3 Ker-Bohme high-pass filter and a 3×3 and a 15×15 averaging low-pass filters [13].

A. DETERMINING MAXIMUM DCT COEFFICIENT CHANGE (Q)

The parameter q of the proposed quantized Gaussian embedding model summarized in Algorithm 1 controls the maximum amount that DCT coefficients will be changed during embedding. In other words, our embedding model is a $(2q+1)$-ary embedding. To determine optimal q value for achieving highest security, we evaluate all the JPEG steganography methods with different q values, i.e. $q \in \{1, 2, 3\}$. The results are presented in Table 1. It can be seen that for our proposed Gaussian embedding model, reported in the top three sections of the table, i.e. G-UERD G-JUNI G-JHILL, higher q values results in higher performance, however the improvement is not significant for lower payloads. The security is significantly improved only for JPEG quality factor of 95 and in higher payloads ($p \geq 0.5$) which are less

TABLE 1. Detection error of steganalysis using GFR features for various payloads (p) and various embedding algorithms with different q values resulting in a $(2q+1)$-*ary* embedding scenario.

	JPEG Quality Factor = 75									JPEG Quality Factor = 95								
Algorithm	q	p = .05	0.1	0.2	0.3	0.4	0.5	0.75	1	q	p = .05	0.1	0.2	0.3	0.4	0.5	0.75	1
G-UERD	1	0.4600	0.4037	0.2837	0.1814	0.1065	0.0603	0.0133	0.0048	1	0.4876	0.4663	0.4127	0.3483	0.2802	0.2130	0.0797	0.0218
	2	0.4612	0.4074	0.2838	0.1789	0.1075	0.0637	0.0135	0.0043	2	0.4877	0.4648	0.4111	0.3483	0.2793	0.2141	0.0903	0.0328
	3	0.4581	0.4082	0.2840	0.1803	0.1082	0.0606	0.0136	0.0046	3	0.4864	0.4654	0.4126	0.3474	0.2807	0.2135	0.0967	0.0362
G-JUNI	1	0.4637	0.4085	0.2870	0.1885	0.1081	0.0596	0.0115	0.0034	1	0.4914	0.4767	0.4335	0.3782	0.3141	0.2446	0.0990	0.0292
	2	0.4614	0.4062	0.2880	0.1826	0.1094	0.0606	0.0125	0.0033	2	0.4925	0.4757	0.4341	0.3764	0.3149	0.2545	0.1190	0.0457
	3	0.4595	0.4063	0.2895	0.1831	0.1097	0.0615	0.0131	0.0044	3	0.4938	0.4747	0.4354	0.3758	0.3161	0.2567	0.1242	0.0520
G-JHILL	1	0.4650	0.4134	0.2986	0.1893	0.1139	0.0631	0.0131	0.0048	1	0.4943	0.4794	0.4437	0.3945	0.3354	0.2727	0.1336	0.0439
	2	0.4640	0.4138	0.2968	0.1908	0.1160	0.0689	0.0156	0.0047	2	0.4939	0.4805	0.4436	0.3943	0.3384	0.2775	0.1519	0.0705
	3	0.4637	0.4141	0.2971	0.1896	0.1174	0.0686	0.0163	0.0061	3	0.4944	0.4802	0.4435	0.3941	0.3395	0.2798	0.1554	0.0787
UERD	1	0.4560	0.3942	0.2729	0.1874	0.1179	0.0665	0.0169	0.0064	1	0.4857	0.4655	0.4121	0.3466	0.2788	0.2114	0.0845	0.0216
	2	0.4491	0.3807	0.2464	0.1629	0.0974	0.0547	0.0117	0.0044	2	0.4855	0.4654	0.4083	0.3384	0.2701	0.2053	0.0846	0.0293
	3	0.4480	0.3785	0.2422	0.1579	0.0913	0.0510	0.0109	0.0037	3	0.4883	0.4605	0.4011	0.3279	0.2593	0.1920	0.0782	0.0305
JUNI	1	0.4623	0.4056	0.2813	0.1852	0.1052	0.0582	0.0108	0.0018	1	0.4948	0.4796	0.4324	0.3749	0.3089	0.2357	0.0852	0.0153
	2	0.4602	0.4000	0.2700	0.1799	0.1029	0.0555	0.0104	0.0027	2	0.4951	0.4796	0.4304	0.3754	0.3031	0.2335	0.0949	0.0313
	3	0.4585	0.3987	0.2645	0.1753	0.0991	0.0500	0.0093	0.0023	3	0.4925	0.4796	0.4316	0.3740	0.2978	0.2271	0.0925	0.0341

TABLE 2. Detection error of steganalysis using GFR features for various payloads (p) and various embedding algorithms.

	JPEG Quality Factor = 75								JPEG Quality Factor = 95							
Algorithm	p = .05	0.1	0.2	0.3	0.4	0.5	0.75	1	p = .05	0.1	0.2	0.3	0.4	0.5	0.75	1
UERD	0.4560	0.3942	0.2729	0.1874	0.1179	0.0665	0.0169	0.0064	0.4880	0.4655	0.4121	0.3466	0.2788	0.2114	0.0845	0.0216
G-UERD	0.4600	0.4037	0.2837	0.1814	0.1065	0.0603	0.0133	0.0048	0.4876	0.4663	0.4127	0.3483	0.2802	0.2130	0.0797	0.0218
JUNI	0.4623	0.4056	0.2813	0.1852	0.1052	0.0582	0.0108	0.0018	0.4948	0.4796	0.4324	0.3749	0.3089	0.2357	0.0852	0.0153
G-JUNI	0.4637	0.4085	0.2870	0.1885	0.1081	0.0596	0.0115	0.0034	0.4914	0.4767	0.4335	0.3782	0.3141	0.2446	0.0990	0.0292
G-JHILL	0.4650	0.4134	0.2986	0.1893	0.1139	0.0631	0.0131	0.0048	0.4943	0.4794	0.4437	0.3945	0.3354	0.2727	0.1336	0.0439
GUED	0.4630	0.4063	0.2841	0.1869	0.1065	0.0589	0.0112	0.0025	0.4932	0.4759	0.4329	0.3764	0.3108	0.2389	0.0873	0.0193

TABLE 3. Detection error of steganalysis using DCTR features for various payloads (p) and various embedding algorithms.

