Minimizing the Embedding Impact in Steganography

Jessica Fridrich
SUNY Binghamton
Department of ECE
Binghamton, NY 13902-6000
001 607 777 6177
fridrich@binghamton.edu

ABSTRACT

In this paper, we study the trade-off in steganography between the number of embedding changes and their amplitude. We assume that each element of the cover image is assigned a scalar value that measures the impact of making an embedding change at that pixel (e.g., the embedding distortion). Given the embedding impact profile of all pixels, we derive an analytic formula for the optimal number of pixels that should be used in combination with syndrome coding to minimize the overall embedding impact. We interpret the results and formulate several "rules of thumb" that should improve steganographic security. Contrary to what has been recommended in the literature before, our analysis implies that it is never optimal to only use the pixels with the smallest embedding impact. We also study *q*-ary embedding in the spatial domain and conclude that the smallest embedding impact is achieved for ternary schemes. This confirms some empirically derived facts previously published elsewhere.

Categories and Subject Descriptors

E.4 Coding and Information Theory, I.4 Image processing and computer vision

General Terms

Algorithms, Security, Theory

Keywords

Steganography, steganalysis, matrix embedding, distortion, perturbed quantization, syndrome coding

1. MOTIVATION

The primary goal of steganography is to build a statistically undetectable communication channel (the famous Prisoner Problem [1–3]). In order to embed a secret message, the sender slightly modifies the cover object and obtains the embedded stego object. In steganography under the passive warden scenario, the goal is to communicate as many bits as possible without introducing any detectable artifacts into the cover object. Attempts to give a formal definition of steganographic security can be found in [4–7]. In practice, a steganographic scheme is considered secure if no existing attack can distinguish between cover and stego images with a success better than random guessing.

Conference ACM Multimedia Security'06, Sep 26–27, 2006, Geneva, Switzerland.

Although the results of this study are applicable to steganography in general, we limit our discussions to digital images. The security of a steganographic scheme is a function of its attributes, which are (1) the *cover image source* whose properties are known to the attacker (Kerckhoffs' principle), (2) the *embedding operation* that is applied to pixels to embed a message, and (3) the *selection channel*, which is a rule according to which pixels are selected for embedding. By imposing an upper bound on the maximal number of allowed embedding changes, we obtain an upper bound on the maximal number of bits one can communicate. To minimize the impact of embedding, we should intuitively only use those pixels whose modifications will introduce the least detectable artifacts. On the other hand, allowing the embedding scheme to use more pixels gives us the possibility to apply syndrome coding (matrix embedding [9, 12, 13]) and decrease the number of embedding changes. The questions are "what is the optimal strategy the sender should choose? How should he balance the number of embedding changes and their amplitude?" We now explain these issues on the example of Perturbed Quantization steganography (PQ) [8].

In PQ, the sender uses side information about the cover image, such as its high-resolution form, to determine the selection channel. For example, the sender may embed data into a JPEG file while utilizing his knowledge of the unquantized DCT coefficients and constrain the embedding changes to those DCT coefficients that experience the largest quantization error – the coefficients that are closest to the middle of the quantization intervals. Such coefficients, when rounded to the other value, leave the smallest embedding distortion. The concept of PQ is very general and can be applied whenever the sender processes the cover image before embedding (e.g., using resizing, decreasing color bit depth, filtering, A/D conversion, etc.). PQ must be combined with codes for memory with defective cells (Wet Paper Codes (WPC) [9]) to enable the recipient to extract the message.

In PQ, the sender, however, has more options. In order to embed m bits, he can either select m pixels with the smallest embedding distortion or select the best k pixels for embedding, $k > m$, and apply syndrome coding for relative payload m/k. The optimum choice of k obviously depends on the embedding distortion profile. For example, if the embedding distortion was the same for

all pixels in the image, the best strategy is to use all pixels in the image because this will allow us to minimize the total number of embedding changes using syndrome coding. On the other hand, one intuitively feels that if the embedding distortion sharply increases as we add more pixels than m, the best strategy might be to embed the message only in the best m pixels and not use syndrome coding at all.

The recent work by Kim et al. [10] addresses the issues above for the case of JPEG steganography. The authors modify the F5 steganographic algorithm [11] and allow more than one change in matrix embedding [12–13] realized using binary Hamming codes. By allowing more than one change in each block of DCT coefficients, the sender is presented with multiple possibilities and can thus select the one that introduces the smallest distortion. The authors use the knowledge of the raw, uncompressed image and minimize the combined quantization and embedding error. In this sense, this method is similar to PQ. While this approach can significantly decrease the embedding distortion when compared to F5, no effort is made to compare the proposed scheme with respect to the smallest achievable embedding distortion.

The subject of this paper is to investigate the optimum strategy the sender can choose to minimize the overall embedding impact. Our goal is to establish the theoretical bounds assuming the sender uses the best possible strategy. Towards this goal, we allow the sender to use a more general measure of the embedding impact that does not have to necessarily be the embedding distortion. Assuming that the sender uses syndrome coding with the best possible performance, we can analytically derive the optimum strategy for a given embedding distortion profile.

In Section 2, we define the notation and concepts used in this paper and review some basic facts concerning syndrome coding. The optimum embedding strategy is derived in Section 3, where we analyze it and draw some interesting conclusions. In Section 4, we apply the same framework to q-ary embedding in the spatial domain and show that embedding using ternary symbols is optimal. We also establish that pooling pixels to groups to obtain q-ary symbols does not decrease the embedding impact. The paper is concluded in Section 5.

2. PRELIMINARIES

Throughout the paper, bold symbols denote vectors or matrices and capitals denote sets. The function $H(x)$ is the binary entropy $H(x) = -x \log(x) - (1 - x) \log(1 - x)$, where 'log' is the logarithm at the base 2. The inverse of $H(x)$ on the interval $[0, 0.5]$ is denoted by $H^{-1}(x)$. Important concepts are italicized in the text.

