

# 'Print and Scan' Resilient Data Hiding in Images

Kaushal Solanki, *Member, IEEE*, Upamanyu Madhow, *Fellow, IEEE*, B. S. Manjunath, *Fellow, IEEE*, Shiv Chandrasekaran, and Ibrahim El-Khalil, *Student Member, IEEE*

**Abstract**—Print-scan resilient data hiding finds important applications in document security and image copyright protection. This paper proposes methods to hide information into images that achieve robustness against printing and scanning with blind decoding. The selective embedding in low frequencies scheme hides information in the magnitude of selected low-frequency discrete Fourier transform coefficients. The differential quantization index modulation scheme embeds information in the phase spectrum of images by quantizing the difference in phase of adjacent frequency locations. A significant contribution of this paper is analytical and experimental modeling of the print-scan process, which forms the basis of the proposed embedding schemes. A novel approach for estimating the rotation undergone by the image during the scanning process is also proposed, which specifically exploits the knowledge of the digital halftoning scheme employed by the printer. Using the proposed methods, several hundred information bits can be embedded into images with perfect recovery against the print-scan operation. Moreover, the hidden images also survive several other attacks, such as Gaussian or median filtering, scaling or aspect ratio change, heavy JPEG compression, and rows and/or columns removal.

**Index Terms**—Copyright protection, data hiding, digital watermarking, document authentication, print-scan modeling.

## I. INTRODUCTION

THE advent of the digital age with the Internet revolution has made it extremely convenient for users to access, create, process, copy, or exchange multimedia data. This has created an urgent need for protecting intellectual property in both the digital and the print media. Digital watermarking is a technology being developed, in which copyright information is embedded imperceptibly into the host in a way that is robust to a variety of intentional or unintentional attacks. The ease with which images can be converted from print to digital form and vice versa makes it necessary that the embedded digital watermark is resilient to the print and scan operation.

Strong deterrents against forgery of important documents, such as passports, driving licenses, and ID cards need to be developed at this time, when the concerns over security are higher than ever before. Print-scan resilient data hiding provides a viable solution to this problem: security information (such as fingerprints, signature, or passport number) can be imperceptibly

embedded into a picture in the document. Only specific devices, which have access to a secret key, can decode and authenticate the hidden information. Forgery of such documents becomes extremely difficult because the embedded data are inseparable from the picture.

In this paper, we present methods for hiding information into images in a manner that is robust to printing and scanning. The proposed methods are blind (i.e., the original image is not required at the decoder to recover the embedded data). Using these techniques, several hundred information bits can be embedded into images with perfect recovery after the print-scan operation, which is a significant improvement over the state of the art. An important contribution of this paper is a systematic analytical modeling of the print-scan process by breaking it down into simpler subprocesses, which is appropriately complemented by extensive practical experiments. The analytical and experimental findings form the basis of the proposed embedding schemes, in which data are hidden in dynamically chosen transform coefficients, with synchronization and error correction using powerful turbo-like channel codes. Also proposed is a novel approach for estimating the rotation that an image might undergo during the scanning process, by exploiting knowledge of the digital halftoning scheme employed by the printer. An example is presented in Fig. 1 in which we embed several bytes of information using a technique proposed in the paper, and successfully recover the embedded data after the print-scan operation.

There has been a growing interest among researchers in the area of print-scan resilient embedding, but little progress has been made because of the complex nature of the problem. One of the first approaches was by Lin and Chang [1], who model the print-scan process by considering the pixel value and geometric distortions separately. There are some watermarking methods [2]–[4] that were not specifically designed for the print-scan attack, but they do report robustness against the print-scan operation under a specified experimental setup. Ruanaidh and Pun [2] propose a watermarking method based on the log-polar map of discrete Fourier transform (DFT) magnitudes (i.e., the Fourier–Mellin (FM) transform). Lin and Chang's approach [1] also uses the FM transform to hide information. The technique proposed in [3] involves DFT magnitudes as well, but the watermark itself is made circularly symmetric so that the log-polar coordinate transformation is not required. Bas *et al.* [4] use geometrically invariant feature points to embed the watermark. A few approaches focus on hiding in halftone images [5], [6], wherein the halftone cells of the host image are shifted based on the data to be hidden, and a composite halftone image is given out directly. More recent related works include Voloshynovskiy *et al.* [7] and Mikkilineni *et al.* [8], who focus on document security in general rather than specifically considering printing and scanning of digital images.

Manuscript received August 28, 2005; revised July 30, 2006. This work was supported by a Grant from ONR N00014-01-1-0380. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Gaurav Sharma.

The authors are with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106 USA (e-mail: solanki@ece.ucsb.edu; madhow@ece.ucsb.edu; manj@ece.ucsb.edu; shiv@ece.ucsb.edu; elkhilil@ece.ucsb.edu).

Color versions of Figs. 4, 5, 10, and 11 are available at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TIFS.2006.885032

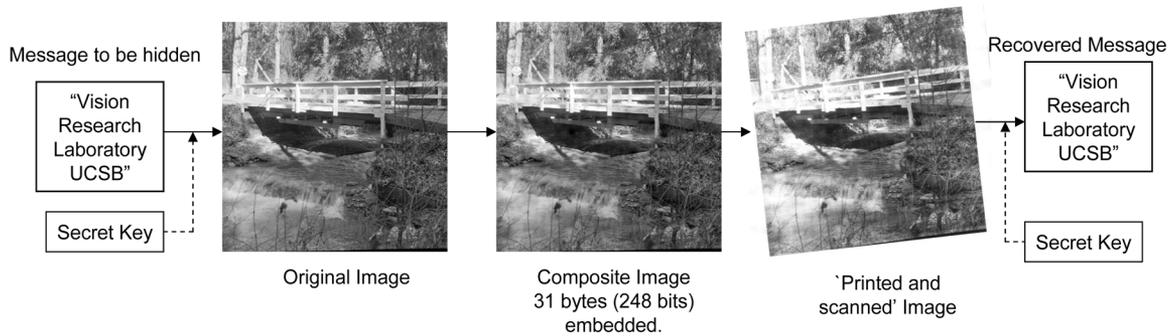


Fig. 1. Example of print-scan resilient data hiding presented in the paper. The number of bits that can be embedded (with perfect recovery) in a typical  $512 \times 512$  image varies from 200 to 500 bits depending on the detail and texture content in the image.

Most of the above methods embed only a single bit (or a few bits) of information, as they assume the availability of the watermark sequence at the decoder. In our recent work on print-scan resilient hiding [9]–[11], an improvement over these methods is achieved in terms of volume of embedding. We are able to hide several hundred bits into images against the print-scan attack with blind decoding. We propose a model for the print-scan process which is comprised of three main components: geometric transformations, nonlinear effects, and colored noise. We infer from the model that data must be embedded into high magnitude coefficients in a band of low frequencies. This is also found to be true in a series of practical experiments to understand the effect of the print-scan process.

Two methods for hiding information resilient to print-scan operation are proposed. The first technique, called selective embedding in low frequencies (SELF), hides data in the magnitude of dynamically selected low-frequency DFT coefficients. This is in contrast to previous DFT magnitude-based approaches (e.g., [1] and [3]), in which a predefined set of midfrequency coefficients is used for embedding. The second method is for hiding data in the phase spectrum of the host image. In this technique, data are embedded by quantizing the difference in phase of the adjacent frequency locations. The method is accordingly termed differential quantization index modulation (DQIM), drawing from QIM, now-famous class of data-hiding methods proposed by Chen and Wornell [12].