	JPEG Quality Factor = 75								JPEG Quality Factor = 95							
Algorithm	p = .05	0.1	0.2	0.3	0.4	0.5	0.75	1	p = .05	0.1	0.2	0.3	0.4	0.5	0.75	1
UERD	0.4698	0.4211	0.3257	0.2417	0.1654	0.1039	0.0240	0.0056	0.4958	0.4852	0.4509	0.4001	0.3313	0.2615	0.0981	0.0228
G-UERD	0.4750	0.4350	0.3379	0.2422	0.1614	0.0982	0.0274	0.0062	0.4948	0.4869	0.4497	0.4022	0.3400	0.2706	0.1084	0.0301
JUNI	0.4801	0.4494	0.3560	0.2570	0.1715	0.1040	0.0187	0.0023	0.4960	0.4866	0.4613	0.4158	0.3602	0.2923	0.1030	0.0128
G-JUNI	0.4814	0.4543	0.3637	0.2647	0.1780	0.1076	0.0196	0.0035	0.4954	0.4891	0.4625	0.4216	0.3722	0.3103	0.1307	0.0335
G-JHILL	0.4819	0.4549	0.3646	0.2678	0.1810	0.1114	0.0191	0.0042	0.4982	0.4892	0.4610	0.4259	0.3731	0.3167	0.1444	0.0409
GUED	0.4810	0.4521	0.3586	0.2593	0.1762	0.1053	0.0192	0.0028	0.4969	0.4884	0.4618	0.4192	0.3654	0.3017	0.1128	0.0294

important comparing to lower payloads due to high detection probability. Note that using higher q values results in a more complex encoding algorithm [12]. As a result, for the rest of the experiments, we only consider $q = 1$ which has similar performance comparing to $q = 2$ and $q = 3$ for most of the payloads and requires a less complex encoder.

We have also shown the results of different $(2q + 1)$-ary embedding scenarios for non-Gaussian embedding algorithms, UERD and JUNIWARD, in the bottom two sections of the Table 1. It can be concluded that higher q values results in lower security for almost all the payloads.

These observations suggest that the proposed quantized Gaussian embedding model is more accurate comparing to the widely used Gibbs form [11] for calculating embedding probabilities.

B. COMPARISON OF QUANTIZED GAUSSIAN EMBEDDING WITH PRIOR ARTS

In this section, we compare the security of the proposed steganography method with the state-of-the-art JPEG steganography methods against steganalysis using DCTR and GFR features. We conclude that using the proposed embedding model results in performance improvement for all the algorithms in most of the payloads. We also show that the proposed G-JHILL method outperforms all the previously developed methods in all the payloads.

We compare the detection error of UERD, G-UERD, JUNIWARD, G-JUNIWARD, G-JHILL, and GUED using GFR features in Table 2. For UERD algorithm, the proposed Gaussian version (G-UERD) outperforms UERD significantly in payloads less than 0.3 bpnzac for images with JPEG quality 75 and its detection probabilities at these payloads are similar to the one for JUNIWARD which is a more time consuming algorithm. For JPEG quality of 95, G-UERD has statistically similar performance comparing to UERD. For JUNIWARD, the proposed Gaussian version (G-JUNIWARD) performs better than or similar to the original JUNIWARD algorithm, and the improvement is statistically significant for JPEG quality of 95 and payload greater than 0.3 bpnzac. The GUED performs better than JUNIWARD, but its performance is less than the proposed Gaussian version (G-JUNIWARD) in both JPEG qualities. The proposed G-JHILL outperforms all the mentioned algorithms in all the payloads and JPEG quality factors (or performs similarly to the most secure one). For images with JPEG quality factor of 75, the gap between the performance of G-JHILL and the most secure algorithm amongst the other methods (G-JUNIWARD for $p \leq 0.3$ and UERD for $p > 0.3$)

TABLE 4. Detection error of steganalysis using GFR features in various payloads (p), and different JPEG quality factors (Q.F.) for G-JHILL with and without whitening (Wh.).

Q.F.	Wh.	p=.05	0.1	0.2	0.3	0.4	0.5	0.75	1
75	No	.4650	.4134	.2986	.1893	.1139	.0631	.0131	.0048
	Yes	.4639	.4145	.2991	.1906	.1156	.0670	.0152	.0081
95	No	.4943	.4794	.4437	.3945	.3354	.2727	.1336	.0439
	Yes	.4949	.4786	.4443	.3918	.3384	.2749	.1460	.0642

TABLE 5. Average computational time in seconds for embedding a coded hidden message with size of p bpnzac in a JPEG image with quality factor Q.F.

Q.F.	p	UERD	G-UERD	JUNI	G-JUNI	G-JHIL	G-JHILL (Wh.)	GUED
75	0.1	.2978	1.141	2.326	4.863	4.373	12.03	1.932
	0.2	.2705	1.626	2.669	5.086	4.877	12.32	2.310
95	0.1	.2587	1.344	2.451	5.003	4.598	12.16	1.984
	0.2	.3054	1.472	2.631	5.200	4.866	12.25	2.426

is statistically significant at 0.1 and 0.2 bpnzac. For images with JPEG quality factor of 95, the gap is significant at 0.2, 0.3, 0.4, 0.5, 0.75, and 1 bpnzac.

In addition to running experiments using GFR features, we utilize DCTR features as well and the results are reported in Table 3. Similar behaviors as the ones seen using GFR can be seen there, however the performance gaps are greater comparing to Table 2.

We also provide examples of stego images for the six embedding algorithms as shown in Fig. 2. These images are generated using payload equals 0.3 bpnzac and JPEG quality factor equals 75.

We believe that the proposed quantized Gaussian embedding model improves performance due to the fact that it embeds more bits in low cost or high variance DCT coefficients and less bits in high cost or low variance ones comparing to the Gibbs measure used by all the spatial and JPEG steganography methods.