2.1 Steganographic embedding scheme

Let X be the set of all possible cover objects $\mathbf{x}$, M the set of all messages $\mathbf{m}$ that can be communicated, and K the key space. Depending on the format of the image, $\mathbf{x}$ could be a vector of integers in the range $[0, 255]$ (for an 8-bit grayscale image) or the range of all integers for quantized DCT coefficients of a JPEG file. The length of $\mathbf{x}$ and $\mathbf{y}$ is equal to the number of elements in the cover object, n, the length of $\mathbf{m}$ corresponds to the maximal number of bits one can communicate – the *embedding capacity*.

A *steganographic scheme* is a pair of embedding and extraction functions Emb: $X \times M \times K \to X$, Ext: $X \times K \to M$ with the property

$$\begin{aligned} \mathbf{y} &= Emb(\mathbf{x}, \mathbf{m}, \mathbf{k}) \\ \mathbf{m} &= Ext(Emb(\mathbf{x}, \mathbf{m}, \mathbf{k}), \mathbf{k}) \end{aligned} \tag{1}$$

for all cover images $\mathbf{x} \in X$, secret messages $\mathbf{m} \in M$, and secret keys $\mathbf{k} \in K$.

2.2 Measure of detectability

The embedding map Emb introduces distortion to the cover object $\mathbf{x}$ so that the stego object $\mathbf{y}$ conveys the desired message $\mathbf{m}$. We assume that Emb either leaves each element of the cover object unchanged or it modifies it in a pre-determined manner. For example, in Least Significant Bit Embedding (LSB), the LSB of x_i is flipped.

The impact of embedding will be evaluated in the following manner. Each pixel is assigned a scalar value (*detectability measure*) that describes the impact of having to change the pixel to embed a message. Sorting these values from the smallest to the largest and normalizing so that the last value is equal to one, we obtain a non-decreasing sequence ρ_i, $i = 1, \ldots, n$, $\rho_i \le 1$, that we will call *detectability profile*. We also assume that the impact of embedding at multiple pixels is an additive function of ρ_i at all modified pixels. In other words, the combined impact of modifying pixels $i_1, \ldots, i_q$ is $\rho_{i_1} + \cdots + \rho_{i_q}$.

The detectability measure is not necessarily equal to the distance between $\mathbf{x}$ and $\mathbf{y}$. As an example, consider the PQ steganography in an 8-bit grayscale image. Let z denote the raw, unquantized value of a given pixel after some processing. The quantization error $e = |z - [z]|$, where $[x]$ is the operation of rounding to the nearest integer. It is known that $e \in [0, 0.5]$ is approximately uniformly distributed in this range when considered as a random variable over all pixels in the image. When in PQ the sender rounds z "to the other side" during embedding, the error becomes $1 - e$. Thus, we can say that the additional embedding distortion is the difference between both errors

$$1 - e - e = 1 - 2e. \tag{2}$$

This is why the authors in [8] proposed to choose for embedding those pixels whose quantization error e is the largest, i.e., closest in absolute value to 0.5. Such values, when rounded to the "other side" experience the smallest embedding distortion.

We can use the embedding distortion (2) as the detectability measure directly or we can multiply (2) by a weight that takes into account the empirical fact that modifications are less detectable in textured areas than in smooth regions. For example, we can choose

$$\rho = \frac{1 - 2e}{1 + \sigma^2}, \tag{3}$$

where σ^2 is the variance of pixels in a local neighborhood of the pixel.

As another example, consider PQ for embedding during JPEG compression. Let z be the unquantized DCT coefficient and let Q be the quantization step for this coefficient from the JPEG quantization table. The quantization error is $e = Q\,|z/Q - [z/Q]|$ and the error when rounding to the opposite direction is $Q\,(1 - |z/Q - [z/Q]|)$ leading to embedding distortion

$$Q\,(1 - 2|z/Q - [z/Q]|) = Q - 2|z - Q[z/Q]|. \tag{4}$$

Note that this detectability measure takes into account the fact that the impact of embedding changes depends on the quantization step Q. Thus, modifying high-frequency coefficients will have a larger impact than modifying a low-frequency DCT coefficient.

2.3 Syndrome coding

Syndrome coding enables minimizing the number of embedding changes if the secret message is shorter than the embedding capacity. This concept was for the first time described by Crandall [12]. A more recent treatment is in [13, 14] and the references therein.

Let us assume that we want to embed m random bits in k pixels using the minimal average number of embedding changes. By allowing up to d embedding changes, it is clear that we cannot embed more than

$$\binom{k}{0}+\binom{k}{1}+\cdots+\binom{k}{d} \tag{5}$$

messages. It is a well-known fact (see, e.g., Lemma 2.4.4 in [15]) that for large k and $d \le k/2$ the expression (5) is asymptotically

$$\binom{k}{0}+\binom{k}{1}+\cdots+\binom{k}{d} \approx 2^{kH(d/k)} . \tag{6}$$

Thus, if we want to embed m bits (i.e., up to 2^m messages), we must have $m \le k\,H(d/k)$, which implies that the number of embedding changes d must be at least

$$d \ge k\,H^{-1}(m/k). \tag{7}$$

Matrix embedding is an embedding method based on linear codes in which the message is communicated as a syndrome for an appropriate linear code. Let us assume that we have a linear $[k, k–m]$ code with covering radius R and parity check matrix $\mathbf{H}$. Furthermore, let $\mathbf{x}$ be the vector of LSBs of the k pixels from the cover object where the payload $\mathbf{m}$ consisting of m bits is to be embedded. The matrix embedding method based on this code modifies $\mathbf{x}$ to $\mathbf{y} = \mathbf{x} + \mathbf{e}(\mathbf{m}–\mathbf{Hx})$, where $\mathbf{e}(\mathbf{m}–\mathbf{Hx})$ is the coset leader of the coset corresponding to the syndrome $\mathbf{m}–\mathbf{Hx}$. In other words, the message is communicated to the recipient as the syndrome $\mathbf{Hy} = \mathbf{Hx} + \mathbf{He} = \mathbf{Hx} + \mathbf{m} – \mathbf{Hx} = \mathbf{m}$, as required. The extraction rule applied by the recipient is simply $\mathbf{Hy}$ or multiplying the LSBs of the stego image pixels by the parity check matrix. This method can embed m bits in k pixels by making at most R changes because the Hamming weight of the difference $\mathbf{x} – \mathbf{y} = \mathbf{e}$ is weight of the coset leader $\mathbf{e}$ and must thus be less than R. Note that in matrix embedding the modified pixels are determined by the coset leader $\mathbf{e}$. Thus, if the message bits form a random bit-stream, the embedding modifications also occur randomly in $\mathbf{x}$.