We employ turbo-like error and erasure correcting codes to counter the synchronization problem caused by image-adaptive hiding. This also provides robustness to the hidden data against a variety of other attacks such as those in *StirMark* [13] (e.g., heavy JPEG compression, scaling or aspect ratio change, Gaussian or median filtering, rows and/or columns removal, and to a lesser extent, random bending). These attacks were carried out independently of the print-scan attack. Thus, even if the images are released in the digital format, they can survive these attacks along with the print-scan process. We construct a prototype system to demonstrate potential for large-scale deployment of our schemes. We also present results of psycho-visual experiments to demonstrate the perceptual quality of the composite images.

Prior to decoding, the scanned digital image is preprocessed by an automated algorithm for estimating and undoing the rotation caused by random placement of the printed image in the

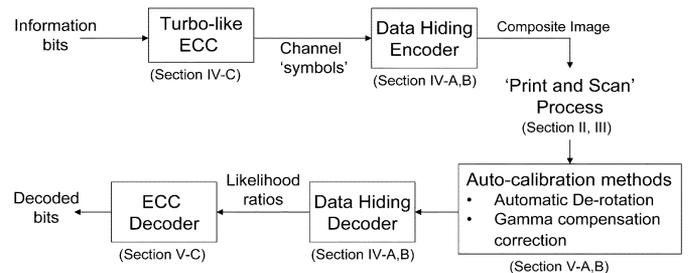


Fig. 2. Outline of how various parts of the embedding schemes fit into the big picture. Below the block, we list the particular section(s) of the paper that discusses it. Note error correcting code (ECC).

scanner. The method is based on the fact that laser printers use an ordered digital halftoning algorithm for printing. The employed derotation method is completely different from the previously used approaches, in which rotation invariance is typically achieved by using FM transform [1], [2]. The advantage of the proposed technique for print-scan resilient hiding is that there is no penalty in hiding rate for achieving robustness against rotation.

A block diagram with the various components of our embedding techniques is provided in Fig. 2. The figure also presents how various sections of the paper are interconnected. The paper is organized as follows. We start, in Section II, with an intuitive and analytical study of the print-scan process. Here, we lay out the three main components of the print-scan model: cropping, nonlinear effects, and colored noise. We then move on to practical experiments, and list the observations made in Section III. Based on the analytical and experimental findings, in Section IV, we propose practical methods to hide data resilient to the print-scan operation. The recovery of the embedded data is discussed in Section V, where we describe a method to estimate and undo rotation undergone by the image during scanning. Numerical results are presented next (Section VI), followed by the concluding remarks in Section VII.

## II. MODELING THE PRINT-SCAN PROCESS

We now present a model for the print-scan operation by breaking it down into simpler subprocesses and study how they distort the image when it is printed and scanned. We know from the watermarking literature that, for robust embedding, data must be hidden in the transform domain. Therefore, in

our model, we specifically analyze the effect of the print-scan process on the DFT coefficients. Before proceeding with a detailed study, let us briefly list the most interesting findings of this section.

- 1) *Frequency bands*: Most components of our print-scan model tend to affect high-frequency coefficients more than the low and midfrequency ones.
- 2) *Effect on DFT magnitude spectrum*: High-magnitude DFT coefficients are preserved better than the low-magnitude ones.
- 3) *Effect on DFT phase spectrum*: The difference in phase of adjacent frequency locations is preserved during the print-scan operation for the high-magnitude coefficients.

By the end of this section, we arrive at the above conclusions based on some simple assumptions. Readers who are interested in practical schemes can skip the rest of the section, keeping the above findings in mind. In the following section, we study the distortions that an image might undergo during the print-scan process. After that, we discuss the three components of our print-scan model (Sections II-B to II-D). Certain issues related to analytical modeling are discussed in Section II-E.

#### A. Distortions During Print Scan

Printing followed by scanning involves conversion from digital to analog and back to digital form. This is inherently a very complex process. The problem is compounded by the fact that a variety of printing and scanning devices are available in the market, which work on one of many different existing technologies. Constructing a unified model will be difficult and, hence, we limit ourselves to laser printers and flatbed scanners.

There have been a few approaches that discuss individual models for printers and scanners. Several models for laser printers that aid the design of digital halftoning methods have been proposed (for example, [14] and [15]). In [14], a model for the electrophotographic (EP) process (the technology employed by laser printers) has been proposed, in which various steps involved in the EP process are analyzed mathematically. This model is then used to design an iterative halftoning method, called direct binary search (DBS). In [15], a physical model is used to train a signal processing model for the printer, which can then be used for halftoning techniques. There have been a few efforts in modeling the scanner as well (e.g., [16] and [17]). In [16], the goal is to calibrate the scanner without using calibration targets. Scanner modeling using specifically designed test targets was accomplished in [17]. In this study, the aim was to perform efficient optical character recognition.

The only prior work that we know of in modeling the print-scan process as a whole is by Lin and Chang [1]. In this work, the authors separate the print-scan distortions into two categories: pixel value and geometric. The model proposed for the pixel value distortion involves a number of parameters, which must be determined experimentally. In our approach, instead of detailed modeling of the print-scan operation as a whole, we divide it into simpler subprocesses, and specifically study the bottleneck components in detail.

Let us now walk through the kinds of distortions an image undergoes when it is printed and scanned, as outlined in Fig. 3. Below we briefly describe each of them.

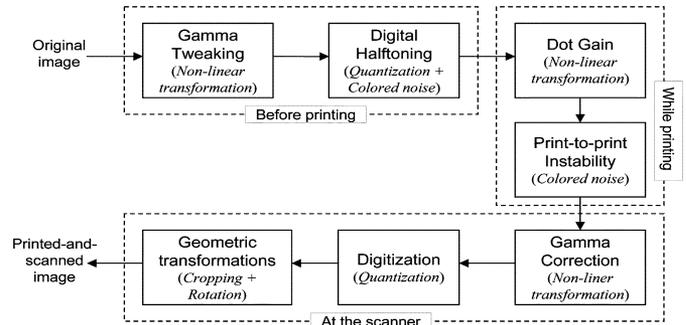


Fig. 3. Various processes that distort the image while undergoing printing followed by scanning.

- **Gamma tweaking**: In order to make sure the printed images appear the same as on a monitor, many printer vendors change the transfer characteristics of the printer to resemble that of an uncalibrated monitor. This nonlinear adjustment is called gamma tweaking.
- **Digital halftoning**: The image is converted to a digital halftone before it is printed. Halftoning algorithms essentially quantize the image into a binary one, typically putting the quantization noise into the high-frequency spectrum of the image. This leads to colored high-frequency noise.
- **Dot gain**: When the digital halftone is printed dot-by-dot, it suffers from a phenomenon called dot gain: the images tend to appear darker than expected due to several reasons (such as spreading of the colorant on the medium, and optical or electrostatic causes). Dot gain is a nonlinear transformation, but it can be roughly approximated by a piecewise-linear curve.
- **Print-to-print instability**: Uncertainties during the printing process can lead to correlated noise. An example of print-to-print instability is banding, which stands for horizontal imperfections appearing in the printouts.
- **Scanner gamma compensation**: When the image is scanned, it must be compensated to make sure it looks fine to us when viewed on a monitor. The scanned image pixel values are raised to a power of  $1/\gamma$ , where  $\gamma$  is the assumed system gamma of the monitor on which the image is to be viewed.
- **Digitization**: The scanned image must be digitized before storing, which invariably leads to quantization errors. Since it follows nonlinear adjustment of the previous step, the effect of quantization noise may get amplified.
- **Geometric Transformations**: At the time of scanning, the image can be subjected to a number of geometric transformations, such as cropping, rotation, and scaling. These effects must be explicitly taken into account because, even with most careful scanning procedure, one cannot completely avoid such geometric transformations.