C. WHITENING

In this section, we conduct experiments on G-JHILL algorithm to check the empirical results of applying whitening explained in Sec. II-A. For applying whitening to all the blocks, there are two extra steps that are added to the algorithm explained in Algorithm 1. First, instead of the residual variances computed in "if" clause in lines 1 through 7, we use variances of the whitened residuals using the eigen-decomposition. In other words, in each block, we first decompose each block residual covariance matrix by eigen-decomposition to $\mathrm{U}_b\Gamma_b\mathrm{U}_b^T$, where U_b is the orthogonal 64×64 matrix of eigenvectors and Γ_b is the diagonal matrix of eigenvalues. Then the diagonal elements of Γ_b are used instead of residual variances, i.e. σ^2_{bij}. The second extra step is that in each Newton-Raphson iteration for solving Eq. (27) after computing $B^*_b = [\beta^*_{bij}]_{8\times 8}$, the hidden message elements are transformed back by $\mathrm{U}_b \cdot \mathrm{vec}(B^*_b)$ where vec is vectorization function. This process increases the time complexity of the embedding method, but it increases the performance. In Table 4, the performances of G-JHILL algorithm is reported for both cases of using and not using whitening. It can be seen that there is no statistically significant change in the detection error for payloads up to 0.5 bpnzac. However, in 0.75 an 1 bpnzac, the G-JHILL version that employs whitening performs significantly better. In the next section, we discuss the amount of increase in computation time for using whitening.

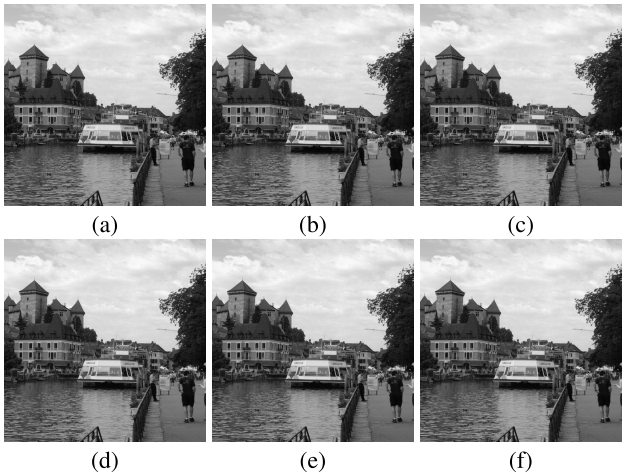

FIGURE 2. Examples of stego images using the six embedding algorithms with payload equals 0.3 bpnzac and JPEG quality factor equals 75. (a) A stego image using UERD. (b) A stego image using G-UERD. (c) A stego image using JUNI. (d) A stego image using G-JUNI. (e) A stego image using G-JHILL. (f) A stego image using GUED.

D. COMPUTATIONAL TIME

In this section, we compare the computation time needed for all of the steganography algorithms studied in this paper. The computation times are reported in seconds per image in Table 5 for two JPEG quality factors, i.e. 75 and 95, and two payloads, i.e. 0.1 and 0.2 bpnzac. It is observed that the proposed Gaussian embedding versions of UERD and JUNIWARD are 2 to 5 times slower than the original algorithms, which is still reasonable given their higher performance. The computation time of GUED is slightly less than the computation time of JUNIWARD. G-JHILL (Wh.) is the G-JHILL algorithm with whitening which is 2 to 3 times slower than G-JHILL. It can be seen that higher payload increases the embedding time but the JPEG quality factor does not affect the computation time significantly.

E. POOL STEGANALYSIS DETECTION ERROR

In this section, we conduct experiments regarding Sec. IV and Theorem 1, where we have shown that instead of running cumbersome pool steganalysis experiments, one can estimate the detection error for pool sizes greater than 1 based on Eq. 30 and empirically computed detection error for pool size equal to 1. We use various pool sizes, i.e. $l \in \{1, 3, \ldots, 99\}$, for both empirical and estimated results. The pooling strategy here is using the summation of detection statistics of all images in a pool. This strategy is shown to be optimal in Theorem 1 in case of embedding the same payload in all images or using the state-of-the-art batch steganographer [24].

Results for using G-UERD embedding algorithm are shown in Fig. 3. In each plot, the pink lines are the empirical