It is shown in [13] that matrix embedding realized with random linear codes of increasing code length can asymptotically achieve the bound (7) under the assumption of a fixed relative message length m/k. In [9], the authors generalized this scheme to the case when only a subset of pixels known only to the sender is allowed to be modified (so called wet paper codes). Most known structured codes, such as Hamming codes, however do not come very close to the bound. The recent progress in linear binary quantizers using low density generator matrices by Wainwright and Maneva [16], however, seem to provide a venue towards practical syndrome coding schemes with performance very close to the theoretical bound (7). Thus, in the rest of this paper, we will assume that the sender can embed m bits in k pixels by making on average $k\,H^{-1}(m/k)$ embedding changes.

3. MINIMIZING THE EMBEDDING IMPACT

3.1 Discrete case

Let us assume that we have a cover image with n pixels and that we want to embed m bits, $0 \le m \le n$, while minimizing the embedding impact. The sender should ideally make use of *all* n pixels and select them with probabilities determined by their embedding impact. Indeed, this would lead to the *optimal* embedding strategy (see [27]). However, it is not clear how to obtain practical capacity-reaching codes for such schemes. In this paper, we assume that the sender uses a simpler strategy for which syndrome-coding approaches discussed in Section 2.3 can be used.

The sender starts by reserving k pixels, $m \le k \le n$, with the smallest detectability ρ, and then uses capacity-reaching syndrome codes as discussed in Section 2.3. Assuming the message consists of random bits (e.g., if the message is encrypted), the pixels eventually modified by the syndrome coding scheme will be distributed uniformly among the k selected pixels. Thus, the impact of embedding will be on average equal to the expected number of changes × the average embedding impact per pixel:

$$kH^{-1}(m/k)\times\frac{1}{k}\sum_{i=1}^{k}\rho_i = H^{-1}(m/k)\sum_{i=1}^{k}\rho_i . \tag{8}$$

Therefore, the optimal choice of k that minimizes the embedding impact is

$$k_{opt} = \arg\min_{m\le k\le n} H^{-1}(m/k)\sum_{i=1}^{k}\rho_i . \tag{9}$$

For each detectability profile ρ, (9) can be solved simply by enumerating all $n – k +1$ possibilities. The value of k_{opt} is the only parameter that needs to be communicated to the recipient. In particular, note that by applying WPCs [9], the recipient does not need to know ρ or which pixels were used for embedding. The parameter k_{opt} can be communicated, for example, by reserving a small part of the image and embedding its binary encoding (at most $\lceil \log n \rceil$ bits) using some other method. The choice of the matrix embedding method can be arranged between the sender and the recipient so that it is uniquely determined from k and n.

3.2 Continuous case

We now analyze (9) for large n, assuming a fixed relative message length $\beta = m/n$. Denote $x = k/n$. Assuming $\rho_i = \rho(i/n)$ for a non-decreasing function ρ (and thus Lebesque integrable) on the interval [0, 1], we obtain

$$\lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{k}\rho(i/n) = \int_0^x \rho(x)dx . \tag{10}$$

Thus, (9) becomes

$$x_{opt} = k_{opt}/n = \arg\min_{\beta\le x\le 1} H^{-1}(\beta/x)R(x) , \tag{11}$$

where

$$R(x) = \int_0^x \rho(x)dx \,. \tag{12}$$

We can now find the minimum in (11) using calculus. Utilizing the fact that

$$\frac{dH^{-1}(x)}{dx} = \left(\log \frac{1-H^{-1}(x)}{H^{-1}(x)} \right)^{-1} \tag{13}$$

and $R'(x) = \rho(x)$, we obtain that x_{opt} is determined as the solution to the equation

$$\frac{\rho(x)}{R(x)} = \frac{\beta}{\beta x + x^2 \log(1-H^{-1}(\beta/x))} \,. \tag{14}$$

If (14) does not have a solution, in (11) the minimum is reached at one of the end points.

We now analyze (14) in more detail. First, note that for any $0 < \beta < 1$

$$\lim_{x \to \beta^+} \frac{\beta}{\beta x + x^2 \log(1-H^{-1}(\beta/x))} = +\infty \,.$$

This means that (14) can never have $x = \beta$ as a solution. Thus, quite surprisingly, it is never optimal to choose m pixels with the smallest detectability measure for embedding! It is always better to use more than m best pixels. This result holds for the limiting continuous case for large n. For finite values of n in the discrete case, it is possible that the minimum in (9) is achieved at $k = m$ for some detectability profiles ρ.

Let us now take a look at the other extreme – the case when the optimum is reached when taking all pixels ($x = 1$). Equation (14) has $x = 1$ as a solution if and only if

$$\frac{1}{R(1)} = \frac{\beta}{\beta^2 + \log(1-H^{-1}(\beta))} \,. \tag{15}$$

Denoting the right hand side of (15) as $G(\beta)$, we have $\lim_{\beta \to 1^-} G(\beta) = +\infty$ and $\lim_{\beta \to 0^+} G(\beta) = 1$ using the L'Hospitals rule. It is also straightforward to establish by differentiating that G is increasing on [0, 1). Because $1 \le 1/R(1)$ for any detectability profile ρ, we just established that for any ρ, it is always optimal to use *all* pixels for embedding whenever the relative message length $\beta \ge \beta_0$, where β_0 is the unique solution to (15).