In the above discussion, we have identified, roughly, various processes that distort the image when it is printed and scanned. We further simplify our study by grouping similar processes together, so as to divide the distortions into three broad categories: geometric transformations, nonlinear effects, and colored noise. We now describe these individual components of our model in more detail (Sections II-B–II-D). As stated before, rotation and

cropping are the main geometric distortions that an image undergoes during scanning. Since we have a method to estimate and undo rotation (to be discussed in Section V), we do not consider rotation for a detailed study here. In the following, we study the effects of image cropping.

### B. Cropping

Consider an image  $f(n_1, n_2)$  with  $N_1$  rows and  $N_2$  columns, so that it is defined over the domain  $\Omega = \{0, 1, \dots, N_1 - 1\} \times \{0, 1, \dots, N_2 - 1\}$ . Cropping of the image can be thought of as a multiplication with a masking window. Assuming that the image is cropped to new dimensions of  $M_1 \times M_2$  (with  $M_1 \leq N_1$ , and  $M_2 \leq N_2$ ), the masking window  $r(n_1, n_2)$ , also defined over  $\Omega$ , can be written as

$$r(n_1, n_2) = \begin{cases} 1, & \text{if } M_{1a} \leq n_1 < M_{1b}, \text{ and } M_{2a} \leq n_2 < M_{2b} \\ 0, & \text{otherwise.} \end{cases}$$

Here,  $M_{1a}$  and  $M_{1b}$  define the top and bottom cropping locations, respectively, so that  $M_1 = M_{1b} - M_{1a}$ . Likewise,  $M_2 = M_{2b} - M_{2a}$ . We can now define the cropped image  $c(n_1, n_2)$  as

$$c(n_1, n_2) = f(n_1, n_2) \times r(n_1, n_2) \quad \forall \{n_1, n_2\} \in \Omega.$$

This product is equivalent to a circular convolution in the DFT domain. Defining  $F(k_1, k_2)$ ,  $R(k_1, k_2)$ , and  $C(k_1, k_2)$  as the two-dimensional (2-D) DFT of  $f(n_1, n_2)$ ,  $r(n_1, n_2)$ , and  $c(n_1, n_2)$ , respectively, the circular convolution can be written as

$$C(k_1, k_2) = \sum_{l_1=0}^{N_1-1} \sum_{l_2=0}^{N_2-1} F(l_1, l_2) \cdot R(\langle k_1 - l_1 \rangle_{N_1}, \langle k_2 - l_2 \rangle_{N_2}). \quad (1)$$

Here,  $\langle \cdot \rangle_N$  denotes the modulo  $N$  operator. The DFT of the masking window  $r(n_1, n_2)$  would be a sinc-like function,<sup>1</sup> with its shape being a function of  $M_1$  and  $M_2$ , and a phase shift that depends on the location of the masking window (i.e.,  $M_{1a}$  and  $M_{2a}$ ). When the cropping is mild,  $N_1 - M_1$  and  $N_2 - M_2$  are small and the sinc-like function would be quite narrow. For mild cropping, most of the energy of  $R(k_1, k_2)$  is concentrated in the  $\{0, 0\}$  coefficient, along with the low-frequency part of the first row and first column. Thus, the blurring of the original image spectrum will be mild for those DFT coefficients whose magnitude is of the same order as its neighbors. However, for coefficients whose magnitude is significantly lower than its neighbors, the blurring will cause its magnitude to increase. This will affect low magnitude coefficients in all frequency bands—high, mid, or low. This is the first significant inference regarding the effect of print-scan processes on the DFT coefficients: the high-magnitude coefficients are better suited for embedding information compared to the low-magnitude ones.

Note that the convolution with sinc-like function can also cause attenuation of peaks (coefficients whose magnitude is much higher than their neighbors) along with filling of holes

<sup>1</sup>Note that  $R(k_1, k_2)$  is a discrete function, which does not strictly follow the sinc definition. It still has a shape similar to the sinc function and, hence, we call it sinc-like.

(coefficients with lower magnitude than their neighbors). However, in the log domain (used by our practical schemes to be discussed in Section IV), the effect of filling of the holes is more pronounced than attenuation of the peaks. Also note that spectra of typical images has a larger number of holes than peaks.

Let us continue focusing on mild cropping, and investigate its effect on the magnitude and phase of the DFT coefficients. It should be noted that the cropping window  $r(n_1, n_2)$  is not known to the decoder and, hence, we cannot simply use deconvolution to estimate the original DFT coefficients. However, under the assumption of mild cropping, and considering only those coefficients that do not have a significantly lower magnitude than their neighbors, we can write the convolution expression (1) with only two dominant terms

$$C(l_1, l_2) = R(0, 0) \cdot F(l_1, l_2) + R(l_1, l_2) \cdot F(0, 0) + \text{other terms.} \quad (2)$$

Once the size of the masking window is fixed (i.e.,  $M_1, M_2$  fixed), the magnitude of  $R(k_1, k_2)$  does not change with the actual location of the masking window (determined by  $M_{1a}, M_{2a}, M_{1b}$  and  $M_{2b}$ ). Furthermore, the blurring caused by mild cropping is not significant for high magnitude coefficients. In other words, for the magnitude spectrum, the contribution from all terms in (2) other than the first one would be small and the variation in the exact location of the masking window would not make a significant difference to the high magnitude coefficients. This leads to the second important inference: embedding data directly into the magnitudes would work.

The phase of  $R(k_1, k_2)$  would vary as the location of the masking window (i.e.,  $M_{1a}, M_{2a}$ ) changes. Looking at the phase shift between the original image spectrum  $F(k_1, k_2)$  and the scanned image spectrum  $C(k_1, k_2)$  from (2), we see that the first term does not cause a phase shift, but the second term does. The amount of shift depends on the phase of  $R(l_1, l_2)$  which, as discussed above, varies with the location of the masking window, but is fixed for a particular instance of the cropped image. Also, since the phase of  $R$  varies slowly, the shift seen by nearby frequency locations is approximately the same. Thus, for the phase spectrum, there is an unknown phase shift between the corresponding original and cropped image DFT coefficients, which varies slowly across the spectrum for mild cropping. This unknown shift can be canceled by taking the difference in phase of adjacent frequency locations. This leads to another inference: data embedding in the phase difference of adjacent DFT coefficients would work.

### C. Nonlinear Effects

The main sources of nonlinear effects during the print-scan process are gamma tweaking, dot gain, and gamma compensation. As seen in Fig. 3, each nonlinear processing block is followed by quantization (digitization or halftoning), which tends to amplify the effect of the quantization noise.

We conducted some preliminary studies to understand the effect of nonlinear processing of images on their DFT coefficients. The nonlinear transformations were simulated using

simple models,<sup>2</sup> and their effect on the DFT coefficients was studied. It was observed that these nonlinear transformations affect the mid and high-frequency coefficients more than the low-frequency ones. Further, we see that in the low-frequency band, only the coefficients with low magnitude were affected. From these observations, we infer that, in general, low-frequency coefficients are better suited for data embedding compared to the high-frequency ones.

Dealing with nonlinearity requires us to calibrate the devices, and/or learn the transfer characteristics experimentally. We do not take this up in the current work mainly because the nonlinear effects do not significantly affect low-frequency coefficients. We do, however, present a practical way to get around incorrect gamma compensation happening at the scanner. The technique, described in Section V-B, can be employed to correct any discrepancy in scanner gamma compensation which may occur when the devices are not calibrated.