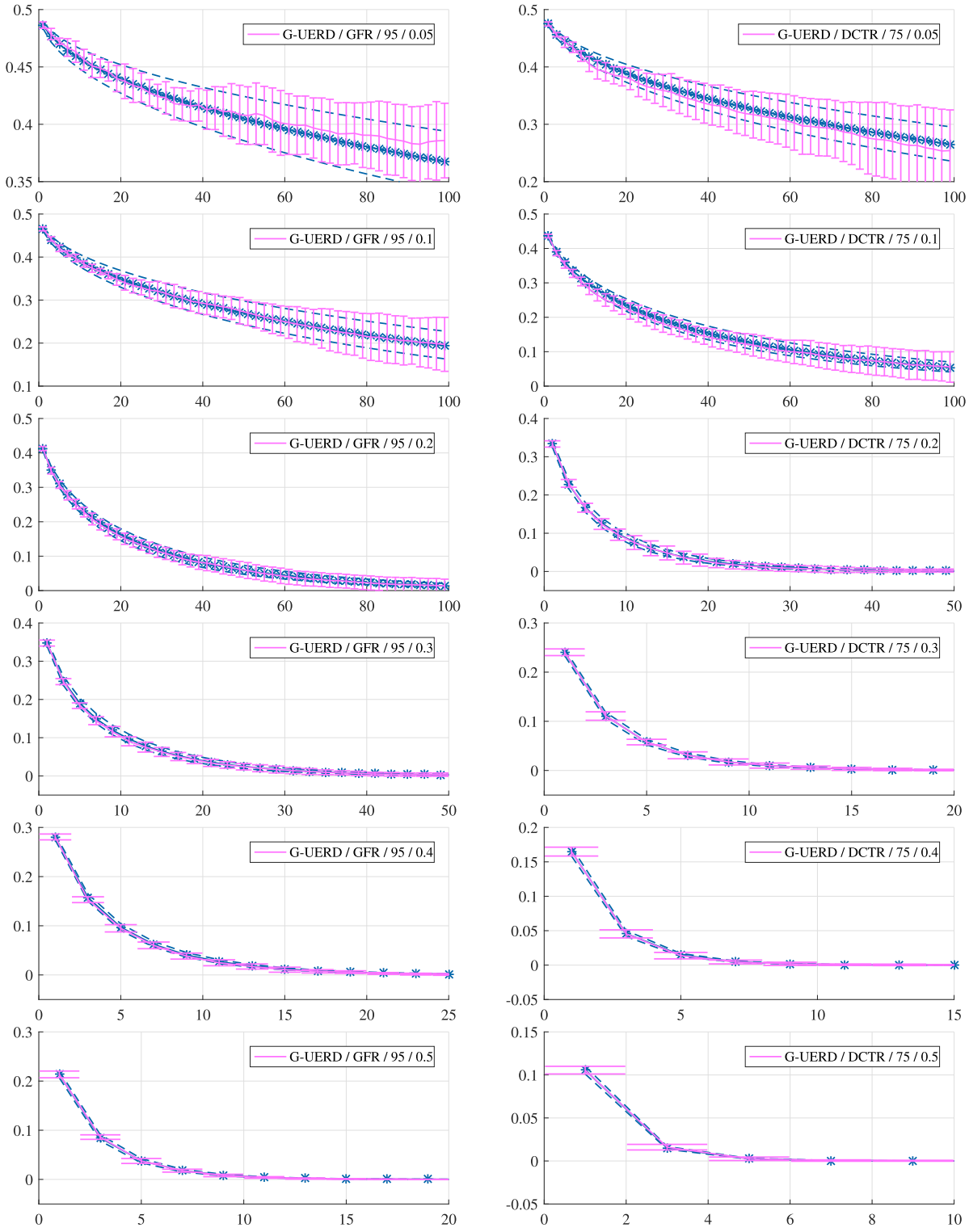

FIGURE 3. Empirical pool steganalysis detection error (pink lines), and the estimated one and its standard deviation calculated by Eq. 30 and Eq. 31 respectively (solid blue lines with "∗" markers and dashed blue lines respectively), versus pool size for G-UERD algorithm, two steganalysis features (GFR and DCTR), different JPEG quality factors (75 and 95) and payloads (0.05, 0.1, 0.2, 0.3, 0.4, 0.5). Plot legends are read as "Embedding method / Steganalysis feature / JPEG quality / Payload."

results and their error bars show the detectors error standard deviation. The solid blue lines with "∗" markers are the results computed by the proposed estimation. The results for JPEG quality factor of 95, using GFR feature, and payloads of 0.05, 0.1, 0.2, 0.3, 0.4, 0.5 bpnzac are provided in the left column in which it can be seen that our estimation is precise.

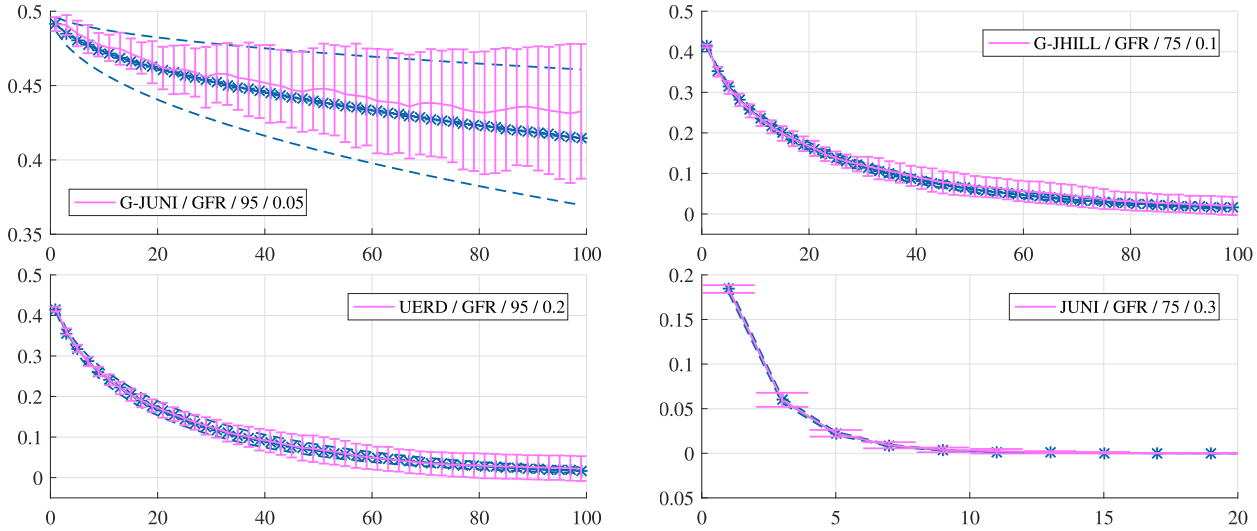

FIGURE 4. Empirical pool steganalysis detection error (pink lines), and the estimated one and its standard deviation calculated by Eq. 30 and Eq. 31 respectively (solid blue lines with "$*$" markers and dashed blue lines respectively), versus pool size for various algorithm, GFR as steganalysis feature, different JPEG quality factors (75 and 95) and payloads (0.05, 0.1, 0.2, 0.3). Plot legends are read as "Embedding method / Steganalysis feature / JPEG quality / Payload."

To show that the proposed estimation is precise for other quality factors and other steganalysis features as well, we show similar plots for JPEG quality factor of 75, using DCTR feature, and payloads of 0.05, 0.1, 0.2, 0.3, 0.4, 0.5 bpnzac on the right column. Based on Fig. 3, the proposed estimation is valid in all the payloads, JPEG quality factors, and steganalysis features for G-UERD algorithm. To show that it is valid for all embedding methods regardless of them using the proposed Gaussian embedding model or not, we provide similar plot for G-JUNIWARD, G-JHILL, UERD, and JUNIWARD in Fig. 4. In this Figure, we have tried to cover all experimented embedding algorithms, JPEG quality factors with different payloads by the fewest possible number of plots due to space and computation limitations.