The right hand side of (15), the function $G(\beta)$, is shown in Figure 1. Because it increases only very slowly with β, for most detectability profiles using all pixels for embedding only pays off for large messages ($\beta \approx 1$). For example, if we use (2) as the detectability measure, we have $\rho(x) = x$ because the quantization error e is uniformly distributed. In this case, $R(1) = 0.5$, and $\beta_0 = 0.8$. In other words, taking all pixels for embedding is the best strategy only when the payload is larger than 80% of embedding capacity.

Note that when all pixels have the same detectability measure (e.g., $\rho(x) = 1$), it does not matter what pixels are used for embedding. Therefore, the best intuitive strategy is to always use all pixels or $x_{opt} = 1$. This fact is confirmed by inspecting (15) as well as Figure 1 because in this case $R(1) = 1$ and thus, indeed, $\beta_0 = 0$.

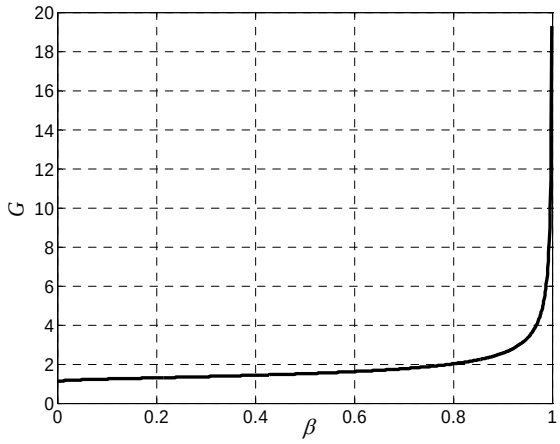

Figure 1. Function $G(\beta)$.

3.3 Detectability profile $\rho(x) = x^p$

Next, we look in detail at a specific class of detectability profiles that can be expressed in the form $\rho(x) = x^p$, where p is a positive parameter. The case $p = 1$ is the proper model for PQ in the spatial domain because the rounding distortion is uniformly distributed on (0.5, 0.5]. The case when $p > 1$ is a good model for PQ in quantities whose distribution has one large peak, such as DCT or wavelet coefficients. Moreover, the analysis of this model of detectability profile is analytically tractable.

When $\rho(x) = x^p$, $\rho(x)/R(x) = x^p/(x^{p+1}/(p+1)) = (p+1)/x$, and (14) becomes

$$\frac{p+1}{x} = \frac{\beta}{\beta x + x^2 \log(1-H^{-1}(\beta/x))} \,, \tag{16}$$

which can be simplified to

$$\frac{x}{\beta} \log\left(1 - H^{-1}\left(\frac{\beta}{x}\right)\right) = -\frac{p}{p+1} \,. \tag{17}$$

Thus,

$$x_{opt} = \begin{cases} c(p)\beta & \text{for } \beta \le \beta_0 \\ 1 & \text{for } \beta > \beta_0 \end{cases} \,, \tag{18}$$

where $c(p)$ is the solution to the following equation

$$c \log\left(1 - H^{-1}\left(1/c\right)\right) = -\frac{p}{p+1} \,. \tag{19}$$

The value $c(p)$ is a multiplicative parameter by which we should multiply the message length to obtain the optimal number of pixels for embedding. Figure 2 shows the value of $c = x_{opt}/\beta$ as a function of the parameter p. In agreement with our previous finding, c is always larger than one meaning that it is always advantageous to use more pixels for embedding than the message length. Also, for the detectability profile $\rho(x) = x^p$, the rule for choosing the optimum value of x (or k, for the discrete case) is particularly simple. When embedding m bits, select for embedding $k = cm$ pixels with the smallest detectability measure. For example, for $p = 1$ (the case of uniformly distributed

embedding impact), $c(1) \approx 1.254$. Thus, the rule of thumb is to use 25% more pixels for embedding than the message length.

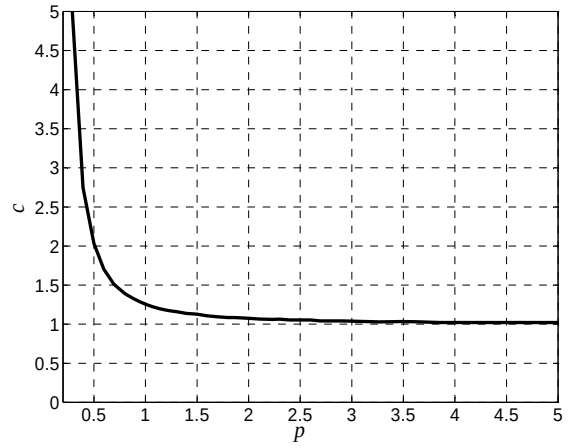

Figure 2 x_{opt}/β for various values of the exponent p.