#### D. Colored Noise

Before an image is printed, it is converted into a digital halftone. Digital halftoning algorithms tend to put the quantization noise in high frequencies [18] since the human visual system is not very sensitive to high-frequency noise. This introduces high-frequency noise into the image, which can potentially be reduced by using inverse halftoning (see, for example, [19] and [20]). Another source of colored noise is the printing process itself. Uncertainties during the printing operation, or the print-to-print instability, add correlated noise which varies every time a printout is taken. The effect of this component of the model is mostly limited to high and midfrequency coefficients and, hence, we do not analyze this component in more detail here.

#### E. Discussion on Modeling Issues

Of all the three components of our print-scan model, only cropping contributes to distortion in all of the frequency bands equally. The other two components tend to affect mid and high-frequency coefficients more than the low-frequency ones. In their model, Lin and Chang [1] also consider cropping to be an important factor. They view it as an additional source of noise. Moulin and Briassouli [21] consider cropping as well, although not in the context of print-scan. Similar to our observation, they view cropping as causing blurring in the frequency domain. In [10], we present an example comparing the effect of the print-scan attack simulated using the above model versus an actual print-scan process. It is seen that the model closely matches the actual process.

In the print-scan model proposed by Lin and Chang [1], low-pass filtering (or blurring) of the image has been considered via a couple of point spread functions. Voloshynovskiy *et al.* [7] also view the printing process as causing blurring of the image. Here, the authors specifically consider the error diffusion halftoning method, which has been modeled as a combination of two filters [22]. It should be noted that in [1] too, authors use inkjet printers in their experiments, which typically employ

error diffusion halftoning. Since our focus in this work is on laser printers, we do not consider image blurring in our current proposal of the print-scan model. Also, in the printing scenario we consider, the images are printed at high resolution (600 dpi or more), which do not cause significant blurring of the image. Having studied the print-scan operation from an analytical perspective, we now move on to practical experiments in the following section.

### III. EXPERIMENTAL MODELING

We conducted a series of experiments involving printing and scanning of a number of images in order to practically understand the effect of the print-scan process on the transform coefficients, and to determine invariants, in which data could be embedded. Below we discuss the setup and assumptions common to all of our experiments (Section III-A), followed by our findings for the effects on magnitude and phase spectra, respectively (Sections III-B, and III-C). Experimental findings are compared with the analytical model in Section III-D.

#### A. Assumptions and Experimental Setup

The devices involved in this scenario—the printer, the scanner, and the monitor—must be calibrated before use to provide the best results. There is a huge body of literature available on the worldwide web and elsewhere on how to calibrate these devices (e.g., the international color consortium (ICC) profiles). However, we note that most of the devices used by common users are uncalibrated. Hence, to mimic a real-world scenario, we do not explicitly calibrate the devices we use in our experiments. Also note that, for simplicity, we limit ourselves to grayscale images.

Several images were printed and scanned using commercially available laser printers and flatbed scanners.<sup>3</sup> The images were printed at resolutions varying from 300 to 1200 dpi. In the typical printing scenario,  $512 \times 512$  images were printed with 72 pixels per inch (ppi) setting on letter papers, so that the size of the image on the paper is  $7.11 \times 7.11$  in. Widely used Xerox recycled papers (for copiers and laser/inkjet printers) were used for printing. At the time of scanning, the images were cropped and resized using bicubic interpolation to their original size. The resolutions typically used for scanning were 300 to 1200 samples per inch (spi).

Various parameters, such as printer and scanner resolutions, scanner gamma correction, and print image size, were varied and their effects on several image features were studied in order to find features that are invariant to the print-scan operation, across these parameter variations. No effort was made to explicitly register the scanned and original image or their features in the experiments because our goal is to build a blind system where the original image would not be available at the decoder. The DFT coefficients were identified for a more detailed study. The trends presented below were observed across several images and printer/scanner resolutions. Using lower resolutions slightly increases the noise in the system, but does not have a drastic effect.

<sup>2</sup>Gamma compensation was simulated by raising the normalized intensity values by  $1/\gamma$ , and dot-gain was simulated by a piecewise-linear approximation.

<sup>3</sup>Laser printers used in our experiments: Lexmark Optra S 1620, Sharp AL-1641CS, HP LaserJet 2300 Series, HP ColorJet 2600 Series, and HP LaserJet 4050 Series. Scanner used: CanoScan N670U flatbed scanner.

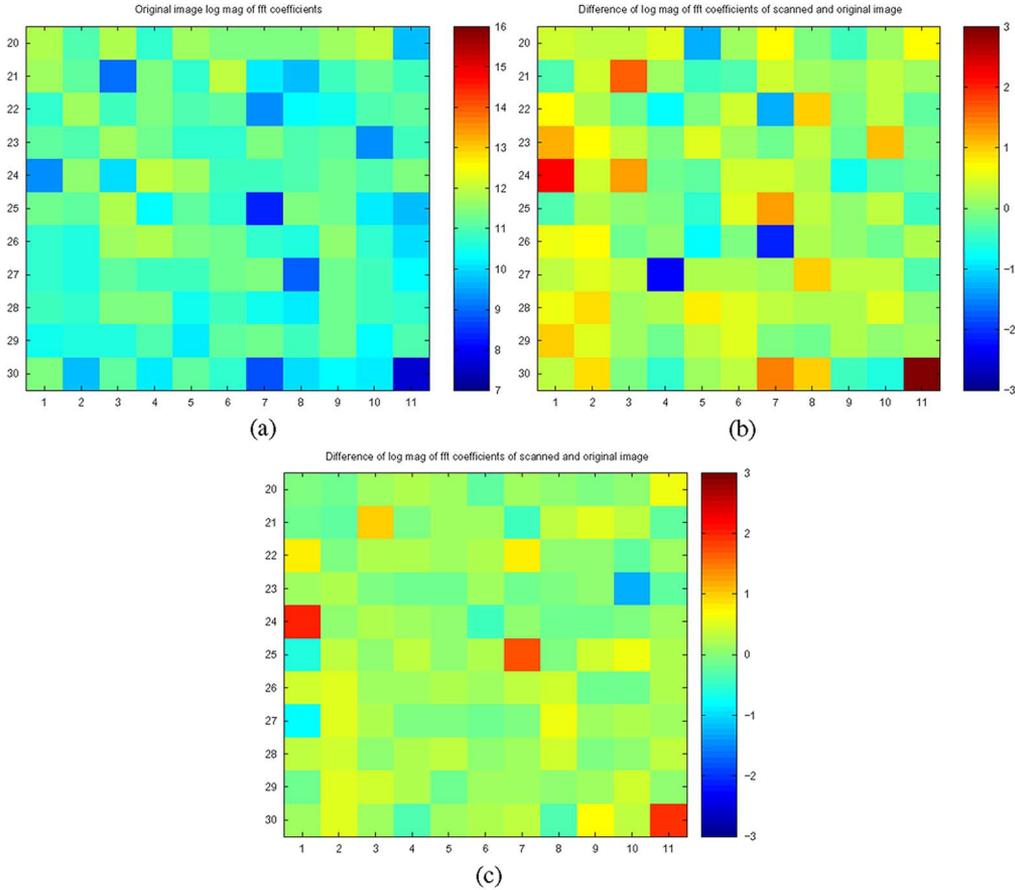


Fig. 4. Print-scan channel: Almost all dark blue coefficients in the original image magnitude spectrum of (a) correspond to dark red points in the log transfer function of (b) and (c) (e.g., (24,1),(25,7),(30,11), and so on). It indicates that the error is high for all coefficients that have low magnitudes. Note that the image in (c) has been printed and scanned with higher resolutions than the one in (b). (a) Original image spectrum in log domain. (b) Difference in log DFT magnitudes of scanned and original image. (c) Another instance of scanned image: diff. in log DFT magnitudes.