F. POOL STEGANALYSIS DETECTION ERROR VARIANCE

In this section, we discuss the behavior of the variance of the pool steganalysis detector. In Sec. IV and Theorem 1, we have shown that although pooling improves detection error, it increase the variance of the detector for some payloads depending on the value of single image steganalysis detection error. In other words, according to Theorem 1, the variance of the detection error is an increasing function of pool size for pool sizes smaller than l_0, defined in Eq. (32), and it is a decreasing function for greater pool sizes.

To examine this finding, in all the plots in Fig. 3 and Fig. 4, in addition to the empirical and estimated pool steganalysis results shown by pink error bars and solid blue lines with "$*$" markers respectively, we show the estimated standard deviation shown in Eq. (31) with dashed blue lines. In other words, in all the mentioned plots, the upper and the lower dashed blue lines are $\hat{\mathrm{P}}_{\mathrm{E}}(l) + \hat{\sigma}_l$ and $\hat{\mathrm{P}}_{\mathrm{E}}(l) - \hat{\sigma}_l$ respectively. It can be observed that the pink error bar sizes have similar behaviors as the distances between dashed blue lines. In other words, as the pool size increases, when dashed blue lines are diverging the error bars sizes increase and when dashed blue lines are converging the error bars sizes decrease. The turning point of the explained behavior depends on the value of $\mathrm{P}_{\mathrm{E}}(1)$ and it decreases as $\mathrm{P}_{\mathrm{E}}(1)$ for empirical results. This is similar to the behavior of the estimated turning point l_0 shown in Eq. (32) which validates Theorem 1.

Here, we go through a few examples from the plots. In the left column of Fig. 3, in the top plot where $l_0 \approx 1035$, it can be observed that the size of the error bars of the pink line is increasing until $l = 99$. For the second plot from the top, where $l_0 \approx 139$, the error bars expand as well until $l = 99$. In contrast to the last two examples, in the third plot from the top, where $l_0 \approx 20.5$, the error bars are becoming larger until around $l = 41$ and then they start growing smaller in size. Similarly for the fourth plot from the top where $l_0 \approx 6.6$, the size of the pink error bar is increasing as l increases until around $l = 9$ where it starts to decrease. For the second plot from the bottom in the left column of Fig. 3, where $l_0 \approx 2.9$, error bars start to shrink after approximately $l = 5$. And for the last plot where $l_0 \approx 1.5$, the error bar size is a decreasing function of l.

As a result of the mentioned behavior which we also mathematically proved in Theorem 1, pool steganalysis suffers from instability, i.e. high variance, for small payloads when single image steganalysis detection error is near 0.5. The instability is a serious disadvantage for pool steganalysis specially in low payloads and high pool sizes as the standard deviation can grow form a small number such as 0.004 in pool size equal to 1 to a huge number such as 0.04 in pool size equal to 99.

VI. CONCLUSION

In this study, we extend our previously proposed statistical framework to JPEG steganography in which we employ a Gaussian model for the cover coefficients and also the hidden

message elements. Based on that, we propose a quantized Gaussian embedding model that is able to work with any embedding cost or residual variance computed in spatial or DCT domain. We show that using this embedding model improves the performance of the existing JPEG steganography algorithms in most of the payloads, and also achieves superior performance for all the payloads using cost calculated by HILL. Subsequently, the proposed statistical model allows us to derive the closed-form expression of an optimal omniscience single image steganalyzer error and extended it to pool steganalysis. We use the closed-form expression of pool steganalysis error to accurately approximate the empirical results for pool steganalysis. The main benefit of this approximation is that it is accurate if the pooling method is optimal regardless of payload, steganalysis feature, and embedding method and domain. In addition to approximating the error, we correctly predict the error variance empirical behaviour with respect to pool size, and therefore, reveal a deficiency of pool steganalysis.

As a part of the future work, we plan to investigate side-informed steganography as an immediate extension of this study. In addition, the derived closed-form expressions can be used for calculation of embedding costs and residual variances. Another future path could be using the proposed statistical model for video steganography if frame to frame dependencies are taken into account in computation of the residual variances.

APPENDIX A STATISTICAL MODEL FOR POOL STEGANALYSIS DETECTOR'S ERROR AND VARIANCE

In this section, we discuss the pool steganalysis problem for steganography in raw image or any linear transformation of image. The discussion is based on the Gaussian statistical model which is valid for any linear transformation of image. The model for spatial domain steganography is presented in [42] and for JPEG steganography is shown in Sec. II-A and II-B. Within the adopted statistical model, the detection error of an optimal single image steganalysis is given by

$$\phi\Big(-\sqrt{\frac{n}{32}\lambda(p)}\Big) \tag{33}$$

where $\lambda(p)$ is the Lagrangian multiplier for relative payload p. Now, we discuss the case in which the detector knows that l images are sent by the same source. We prove that in such cases, an optimal pool steganalyzer should examine the images together. To show this, we compare the detection error for both cases of inspecting l images together and separately. Inspecting images together results in a similar detection error with summation of Lagrangian multipliers for all of the l images because the logarithm of the likelihood ratio is equal to summation of logarithm of likelihood ratios for l images. Given that the steganographer is embedding in each image separately, the Lagrangian multiplier values are different for every image, i.e. $\lambda^{(a)}(p)$ is the Lagrangian multiplier for the a^{th} image. The detection error for such a detector is as follows

$$\phi\Big(-\sqrt{\frac{n}{32}\sum_{a=1}^{l}\lambda^{(a)}(p)}\Big) \tag{34}$$

This shows that the optimal detector developed here uses pooling strategy of summing detection statistics of all the images in the pool.