We now show the decrease of the embedding impact when incorporating the optimal choice of the number of pixels used for embedding. From (11), when using x best pixels for embedding message of relative length β, the embedding impact D is

$$D(x) = H^{-1}(\beta / x)R(x) . \tag{20}$$

We express the decrease in embedding impact as the percentage of $D(\beta)$ – the impact when using the fraction of β pixels with the smallest ρ. Assuming the detectability profile $\rho(x) = x^p$, we obtain from (20) for $\beta \le \beta_0$

$$\frac{D(\beta)}{D(x_{opt})} = \frac{H^{-1}(1/c)\dfrac{(c\beta)^{p+1}}{p+1}}{H^{-1}(1)\dfrac{\beta^{p+1}}{p+1}} = 2c^{p+1}H^{-1}(1/c) . \tag{21}$$

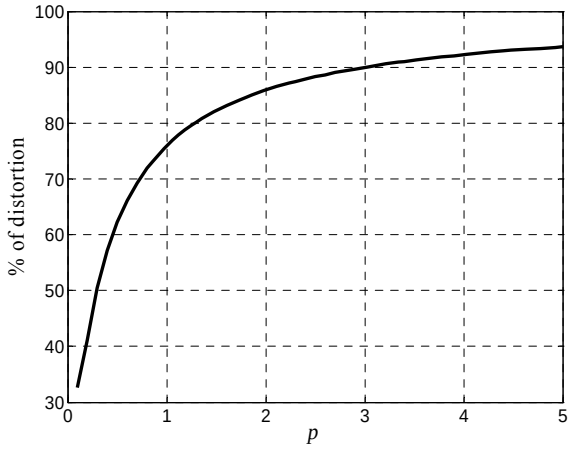

Figure 3 Decrease in embedding impact for various values of the parameter p for detectability profile $\rho(x) = x^p$.

Because for $\rho(x) = x^p$, c is only a function of p, we can plot (21) as a function of p (see Figure 3).

Depending on the detectability profile ρ, the impact of embedding may be reduced by more than 50% (for p approximately less than 0.3). For $p = 1$, which is the case of uniformly distributed detectability measure, the embedding impact is decreased by one quarter when compared to using only the best β pixels.

4. Q-ARY EMBEDDING

We now use the same analysis to investigate embedding schemes that encode messages using a q-ary alphabet and then use embedding operations and syndrome coding mated to alphabet symbols. For example, consider encoding a binary message into a stream of ternary symbols and allowing two changes at each pixel (increase the value by one or decrease by one). This way, a ternary symbol is associated with each pixel and one can apply ternary matrix embedding to minimize the number of embedding changes. Alternatively, we can allow changes by –2, –1, 1, and 2 and encode the message using pentary alphabet in combination with pentary syndrome coding. More details on syndrome coding with q-ary symbols can be found in [14] and [17].

At this point, we would like to point out some differences to the cases treated in the previous section. The PQ paradigm is inherently binary and does not naturally allow using q-ary embedding. This is because the embedding modification in PQ is always chosen to minimize the embedding distortion. Second, the detectability measure ρ is now is a multi-valued function that is the same at each pixel. For ternary embedding, $\rho \in \{-1, 0, 1\}$, while in general for q-ary embedding, $\rho \in \{-\lfloor (q-1)/2 \rfloor, \ldots, \lceil (q-1)/2 \rceil\}$. Let us denote the $q-1$ non-zero elements of this ordered set $\rho^{(1)}, \ldots, \rho^{(q-1)}$.

Larger values of q allow embedding using fewer number of embedding changes at the expense of increasing their amplitude. Because these two trends are working against each other, it is not clear what will eventually lead to a smaller embedding impact.

Reserving k pixels for embedding, we can embed up to

$$\sum_{i=0}^{d} \binom{k}{i} (q-1)^i \text{ bits} \tag{22}$$

by making up to d embedding changes because now the sender has $q-1$ options at each pixel. Because the cost of modifying a pixel is now independent of the pixel, the smallest embedding impact will always correspond to the largest possible k, $k = n$. The sum (22) is the volume of a ball of radius d in the space of k-tuples of symbols from a q-ary alphabet. It is well-known that the volume of a ball is asymptotically (for large k) well approximated as

$$\sum_{i=0}^{d} \binom{n}{i} (q-1)^i \approx 2^{nH_q(d/n)} , \tag{23}$$

where $H_q(x) = H(x) + x \log(q-1)$ is the q-ary entropy function. Similar to the binary case in Section 2.3, this bound is tight in the sense that the ratio of both sides of the inequality approaches 1 with $n \to \infty$, $d/n = \text{const} \le 1 - 1/q$ (for poof, see Lemma 2.4.4 in [15]). Also, there are syndrome coding schemes realized using linear codes of increasing block length whose embedding distortion achieves the bound. Thus, when embedding m bits using the best possible syndrome coding, we can assume that

$$2^m = 2^{nH_q(d/n)} \Leftrightarrow m = nH_q(d/n), \quad (24)$$

which gives us the following distortion per pixel

$$d/n = H_q^{-1}(m/n) = H_q^{-1}(\beta). \quad (25)$$

Assuming we are embedding a pseudo-random stream of q-ary message symbols, when making a change, we are equally likely to choose between the embedding amplitude $\rho^{(1)}$, …, $\rho^{(q-1)}$. Ignoring for simplicity the fact that the values of pixels at the boundaries of the dynamic range cannot be always modified by such amounts, the average embedding distortion per modified pixel $\bar{D} = (\rho^{(1)} + \cdots + \rho^{(q-1)})/(q-1)$ is

$$\bar{D} = \begin{cases} \dfrac{2[1+\cdots+(q-1)/2]}{q-1} & \text{for } q \text{ odd} \\ \dfrac{2[1+\cdots+q/2]-q/2}{q-1} & \text{for } q \text{ even.} \end{cases} \quad (26)$$

Both expressions can be written in a more compact form

$$\bar{D} = \frac{\lceil q/2 \rceil \lceil (q-1)/2 \rceil}{q-1}, \quad (27)$$

which is valid for any q. This allows us to write down the expected value of the embedding impact per pixel (embedding distortion in this case) for the continuous case

$$\frac{1}{q-1}\sum_{i=1}^{q-1} \rho^{(i)} H_q^{-1}(\beta) = \frac{\lceil q/2 \rceil \lceil (q-1)/2 \rceil}{q-1} H_q^{-1}(\beta). \quad (28)$$

In Figure 4, we plotted the expected distortion per pixel for a range of relative message length β and the parameter q. The minimal distortion is always obtained for $q = 3$. This analysis confirms the hypothesis made in [14] based on experiments reported in [18] that it is in general not beneficial to increase the amplitude of embedding changes in exchange for their smaller number.