### B. Effect on DFT Magnitudes

Below are the experimental observations for the effect of printing followed by scanning on the DFT coefficient magnitudes. Note that, unless otherwise stated, we refer to the natural logarithm of the DFT coefficient magnitudes in the following.

- 1) The low and midfrequency coefficients are preserved much better than the high-frequency ones. In general, the lower the frequency, the better its chances are of surviving the print-scan process.
- 2) In the low and midfrequency bands, the coefficients with low magnitudes get washed out, while those with high magnitudes are preserved much better. It can be seen from Fig. 4 that the coefficients with low magnitudes are hit more severely than their neighbors with higher magnitudes. This is a significant characteristic of the channel and has been observed consistently for different images and various printer or scanner resolutions.
- 3) Coefficients with higher magnitudes (which do not get severely corrupted) see a gain of roughly unity (with the default gamma correction). Roughly speaking, if the print-scan operation is approximated as a linear filter (for large enough coefficients and low enough frequencies), then the magnitude gain is unity after application of gamma correction. One possible explanation is that the printing operation, in itself, does not cause blurring, since several printer dots are dedicated to each pixel of a printed image.

- 4) Slight modifications to the selected high magnitude low-frequency coefficients do not cause significant perceptual distortion to the image.

To further verify the observation that low-magnitude coefficients see a higher noise, we conducted another set of experiments in which the values of coefficients were artificially modified. The results, presented in Table I, clearly indicate that low-magnitude coefficients see higher noise irrespective of the fact that they are low values originally or were modified to a lower value (from a higher value) before printing. On the other hand, if a coefficient has high magnitude, it typically sees relatively low noise.

### C. Effect on Phase Spectrum

Our analysis of the model for the print-scan process (in Section II-B) suggests that the difference in phase of adjacent frequency locations would be preserved during the print-scan process. Here, we practically investigate the effect on phase difference of neighboring frequency locations. The following observations are made.

- The phase difference for the high-frequency locations sees very high noise.
- For the low-frequency coefficients, the phase difference of adjacent locations is preserved for coefficients whose magnitude is high. Fig. 5 shows the difference in the phase difference for original and scanned images for two different

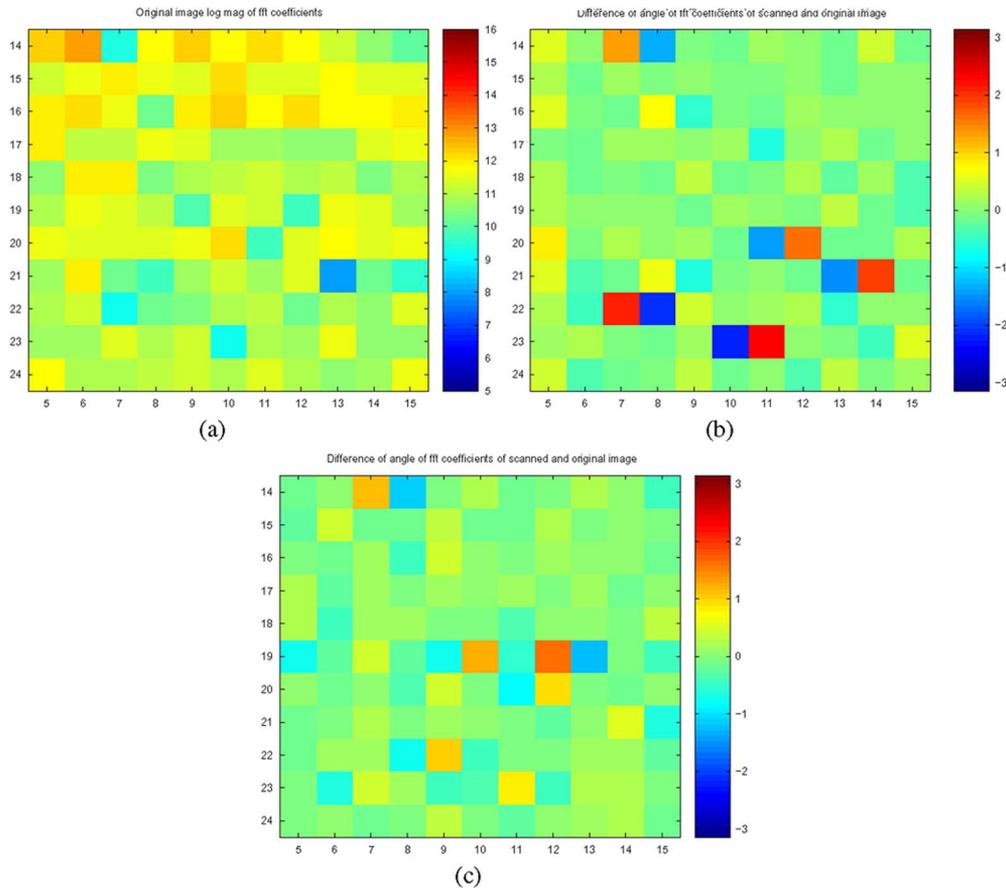


Fig. 5. Effect on phase spectrum during print-scan: The phase difference of adjacent frequency locations is preserved except for those coefficients whose magnitude is lower than their neighbors (e.g., (14,7), (22,7), (23,10), and so on). The exact effect also varies for different instances of scanned images. (a) Original image spectrum in log domain. (b) Difference in the difference of phase of adjacent frequency locations for the scanned and original image. (c) Another instance of scanned image: Difference in the difference of phase of adjacent frequency locations.

TABLE I  
EXAMPLE: COMPARISON BETWEEN THE NOISE OBSERVED DUE TO PRINTING AND SCANNING AT SPECIFIC LOCATIONS IN THE FREQUENCY SPECTRUM AND THE NOISE OBSERVED AFTER THE MAGNITUDE OF THOSE COEFFICIENTS IS DECREASED WITH RESPECT TO NEIGHBORS OR INCREASED TO A VALUE COMPARABLE TO NEIGHBORS

Frequency Index	Original Coeff. log magnitude	Noise	Modified Coeff. log magnitude	Noise
(2,2)	14.44	-0.083	9	0.956
(10,10)	7.98	2.104	12	-0.0741
(6,7)	9.20	0.5231	11.2	-0.0267
(3,3)	13.23	0.0358	10	1.5446

instances of a printed-and-scanned image. It is observed that the phase difference for coefficients with lower magnitudes is severely corrupted. Note that since we are taking the difference of two frequency coefficients, as seen in the figure, a high error in one gets carried to the next location as well.

#### D. Experimental Observations and the Print-Scan Model

We conclude this section by noting that the experimental observations are quite consistent with the analytical inferences made from the model. Our investigation of colored noise and nonlinear effects suggests that high-frequency coefficients are not good for embedding data, which indeed turns out to be the

case practically. In the experiments, we observe that low magnitude coefficients are affected much more than their high magnitude neighbors, a phenomenon that was also predicted by our analysis of the effect of cropping. For the phase spectrum, the analysis suggests that the difference of adjacent frequencies is likely to be preserved, which again, is observed practically as well. Based on all of these findings, we now propose practical print-scan resilient embedding methods in the following section.

#### IV. PRINT-SCAN RESILIENT EMBEDDING

Before discussing the embedding schemes in detail, let us first revisit the block diagram provided in Fig. 2. We now redraw the diagram with more specific details of the employed embedding mechanism in Fig. 6. The system is divided into three main layers: autocalibration at the receiver, data hiding layer, and the coding framework. We study these layers as we proceed in this paper. In the rest of this section, we discuss the hiding methods and the coding framework.