If l images, known to have the same source, are inspected separately, the average detection error is given by

$$\frac{1}{l}\sum_{a=1}^{l}\phi\Big(-\sqrt{\frac{n\lambda^{(a)}(p)}{32}}\Big) \tag{35}$$

Eq. (35) is greater or equal than the formula below based on Jensen's inequality and the fact that $\phi(-\sqrt{x})$ is a convex function of x if $x > 0$.

$$\phi\Big(-\sqrt{\frac{n}{32\,l}\sum_{a=1}^{l}\lambda^{(a)}(p)}\Big) \tag{36}$$

Eq. (36) is greater than detection error shown in Eq. (34) based on the fact that $\phi(-\sqrt{x})$ is a decreasing function of x if $x > 0$. This proves that steganalyzer should inspect all the images from the same source together to achieve a lower detection error. However, this approach will result in a detector with higher variance which is covered later in this section. Now that we have derived the optimal pool steganalysis strategy and its detection error, we show that instead of running time consuming pool steganalysis experiments, one can utilizes Eq. (34) to approximate the results.

Assume that a database of N images (JPEG or raw) is used for embedding a relative payload of p nats (p nats per non zero AC DCT coefficients for JPEG images and p nats per pixel for raw images) using the proposed Gaussian embedding model. The average detection error of an optimal single image steganalyzer for the whole database is given by

$$\hat{\mathrm{P}}_{\mathrm{E}}(1) = \frac{1}{N}\sum_{a=1}^{N}\phi\Big(-\sqrt{\frac{n}{32}\lambda^{(a)}(p)}\Big) \tag{37}$$

which can be approximated as shown below by assuming that all $\lambda^{(a)}(p)$ values are the same and equal to a value $\lambda(p)$

$$\hat{\mathrm{P}}_{\mathrm{E}}(1) \approx \phi\Big(-\sqrt{\frac{n}{32}\lambda(p)}\Big) \tag{38}$$

The mentioned assumption is true for the state-of-the-art batch steganography method which embeds in each image according to its steganographic capacity and uses an image merging sender which results in equal values of λ [24], [31]. The assumption is an approximation for a steganographer that embeds the same payload in all the images but it still results in a precise estimation as shown in Sec. V-E.

If the images are received in pools of l images, the detection error of an optimal pool steganalyzer is given by

$$\hat{\mathrm{P}}_{\mathrm{E}}(l) = \frac{l}{N}\sum_{t=0}^{N/l-1}\phi\Big(-\sqrt{\frac{n}{32}\sum_{a=t\times l+1}^{(t+1)\times l}\lambda^{(a)}(p)}\Big) \tag{39}$$

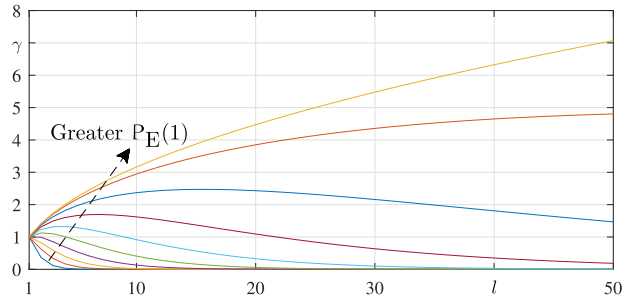

FIGURE 5. Pool steganalysis error variance behaviour shown by plotting variable γ, defined in Eq. (45), versus pool size, l, for different detection errors of single image steganalysis, $\hat{P}_E(1)$.

which can also be approximated as shown below using the same assumption of equal Lagrangian multipliers

$$\hat{P}_E(l) \approx \phi\Big(-\sqrt{\frac{n}{32} l\lambda(p)}\Big) \tag{40}$$

Therefore, based on Eq. (38) and Eq. (40), an approximation of $\hat{P}_E(l)$ based on the value of $\hat{P}_E(1)$ is given by

$$\hat{P}_E(l) \approx \phi\Big(\phi^{-1}\big(\hat{P}_E(1)\big)\sqrt{l}\Big) \tag{41}$$

where ϕ^{-1} is the inverse function of cumulative standard normal distribution, ϕ.

In the rest of this section, we discuss the error of this approximation if $\hat{P}_E(1)$ has an error with standard deviation of $\hat{\sigma}_1$. We show the standard deviation of error of $\hat{P}_E(l)$ with $\hat{\sigma}_l$. Suppose that all the errors are small, i.e. $\forall l\ \hat{\sigma}_l \ll 1$. Therefore, our approximation shown in Eq. (41) has error with standard deviation, i.e. $\hat{\sigma}_l$, given by

$$2\hat{\sigma}_l \approx \phi\Big(\phi^{-1}\big(\hat{P}_E(1)+\sigma_1\big)\sqrt{l}\Big) - \phi\Big(\phi^{-1}\big(\hat{P}_E(1)-\sigma_1\big)\sqrt{l}\Big) \tag{42}$$

This can be further simplified using the following Taylor series expansion

$$\phi\big(\phi^{-1}(x\pm\delta x)\sqrt{l}\big) \approx \phi\big(\phi^{-1}(x)\sqrt{l}\big) \pm \frac{\partial\phi\big(\phi^{-1}(x)\sqrt{l}\big)}{\partial x}\delta x \tag{43}$$

By plugging in this Taylor series in Eq. (42), our approximation error can be calculated as

$$\hat{\sigma}_l \approx \frac{\partial\phi\big(\phi^{-1}(x)\sqrt{l}\big)}{\partial x}\bigg|_{x=\hat{P}_E(1)} \hat{\sigma}_1 = \gamma\hat{\sigma}_1 \tag{44}$$

$$\gamma \doteq \sqrt{l}\exp\left(-\frac{1}{2}\Big(\phi^{-1}\big(\hat{P}_E(1)\big)\Big)^2(l-1)\right) \tag{45}$$

The variable γ's behavior with respect to l depends on $\hat{P}_E(1)$ value. In Fig. 5, γ is shown for different l and $\hat{P}_E(1)$, which shows that for a all $\hat{P}_E(1)$, $\gamma = 1$ when $l = 1$ and it has one global maximum. It can be seen that utilizing pool steganalysis results in greater variances for some pool sizes comparing to single image steganalysis for higher $\hat{P}_E(1)$, because γ is greater than 1. To find out exactly when this happens, we derive the derivation of γ with respect to l which is given by