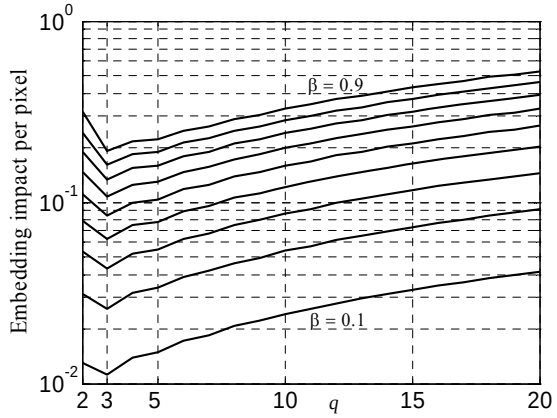

Figure 4 Embedding impact for q-ary embedding for q = 2, 3, …, 20, and β = 0.1, 0.2, …, 0.9.

We note that if we used the energy of modifications as the detectability measure rather than their absolute value (e.g., if we took the square of the embedding distortion instead of its magnitude), we would reach the same conclusion. The smallest impact would be achieved for $q = 3$ because the average energy of an embedding change for $q = 2$ or 3 is the same as their average amplitude and for $q > 3$ the energy is larger then the amplitude.

4.1 Pooling pixels

An obvious attempt to further decrease the embedding impact is to form groups of h pixels and consider them as a symbol from a q^h-ary alphabet. Grouping symbols to vectors is a very fundamental idea that has been very successful in compression and error correction and it is interesting to see if steganography can also benefit from it. Again, we have two conflicting trends here. A larger alphabet will allow embedding more bits per pixel group, however, we now have h-times fewer pixels. Assuming optimal q^h-ary syndrome coding, the average number of embedding changes needed to embed m bits is (from (25))

$$d = \frac{n}{h} H_{q^h}^{-1}(m/(n/h)) = \frac{n}{h} H_{q^h}^{-1}(\beta h). \quad (29)$$

After a moments thought, the average embedding change generalizes from $\bar{D} = (\rho^{(1)} + \cdots + \rho^{(q-1)})/(q-1)$ for $h = 1$ to

$$\bar{D} = \frac{hq^{h-1}\sum_{i=1}^{q-1} \rho^{(i)}}{q^h - 1}. \quad (30)$$

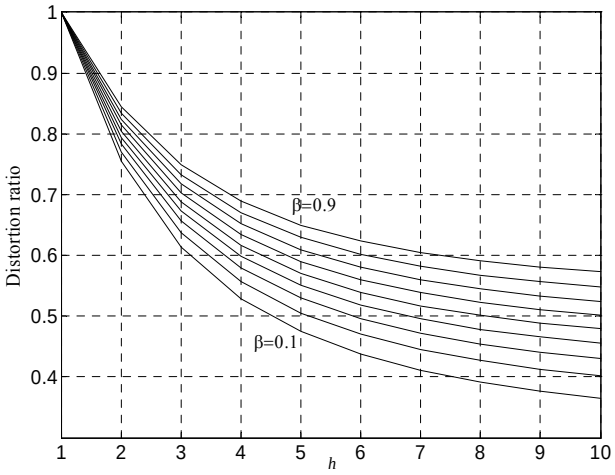

Figure 5 Ratio (32) between distortion when using ternary embedding in single pixels (q = 3) and q^h-ary embedding in groups of h pixels.

Thus, the total average embedding distortion per pixel is

$$\frac{1}{h} H_{q^h}^{-1}(\beta h) \times \frac{hq^{h-1}\sum_{i=1}^{q-1} \rho^{(i)}}{q^h - 1}. \quad (31)$$

The ratio of (28) and (31) is thus

$$\frac{H_q^{-1}(\beta)}{H_{q^h}^{-1}(\beta h)} \sum_{i=0}^{h-1} q^{-i}. \quad (32)$$

This ratio for $q = 3$ as a function of h for various values of β is shown in Figure 5. We can see that grouping pixels does not lead to smaller embedding impact. Similar graphs can be produced for $q > 3$.

An interesting alternative method for pooling pixels has been proposed in [17]. The authors associate with each pixel pair a q-ary symbol in such a manner that allows them to always embed any q-ary symbol in each pair by modifying at most one pixel in the pair by one. Since there are 5 possibilities a pixel pair may be modified (first pixel by 1 or –1, the second pixel by 1 or –1, or both pixels not modified), we can embed a pentary symbol in each pair. An example of assignment of pentary symbols to pixel pairs with grayscales i and j that allows this type of embedding is

	$i \rightarrow$														
j	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4
$\downarrow$	2	3	4	0	1	2	3	4	0	1	2	3	4	0	1
	4	0	1	2	3	4	0	1	2	3	4	0	1	2	3

Although only hypothesized in [17], this can be generalized to groups of h pixels with a $(2h + 1)$-ary alphabet in the following manner. The group of h pixels with grayscale values $(g_1, \ldots, g_h)$ will be assigned the symbol

$$(g_1 + 2g_2 + 3g_3 + \ldots + hg_h) \bmod 2h+1. \quad (33)$$

Since the symbols assigned to the two neighbours along the k-th dimension, $(g_1, \ldots, g_k - 1, \ldots, g_h)$ and $(g_1, \ldots, g_k + 1, \ldots, g_h)$, always differ by $\pm k$, the symbols assigned to all $2d$ neighbours will be pair-wise different. This embedding method could be, for example, conveniently applied to RGB images where three samples are available at each pixel.

The question is, if we adopt this embedding mechanism coupled with optimal syndrome coding, can we obtain a smaller embedding impact than using ternary embedding?