Two practical embedding schemes are proposed. The first is the selective embedding in low frequencies (SELF) scheme that embeds data into the magnitude spectrum of the host image, and the second is the differential quantization index modulation (DQIM) method for hiding in the phase spectrum.

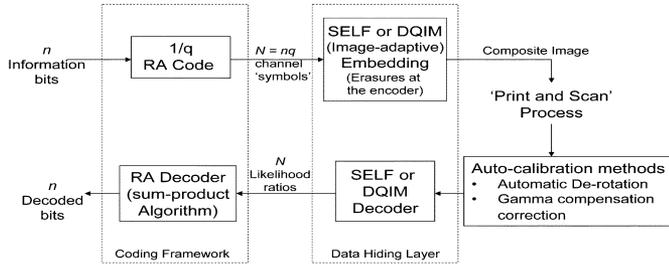


Fig. 6. Overview of how various parts of the embedding schemes fit into the overall system.

### A. SELF Selective Embedding in Low Frequencies

Based on the experimental and analytical modeling of the print-scan process described in the previous sections, we propose a hiding method in which information is hidden into selected high-magnitude low-frequency coefficients. Hence, the name: SELF.

While the idea of embedding into low-frequency bands is not entirely new, the main novelty of this scheme lies in its adaptive design based on experimental and analytical modeling. The use of turbo-like codes (discussed in Section IV-C) allows error-free recovery for several hundred embedded bits despite a dynamic embedding strategy and the presence of severe attacks. Most prior adaptive watermarking schemes are based on spread-spectrum embedding and end up hiding only a few bits of data. Below, we describe the method in more detail.

Consider an  $N \times N$  host image in which data are to be hidden. Let us denote the natural logarithm of the magnitudes of the 2-D DFT of the whole image by  $c_{ij}$ ,  $0 \leq i, j \leq N - 1$ . We embed in a given coefficient  $c_{ij}$ , only if it lies in a predetermined frequency band and exceeds a threshold  $t_{ij}$ . Let us define the band as an indicator function  $b_{ij}$ , such that if  $b_{ij} = 1$ , the coefficient  $c_{ij}$  lies in the band. Note that  $b_{ij}$ ,  $t_{ij}$  and the quantization interval  $\Delta$  are design parameters that are shared between the encoder and the decoder. Embedding is done using the choice of scalar quantizers. We send either  $Q_0(c_{ij})$  or  $Q_1(c_{ij})$  depending on whether the bit to be hidden  $x_n$  is 0 or 1, where  $Q_0(\cdot)$  and  $Q_1(\cdot)$  are uniform scalar quantizers with the quantization interval  $\Delta$ , shifted by  $\Delta/2$  from each other. Thus, the modified coefficient  $d_{ij}$  can be given as

$$d_{ij} = \begin{cases} Q_{x_n}(c_{ij}), & \text{if } b_{ij} = 1, \text{ and } c_{ij} > t_{ij} \\ c_{ij}, & \text{otherwise.} \end{cases} \quad (3)$$

Also note that the symmetry of the DFT coefficients is maintained during the hiding process by modifying two symmetric coefficients in the same manner so that the inverse DFT gives real values. Finally, taking exponential, adding phase, and taking inverse Fourier transform gives the hidden image intensity values.

The choice of the candidate embedding band, the threshold(s), and  $\Delta$  is done empirically through experimentation with several images. The goal is to hide as much information as possible without causing perceptual distortion to the image while maintaining a low error rate. The value for  $\Delta$  we use in our experiments is typically 1.0 (in the log-DFT domain). Using a higher value for  $\Delta$  causes perceptual distortion to the image, while using

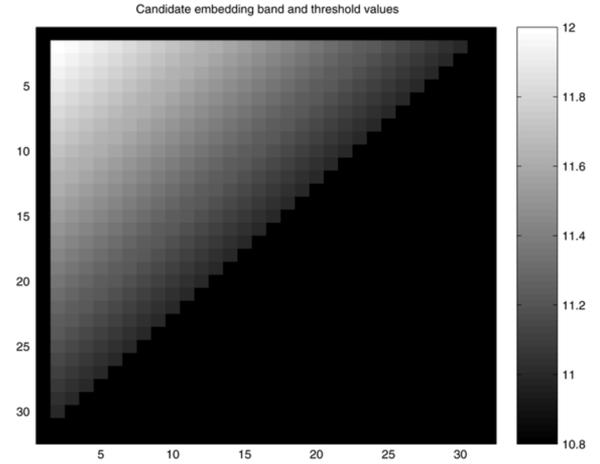


Fig. 7. Typically used candidate embedding band and threshold values: Only one quadrant is shown here with the black part indicating that the coefficients are not in the band. Threshold values are shown for the coefficients that are inside the band. Notice how the threshold value decreases as we go toward higher frequencies. Note that the numbers shown here are for a  $512 \times 512$  image and do not include the  $1/N^2$  scaling in computing the DFT.

a lower value increases the error rate significantly. Perceptual considerations influence our choice of the candidate embedding band  $b_{ij}$  as well. The size of the frequency band used for several sample images is provided in Section VI.

The threshold varies with respect to the frequency band, which follows the same trend as the image spectrum itself. It is known that images have a significant low-frequency component and, in general, the magnitude of the coefficients decreases as we move to the higher frequencies. The coefficient threshold  $t_{ij}$  is chosen such that it also decreases linearly as we move toward higher frequencies. A typical (example) band along with the threshold values is shown in Fig. 7. Since we dynamically chose the embedding locations, we must deal with the synchronization problem inherent to image-adaptive hiding schemes, which we discuss later in Section IV-C. We now move on to the DQIM hiding scheme for the phase spectrum.

### B. Differential Quantization Index Modulation

QIM, proposed by Chen and Wornell [12], is a class of information hiding methods, in which data are embedded into the host sample by the choice of quantizer. Here, we propose a new quantization-based method for data hiding with the goal of surviving mild cropping and the print-scan process. Instead of just quantizing the host signal, we embed data by quantizing the difference of two adjacent host samples. The idea of hiding in the difference of adjacent locations is analogous to differential phase-shift keying (DPSK), used to combat the effect of unknown channel phase shifts in wireless communication. We employ similar nomenclature, and term the proposed method DQIM.

We use DQIM to embed information in the phase spectrum of the images to counter unknown phase shift induced due to mild cropping. As discussed in Section II-B, cropping is equivalent to the circular convolution of the image spectrum with a sinc-like function. This leads to a phase shift between the original and scanned image, which varies slowly across the spectrum of the image. This unknown shift can be canceled by em-

bedding data in the difference of adjacent frequency locations. This inference has also been observed in our practical experiments (Section III-C). Below we describe how embedding in the phase differences is practically implemented.

We first scan the image phase spectrum row-wise. Note that only those coefficients that lie in a predefined band are used for embedding information. Let us denote the row-wise-scanned original image phase values by  $\phi_n$ , where  $n$  is the index ( $n \in \{0, 1, 2, \dots, N_{\max}\}$ ), and the quantized values by  $\theta_n$ . Then, the embedding function is

$$\theta_n = \theta_{n-1} + \langle Q_{x_n}(\phi_n - \theta_{n-1}) \rangle_{2\pi} \quad \forall n \in \{1, 2, \dots, N_{\max}\}.$$

Here,  $Q_{x_n}(\cdot)$  denotes an uniform scalar quantizer chosen based on the bit  $x_n$  to be embedded. For a binary alphabet  $x \in \{0, 1\}$ , we have two quantizers,  $Q_0(\cdot)$  and  $Q_1(\cdot)$ , shifted by  $\Delta_\theta/2$  from each other, where  $\Delta_\theta$  is the quantization interval. Note that since we are dealing with phase, we output modulo- $2\pi$  values after the quantization  $Q_x(\cdot)$  of the difference is done ( $\langle \cdot \rangle$  denotes a modulo operation). We use the quantized values  $\theta_n$  to compute the phase difference for the next coefficient. This is done to maintain consistency for the decoder, which just finds these differences, and determines which of the two quantizers was used.