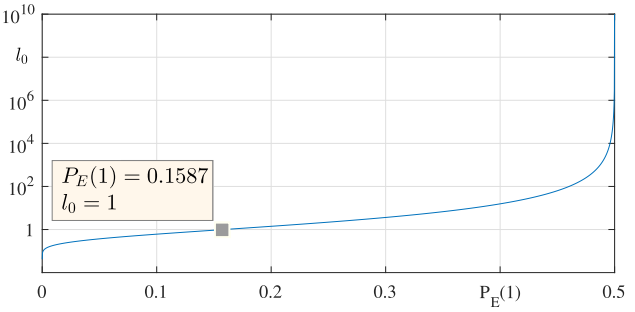

FIGURE 6. l_0 defined in Eq. (47) versus $\hat{P}_E(1)$.

$$\frac{\partial\gamma}{\partial l} = \frac{\gamma\cdot\left(1-\Big(\phi^{-1}\big(\hat{P}_E(1)\big)\Big)^2 l\right)}{2l} \tag{46}$$

Since l takes only natural numbers in practice, γ is a decreasing function of l if its derivation shown in Eq. (46) goes to zero for $l \le 1$. The derivation of γ with respect to pool size, l, is zero if $l = l_0$ where l_0 is

$$l_0 = \Big(\phi^{-1}\big(\hat{P}_E(1)\big)\Big)^{-2} \tag{47}$$

$$l_0 \le 1 \Rightarrow \hat{P}_E(1) \le 0.1587 \tag{48}$$

Fig. 6 depicts l_0 vs $\hat{P}_E(1)$. Therefore, for any $\hat{P}_E(1) > 0.1587$, our approximation show that the variance, $\hat{\sigma}_l$, increases as l increases until $l = l_0$. Then, the variance of detection error decreases. The same behaviour is also observed in practice in Sec. V-F for empirical detection error which reassures the precision of the proposed approximation and mathematical model for pool steganalysis.

ACKNOWLEDGMENT

The researchers would like to thank the Deanship of Scientific Research, Qassim University for funding the publication of this project.

REFERENCES

[1] W. Su, J. Ni, X. Li, and Y.-Q. Shi, "A new distortion function design for jpeg steganography using the generalized uniform embedding strategy," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 12, pp. 3545–3549, Dec. 2018.

[2] A. Cheddad, J. Condell, K. Curran, and P. M. Kevitt, "Digital image steganography: Survey and analysis of current methods," *Signal Process.*, vol. 90, no. 3, pp. 727–752, Mar. 2010.

[3] N. F. Johnson and S. Jajodia, "Exploring steganography: Seeing the unseen," *Computer*, vol. 31, no. 2, pp. 26–34, Feb. 1998.

[4] D. Upham. (1993). *Steganographic Algorithm Jsteg*. [Online]. Available: http://zooid. org/˜ paul/crypto/jsteg

[5] A. Westfeld, "F5—A steganographic algorithm," in *Proc. Int. Workshop Inf. Hiding*. Pittsburgh, PA, USA: Springer, 2001, pp. 289–302.

[6] J. Fridrich, T. Pevnỳ, and J. Kodovskỳ, "Statistically undetectable jpeg steganography: Dead ends challenges, and opportunities," in *Proc. 9th workshop Multimedia Secur.*, 2007, pp. 3–14.

[7] J. Fridrich, M. Goljan, and R. Du, "Reliable detection of LSB steganography in color and grayscale images," in *Proc. Workshop Multimedia Secur., Challenges*, 2001, pp. 27–30.

[8] X. Liao, J. Yin, M. Chen, and Z. Qin, "Adaptive payload distribution in multiple images steganography based on image texture features," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 2, pp. 897–911, Apr. 2020.

[9] X. Liao, Y. Yu, B. Li, Z. Li, and Z. Qin, "A new payload partition strategy in color image steganography," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 3, pp. 685–696, Jan. 2019.