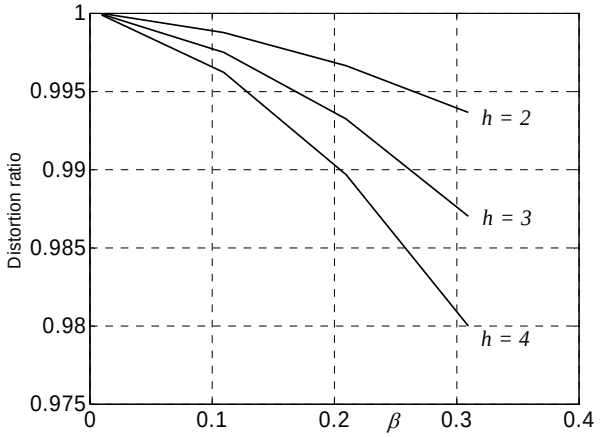

Figure 6 Ratio (34) between distortion when using ternary embedding in single pixels and in groups of *h* pixels.

The average number of embedding changes for ternary embedding (when embedding m bits in n pixels) is $nH_3^{-1}(\beta)$. For groups of h pixels, using the scheme above, the average number of changes is $n/hH_{2h+1}^{-1}(\beta h)$. Because the embedding distortion is always 1 in both cases, the ratio between the embedding distortion for both methods is

$$\frac{hH_3^{-1}(\beta)}{H_{2h+1}^{-1}(\beta h)}. \quad (34)$$

This ratio is plotted for a range of values for β for h = 2, 3, 4, and 5 in Figure 6. We can again see that this method of pooling pixels does not improve the embedding impact either.

We note that for sub-optimal syndrome codes, for example the ones realized using q-ary Hamming codes, this embedding method *can* lead to schemes with a smaller embedding impact (for details, see [17]).

5. CONCLUSIONS

This paper analyzes the trade-off between the magnitude of embedding changes and their number. The fundamental question we tried to answer is whether it is better to make fewer embedding changes with larger embedding impact or more changes with smaller embedding impact. The answer to this question depends on the detectability profile, which is the distribution of the embedding impact among pixels. Based on the assumption that we can perform syndrome coding optimally, we derive an analytic expression for determining the optimal number of pixels that should be used for embedding. For a linear detectability profile, which is most commonly found in perturbed quantization steganography, we determined that as a rule of thumb, the sender should use 25% more pixels for embedding than the message length. We also established that for any detectability profile for sufficiently large number of pixels it is never optimal to only use the pixels with the smallest embedding impact.

Additionally, we analyzed q-ary embedding in the spatial domain and established that ternary embedding is the optimal choice for steganography if our goal is to minimize the embedding distortion. This confirms the heuristic conclusions in [14] that one should not increase the amplitude of embedding changes hoping that their smaller number will lead to a less detectable scheme. This result also justifies the empirical choices made by the authors in [19]. Additionally, we established that grouping pixels to form q-ary symbols does not improve the situation.

While some of the conclusions reached in this paper apply for arbitrary detectability profiles, the quantitative recommendations heavily depend on the detectability profile and should thus be used with caution. It is currently not known how to define the detectability measure compatible with the results obtained by blind steganalyzers. The most frequently used measures are always somehow related to the embedding distortion. It is well known, however, that embedding distortion may be a poor indicator of steganographic security. LSB embedding introduces the smallest possible distortion, yet is easily detectable [20]. Nevertheless, for most other embedding operations the embedding distortion is strongly positively correlated with detectability of the steganographic scheme. This claim is supported by the results obtained for detection of ±K embedding in the spatial domain reported by Soukal et al. [18] and by the attacks on perturbed quantization using blind steganalyzers [21–23].

Another caveat we would like to point out is the assumption that we perform syndrome coding optimally. If sub-optimal syndrome codes are employed, for example binary Hamming codes [11], the conclusions might be different. However, the analysis presented in this paper can still be carried out by replacing the expression

that binds the number of embedding changes *d* and the payload *m* with the appropriate expression derived from the code.

The conclusions reached in this paper concern primarily steganographic schemes that are non-adaptive to image content. We fully acknowledge that incorporating the fact that embedding changes are less detectable in textured areas might change the results obtained in this paper. One might incorporate adaptive schemes [24] by appropriately modifying the detectability measure. Thus, the proposed approach applies to adaptive schemes as well and will be studied in our future work. The detectability measure could also be a second-order property of the local embedding distortion, such as the difference between spatially adjacent pixels. This way we could tailor the approach to steganalyzers that model the cover as Markov chains [23].

Finally, only tests on a large number of images supplied with sensitive blind steganalyzers [19, 21, 25, 26] will decide whether the quantitative conclusions of this paper are, indeed, valid. A discrepancy between the analysis and results from blind steganalyzers can be used as a feedback to determine better detectability measures.

6. ACKNOWLEDGMENTS

The work on this paper was supported by Air Force Research Laboratory, Air Force Material Command, USAF, under the research grant number FA8750-04-1-0112 and AFOSR grant number FA9550-06-1-0046. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation there on. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of Air Force Research Laboratory, or the U. S. Government.

I would like to thank to Rainer Böhme and Miroslav Goljan for discussions and providing valuable comments. Special thanks belong to Petr Lisoněk for providing formula (33).

7. REFERENCES

[1] G.J. Simmons, "The Prisoners' Problem and the Subliminal Channel," *CRYPTO83* – Advances in Cryptology, August 22–24, pp. 51–67, 1984.

[2] F.A.P. Petitcolas and S. Katzenbeisser (editors), *Information Hiding Techniques for Steganography and Digital Watermarking*, Artech House Books, January 2000.

[3] R.J. Anderson and F.A.P. Petitcolas, "On the Limits of Steganography", *IEEE Journal of Selected Areas in Communications*, Special Issue on Copyright and Privacy Protection, vol. **16(**4), pp. 474–481, 1998.

[4] C. Cachin, "An Information-Theoretic Model for Steganography", in Aucsmith, D. (ed.): Information Hiding. 2nd International Workshop. Lecture Notes in Computer Science, vol. 1525. Springer-Verlag, New York, pp. 306–318, 1998.