As discussed before (Section II-B), the assumption of slowly varying phase shift is not valid for those coefficients whose magnitude is significantly lower than its neighbors. Hence, we avoid hiding in these locations, and use turbolike repeat-accumulate (RA) codes to counter the synchronization problem caused due to adaptive hiding, as discussed below.

### C. Coding Framework for Synchronization

An erasure and ECC framework is used to counter the desynchronization problem caused due to the fact that the proposed methods dynamically choose the embedding locations. Readers are referred to our previous work [23], [24] for a detailed account of the coding framework, in which a local adaptive criteria was used to preserve the perceptual quality of the hidden image. Here, we briefly discuss the main ingredients of the framework, and describe how it is adapted for the proposed methods.

Both the methods, the SELF hiding scheme for embedding in magnitudes, and the DQIM method for embedding in phase are image-adaptive methods, in which the encoder selects DFT coefficients to embed based on a threshold criteria. The decoder does not have explicit knowledge of the locations where data are hidden, but employs the same criteria as the encoder to guess these locations. The distortion due to attacks may now lead to insertion errors (the decoder guessing that a coefficient has embedded data, when it actually does not) and deletion errors (the decoder guessing that a coefficient does not have embedded data, when it actually does). In principle, this can lead to desynchronization of the encoder and decoder.

An elegant solution based on erasure and ECCs is provided to deal with the synchronization problem caused by the use of local adaptive criteria. The bit stream to be hidden is coded, using a low-rate code, assuming that all host coefficients that lie in the candidate embedding band will actually be employed for hiding. A code symbol is erased at the encoder if the local adaptive criterion (i.e., the threshold criterion) for the coefficient is

not met. Specifically, we use RA codes [25] in our experiments because of their simplicity and near-capacity performance for erasure channels. A rate  $1/q$  RA encoder involves  $q$ -fold repetition, pseudorandom interleaving, and accumulation of the resultant bit stream. Decoding is performed iteratively using the sum-product algorithm [26].

Let us consider an example wherein we want to hide in a  $512 \times 512$  image. The candidate embedding band is a design parameter known to both encoder and decoder. Let us assume that the band spans 1000 coefficients. Suppose we want to hide 200 bits into the image. We would use a  $1/5$  RA code (i.e.,  $q = 5$ ), which gives a codeword length of 1000. This codeword is now hidden using the adaptive criteria such that if a coefficient does not pass the threshold test, the corresponding code symbol is erased (i.e., not hidden). Note that the RA code rate and the number of bits hidden are predetermined at the design state, and are chosen in such a way that the codeword length is equal to, or slightly greater than the number of candidate embedding coefficients. When the codeword length is greater than the size of the band, the excess code symbols are erased at the encoder.

## V. RECOVERY OF EMBEDDED DATA

We now discuss how the embedded data are recovered and decoded. We first describe two autocalibration procedures: a method for automatic derotation, followed by a technique to deal with incorrect gamma compensation. After that, we briefly discuss the decoding strategy.

### A. Estimating and Undoing Rotation

A novel method, based on printer's digital halftoning algorithm, to estimate the rotation that an image might undergo during the scanning process is proposed in this section. An advantage of the proposed technique for print-scan resilient hiding is that there is no penalty for estimating and undoing rotation, which is unlike previous approaches [1], [2] that typically use FM transform to achieve rotation invariance. It should be noted that the proposed derotation technique cannot be applied to a general rotation attack (e.g., if the image is rotated digitally) since it uses the printer halftone screen to estimate the rotation angle.

Laser printers employ an ordered halftoning algorithm to generate the binary image. In most laser printers, the cells lie in a deterministic periodic array oriented at an angle of  $45^\circ$  for grayscale images. This is because there is a sharp minimum in perceptual sensitivity for spatial frequencies oriented at  $45^\circ$  from horizontal. Note that some modern printers use a different orientation angle ( $33^\circ$ ) when printing at certain specific settings, but the method presented here remains perfectly valid for all of these cases. The idea is to capture the halftone pattern by high resolution scanning, which is then used to estimate the rotation angle.

The angle by which an image gets rotated during the scanning process can be estimated using the fact that the halftone cells in the printout (of the image) lie in a grid of horizontal and vertical cells. Figs. 8(a) and (c) show magnified portions of a printed and scanned image without rotation and with rotation during scanning. Figs. 8(b) and (d) show the magnitude spectrum of the

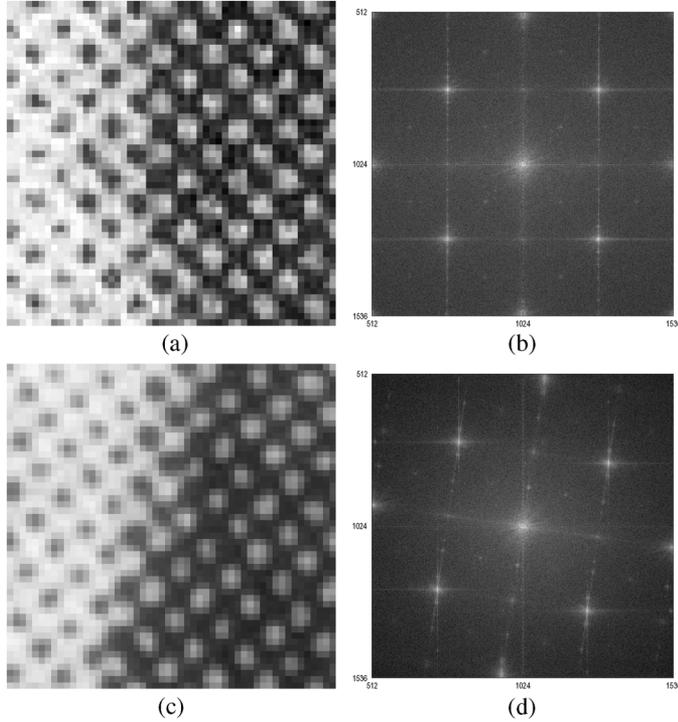


Fig. 8. Zoomed printed-and-scanned images and their Fourier spectra. (a) No rotation. (b) Spectrum of (a). (c) With rotation. (d) Spectrum of (c).

images in Figs. 8(a) and (c) respectively. Due to the orientation of the halftone cells, a peak can be seen at an angle of  $45^\circ$  for the image without rotation. When the image gets rotated during the scanning process, the peaks also get rotated as in 8(d). Note that a number of secondary peaks are observed, but only a part with the primary peaks is displayed here. The angle of the orthogonal lines on which the peaks can then be used to estimate the rotation that the image needs to be derotated with before the hidden data is decoded.

It should be noted that the Fourier transform is symmetric such that out of the four quadrants, the values are equal for a pair of quadrants (for the displayed FFT-shifted spectrum, quadrants I and III have same values, and so do quadrants II and IV). In practice, one cannot use the measurement from one quadrant only because of the following observation. The size of an image on the printout is not exactly the same as that in the digital form. For example, when a  $512 \times 512$  image is printed with 72 pixels per inch, the height measured on the printout turns out to be about 0.05 in. longer than its width. Due to this discrepancy, the angle measured for a peak in the first quadrant of the Fourier magnitude spectrum is slightly different from that in the second quadrant. Below, we show how we can reliably compute the rotation angle despite this discrepancy.