[10] C. E. Shannon, "Coding theorems for a discrete source with a fidelity criterion," *IRE Nat. Conv. Rec*, vol. 4, nos. 142–163, p. 1, 1959.
[11] T. Filler and J. Fridrich, "Gibbs construction in steganography," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 4, pp. 705–720, Dec. 2010.
[12] T. Filler, J. Judas, and J. Fridrich, "Minimizing additive distortion in steganography using syndrome-trellis codes," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 920–935, Sep. 2011.
[13] B. Li, M. Wang, J. Huang, and X. Li, "A new cost function for spatial image steganography," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 4206–4210.
[14] V. Holub, J. Fridrich, and T. Denemark, "Universal distortion function for steganography in an arbitrary domain," *EURASIP J. Inf. Secur.*, vol. 2014, no. 1, pp. 1–13, 2014.
[15] J. Tan, X. Liao, J. Liu, Y. Cao, and H. Jiang, "Channel attention image steganography with generative adversarial networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 2, pp. 888–903, Mar. 2022.
[16] L. Guo, J. Ni, and Y. Q. Shi, "Uniform embedding for efficient JPEG steganography," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 5, pp. 814–825, May 2014.
[17] L. Guo, J. Ni, W. Su, C. Tang, and Y.-Q. Shi, "Using statistical image model for JPEG steganography: Uniform embedding revisited," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 12, pp. 2669–2680, Dec. 2015.
[18] Y. Pan, J. Ni, and W. Su, "Improved uniform embedding for efficient JPEG steganography," in *Proc. Int. Conf. Cloud Comput. Secur.* Nanjing, China: Springer, 2016, pp. 125–133.
[19] T. Pevnỳ, T. Filler, and P. Bas, "Using high-dimensional image models to perform highly undetectable steganography," in *Proc. Int. Workshop Inf. Hiding*. Berlin, Germany: Springer, 2010, pp. 161–177.
[20] J. J. Fridrich and J. Kodovskỳ, "Multivariate Gaussian model for designing additive distortion for steganography," in *Proc. ICASSP*, May 2013, pp. 2949–2953.
[21] V. Sedighi, J. Fridrich, and R. Cogranne, "Content-adaptive pentary steganography using the multivariate generalized Gaussian cover model," *Proc. SPIE*, vol. 9409, Mar. 2015, Art. no. 94090H.
[22] V. Sedighi, R. Cogranne, and J. Fridrich, "Content-adaptive steganography by minimizing statistical detectability," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 2, pp. 221–234, Feb. 2016.
[23] T. Denemark and J. Fridrich, "Model based steganography with precover," *Electron. Imag.*, vol. 2017, no. 7, pp. 56–66, 2017.
[24] M. Sharifzadeh, M. Aloraini, and D. Schonfeld, "Adaptive batch size image merging steganography and quantized Gaussian image steganography," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 867–879, 2019.
[25] A. D. Ker, "Batch steganography and pooled steganalysis," in *Information Hiding*, vol. 4437. Berlin, Germany: Springer, 2006, pp. 265–281.
[26] A. D. Ker, P. Bas, R. Böhme, R. Cogranne, S. Craver, T. Filler, J. Fridrich, and T. Pevnỳ, "Moving steganography and steganalysis from the laboratory into the real world," in *Proc. 1st ACM Workshop Inf. Hiding Multimedia Secur.*, 2013, pp. 45–58.
[27] A. D. Ker and T. Pevnỳ, "A new paradigm for steganalysis via clustering," *Proc. SPIE*, vol. 7880, Feb. 2011, Art. no. 78800U01.
[28] A. D. Ker and T. Pevny, "Batch steganography in the real world," in *Proc. Multimedia Secur.*, 2012, pp. 1–10.
[29] T. Pevny and I. Nikolaev, "Optimizing pooling function for pooled steganalysis," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Nov. 2015, pp. 1–6.
[30] R. Cogranne, "A sequential method for online steganalysis," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Nov. 2015, pp. 1–6.
[31] R. Cogranne, V. Sedighi, and J. Fridrich, "Practical strategies for content-adaptive batch steganography and pooled steganalysis," in *Proc. Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 2122–2126.
[32] A. Zakaria, M. Chaumont, and G. Subsol, "Pooled steganalysis in JPEG: How to deal with the spreading strategy?" 2019, *arXiv:1906.11525*.
[33] J. Kodovskỳ, J. Fridrich, and V. Holub, "Ensemble classifiers for steganalysis of digital media," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 432–444, Apr. 2012.
[34] R. L. Joshi and T. R. Fischer, "Comparison of generalized Gaussian and Laplacian modeling in DCT image coding," *IEEE Signal Process. Lett.*, vol. 2, no. 5, pp. 81–82, May 1995.
[35] E. Y. Lam, "A mathematical analysis of the DCT coefficient distributions for images," *IEEE Trans. Image Process.*, vol. 9, no. 10, pp. 1661–1666, Oct. 2000.
[36] W. Luo, J. Huang, and G. Qiu, "JPEG error analysis and its applications to digital image forensics," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 480–491, Sep. 2010.
[37] B. Li, T. T. Ng, X. Li, S. Tan, and J. Huang, "Statistical model of JPEG noises and its application in quantization step estimation," *IEEE Trans. Image Process.*, vol. 24, no. 5, pp. 1471–1484, May 2015.
[38] W. Zhang, Z. Zhang, L. Zhang, H. Li, and N. Yu, "Decomposing joint distortion for adaptive steganography," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 10, pp. 2274–2280, Oct. 2017.
[39] P. Bas, T. Filler, and T. Pevnỳ, "'Break our steganographic system': The ins and outs of organizing boss," in *Proc. Int. Workshop Inf. Hiding*. Berlin, Germany: Springer, 2011, pp. 59–70.
[40] V. Holub and J. Fridrich, "Low-complexity features for JPEG steganalysis using undecimated DCT," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 2, pp. 219–228, Feb. 2015.
[41] X. Song, F. Liu, C. Yang, X. Luo, and Y. Zhang, "Steganalysis of adaptive JPEG steganography using 2D Gabor filters," in *Proc. 3rd ACM Workshop Inf. Hiding Multimedia Secur.*, 2015, pp. 15–23.
[42] M. Sharifzadeh, M. Aloraini, and D. Schonfeld, "Quantized Gaussian embedding steganography," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 2637–2641.

MOHAMMED ALORAINI (Member, IEEE) received the B.S. degree in electrical engineering from Qassim University, in 2011, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Illinois at Chicago, in 2014 and 2020, respectively. In 2020, he joined Qassim University, where he is currently an Assistant Professor with the Department of Electrical Engineering. His current research interests include image and video analysis, computer vision, multimedia forensics, and information security.

MEHDI SHARIFZADEH (Member, IEEE) received the B.S. degree in electrical engineering from the Sharif University of Technology, in 2012, and the M.S. and Ph.D. degrees in electrical and computer engineering with a specialization in image steganography from the University of Illinois at Chicago, in 2018 and 2019, respectively. He is currently working with Google as a Software Engineer on information retrieval machine learning models. His research interests include image steganography, machine learning, neural networks, and computer vision.

DAN SCHONFELD (Senior Member, IEEE) received the B.S. degree in electrical engineering and computer science from the University of California at Berkeley, in 1986, and the M.S. and Ph.D. degrees in electrical and computer engineering from Johns Hopkins University, in 1988 and 1990, respectively. In 1990, he joined the University of Illinois at Chicago, where he is currently a Professor with the Departments of Electrical and Computer Engineering, Computer Science, and Bio-Engineering. He has authored over 200 technical papers in various journals and conferences. His current research interests include signal processing, image and video analysis, video retrieval and communications, multimedia systems, computer vision, medical imaging, and genomic signal processing. He has been elevated to the rank of fellow of IEEE and SPIE. He has been elected as the University Scholar of the University of Illinois and received the Graduate Mentoring Award of the University of Illinois at Chicago. He has previously served as the Editor-in-Chief for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and an Area Editor for Special Issues of the *IEEE Signal Processing Magazine*.