[5] J. Zöllner, H. Federrath, H. Klimant, A. Pfitzmann, R. Piotraschke, A. Westfeld, G. Wicke, G. Wolf, "Modeling the Security of Steganographic Systems", in Aucsmith, D. (ed.): Information Hiding. 2nd International Workshop. Lecture Notes in Computer Science, vol. 1525. Springer-Verlag, New York, pp. 344–354, 1998.

[6] P. Sallee, "Model Based Steganography", in T. Kalker, I.J. Cox, Yong Man Ro (editors), International Workshop on Digital Watermarking, Lecture Notes in Computer Science, vol. 2939, Springer-Verlag, New York, pp. 154–167, 2004.

[7] S. Katzenbeisser and F.A.P. Petitcolas, "Defining Security in Steganographic Systems", *Proc. SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents IV*, vol. 4675, Electronic Imaging 2000, San Jose, CA, pp. 50–56, 2002.

[8] J. Fridrich, M. Goljan, and D. Soukal, "Perturbed Quantization Steganography", ACM Multimedia and Security Journal, vol. 11(2), pp. 98–107, 2005.

[9] J. Fridrich, M. Goljan, and D. Soukal, "Wet paper Codes with Improved Embedding Efficiency," *IEEE Transactions on Information Security and Forensics*, vol. 1(1), pp. 102–110, March 2006.

[10] Y. Kim, Z. Duric, and D. Richards, "Modified Matrix Encoding Technique for Minimal Distortion Steganography," to appear in *Proc. 8th Information Hiding Workshop*, July 10–12, Washington, DC, 2006.

[11] A. Westfeld, "High Capacity Despite Better Steganalysis (F5–A Steganographic Algorithm)", in Moskowitz, I.S. (editor): Information Hiding. 4th International Workshop. Lecture Notes in Computer Science, vol. 2137, Springer-Verlag, New York, pp. 289–302, 2001.

[12] R. Crandall, "Some Notes on Steganography", posted on Steganography Mailing List, http://os.inf.tu-dresden.de/~westfeld/crandall.pdf, 1998.

[13] F. Galand and G. Kabatiansky, "Information Hiding by Coverings," in *Proc. ITW2003*, pp. 151–154, (Paris, France), 2003.

[14] J. Fridrich, P. Lisoněk, and D. Soukal, "On Embedding Efficiency," to appear in *Proc. 8th Information Hiding Workshop*, Washington, DC, July 10–12, 2006.

[15] G. D. Cohen, I. Honkala, S. Litsyn, and A. Lobstein, *Covering Codes*, vol. 54, Elsevier, North-Holland Mathematical Library, 1997.

[16] M. J. Wainwright E. and Maneva, "Lossy Source Encoding via Message-Passing and Decimation over Generalized Codewords of LDGM Codes," in *Proceedings of the International Symposium on Information Theory*, Adelaide, Australia, 2005.

[17] M. van Dijk and F. M. J. Willems, "Embedding Information in Grayscale Images," in *Proc. of the 22nd Symposium on Information and Communication Theory in the Benelux*, Enschede, The Netherlands, May 15–16, pp. 147–154, 2001.

[18] D. Soukal, J. Fridrich, and M. Goljan, "Maximum Likelihood Estimation of Secret Message Length Embedded using PMK Steganography in Spatial Domain," in *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents VII*, vol. 5681, San Jose, CA, January 16–20, pp. 595–606, 2005.

[19] J. Fridrich, M. Goljan, and T. Holotyak, "New Blind Steganalysis and its Implications," in *Proc. SPIE, Electronic*

Imaging, Security, Steganography, and Watermarking of Multimedia Contents VIII, San Jose, CA, January 16–19, pp. 1–13, 2006.

[20] A. Ker, "A General Framework for Structural Analysis of LSB Replacement," in M. Barni et al. (editors): *Information Hiding. 7th International Workshop*, Lecture Notes in Computer Science, vol. 3727, Springer-Verlag, New York, pp. 296–311, 2005.

[21] J. Fridrich, "Feature-Based Steganalysis for JPEG Images and its Implications for Future Design of Steganographic Schemes", in J. Fridrich (editor): *Information Hiding. 6th International Workshop*, Lecture Notes in Computer Science, vol. 3200, Springer-Verlag, New York, pp. 67–81, 2004.

[22] M. Kharrazi, H.T. Sencar, N.D. Memon: "Benchmarking Steganographic and Steganalytic Techniques," *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents VII*, vol. 5681, San Jose, CA, January 16–20, pp. 252–263, 2005.

[23] K. Sullivan, U. Madhow, B.S. Manjunath, S. Chandrasekaran, "Steganalysis for Markov Cover Data with Applications to Images," to appear in *IEEE Transactions on Information Security and Forensics*, vol. 1(2), June 2006.

[24] M. Karahan, U. Topkara, M. Atallah, C. Taskiran, E. Lin, E. Delp, "A Hierarchical Protocol for Increasing the Stealthiness of Steganographic Methods", *Proc. ACM Multimedia Workshop*, Magdeburg, Germany, September 20–21, pp. 16–24, 2004.

[25] H. Farid and L. Siwei, "Detecting Hidden Messages Using Higher-Order Statistics and Support Vector Machines," in Petitcolas, F.A.P. (editor): Information Hiding. 5th International Workshop. Lecture Notes in Computer Science, vol. 2578, Springer-Verlag, New York, pp. 340–354, 2002.

[26] I. Avcıbaş, M. Kharrazib, N. Memon, B. Sankur, "Image Steganalysis with Binary Similarity Measures," EURASIP JASP, No. 17, pp. 2749–2757, 2005.

[27] J. Fridrich, "Improving Steganographic Security by Minimizing the Embedding Impact," submitted to *SPIE Electronic Imaging, Security, Steganography and Watermarking of Multimedia Contents*, San Jose, California, January 2007.