Approximating the halftone pattern as a 2-D impulse train, as in Fig. 9, printer inaccuracy (and also nonsymmetric halftoning screens) can be modeled as scaling along the  $y$ -axis as in 9(b). We show that the net rotation angle can be calculated from the peaks' angles. Let  $\alpha_1$  be the angle with respect to the  $u$ -axis of the impulse in the first quadrant and  $\alpha_2$  be the angle with respect to the  $v$ -axis of the impulse in the fourth quadrant of 9(d). Then,  $\omega = (\beta_1 + \omega + \beta_2 + \omega/2) - (\pi/4) = (\alpha_1 + \alpha_2/2) - (\pi/4)$ ,

where the second equality follows from the fact that  $\beta_1 + \beta_2 = (\pi/2)$ . Hence,  $((\alpha_1 + \alpha_2/2) - (\pi/4))$  yields the angle of rotation  $\omega$  of the impulse train in the spatial domain.

In the following, we describe the algorithm used in estimating and derotating an image after scanning (at 600-dpi resolution).

- 1) Crop a block of  $2048 \times 2048$  pixels from the center of the scanned image and take its DFT.
- 2) Find peaks (location of the maximum values) in the magnitude spectrum for the first and fourth quadrants (with reference axes as in the above paragraph). Let these angles (in degrees) be denoted by  $\alpha_1$  and  $\alpha_2$ .
- 3) Compute the estimate of the rotation angle as  $\hat{\omega} = (\alpha_1 + \alpha_2)/2 - (\pi/4)$  and use bicubic interpolation to rotate the image by  $\hat{\omega}$ .
- 4) The image is then cropped from the background by finding the edges with largest magnitudes of transition (first-order difference) in intensity values.

The algorithm can be easily modified to work when the rotation angle is more than  $\pm 45^\circ$ . Due to the symmetry of the Fourier transform coefficients, the embedding schemes are also symmetric (Section IV). As a result, if the images are rotated  $180^\circ$ , the information can still be decoded without any modification to the decoding algorithm. Hypothesis testing is used to deal with rotation angles of  $90 \pm 45$  (or  $270 \pm 45^\circ$ ). In the implementation, the receiver attempts to decode after the image is automatically derotated and cropped; if it fails, the image is rotated  $90^\circ$  and the decoding is tried again.

### B. Dealing With Incorrect Gamma Compensation

Every computer monitor has an intensity to a voltage response curve which is a power function with parameter  $\gamma_m$ . In order for the scanned image to be correctly displayed on a monitor, the image data generated at the scanner are "gamma corrected" (i.e., raised to a power  $1/\gamma_c$ , where  $\gamma_c = \gamma_m$  for calibrated systems). The correction applied at the scanner depends on the gamma of the monitor or the screen on which the image is to be displayed. The gamma of an uncalibrated Macintosh is accurately 1.72 and that of a PC system is 2.50. The default compensation is placed at 2.2, a value between the two, as defined in the sRGB standard for the Internet images. If the monitor or the scanner is not calibrated, there could be a mismatch, which must be corrected.

We experimented with various gamma correction values at the scanner in order to find a way to deal with incorrect gamma compensation. Similar to previous experiments, we study the logarithm of DFT coefficient magnitudes here. We observed that when the gamma correction is varied at the scanner, the logarithm of DFT coefficient magnitudes of the scanned image is scaled by a constant factor. Fig. 10 plots the original and scanned image DFT coefficients (or the input/output characteristics) for the default gamma correction ( $\gamma_c = 2.2$ ) and for over-correction ( $\gamma_c = 5.0$ ). Fig. 10(c) shows the same plot when the scanned image DFT coefficient magnitudes are scaled. It can be seen that the plot in (c) is quite close to the unity gain line.

Thus, we can deal with incorrect gamma compensation simply by scaling the log DFT coefficient magnitudes. In order to understand how this scaling factor varies with gamma, we conducted another set of simulations and experiments. Gamma

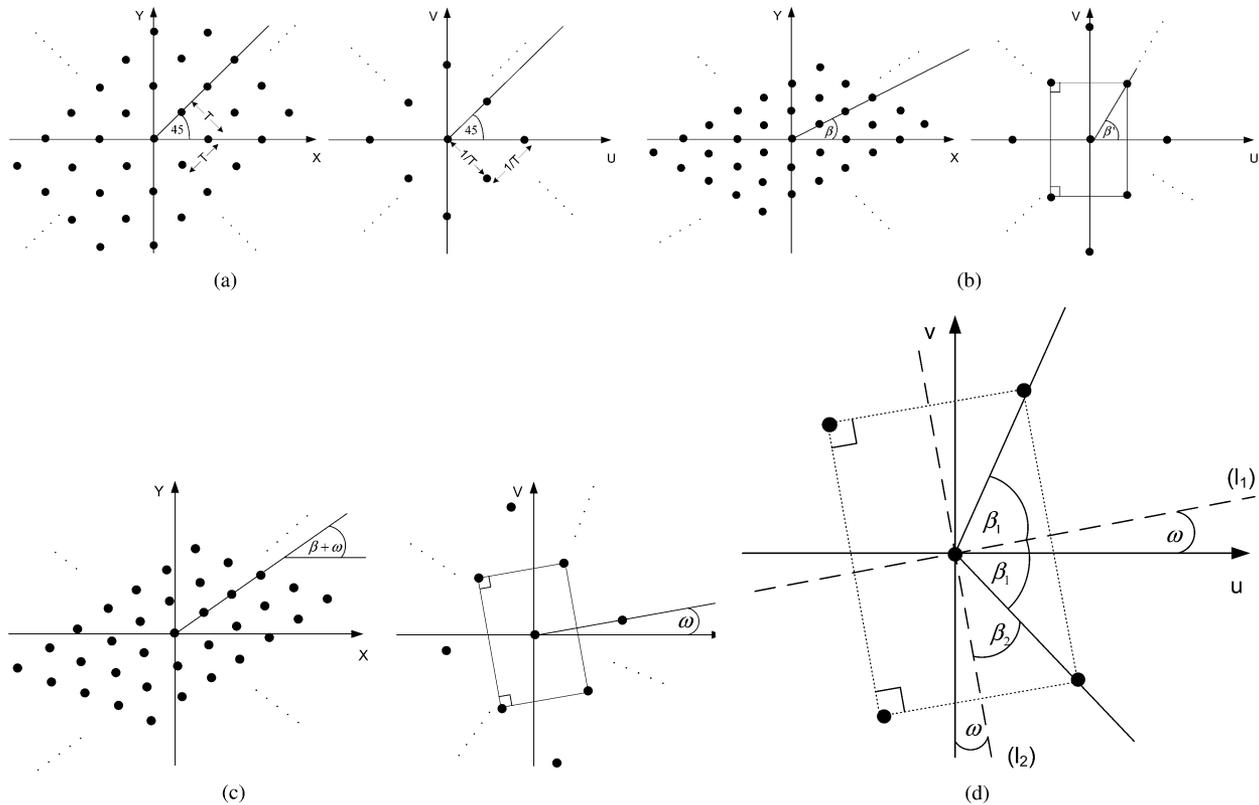


Fig. 9. Effect of scaling and rotation on a 2-D impulse train. Approximating the halftone pattern as an impulse train, we can show that the rotation angle can be computed from the peak angles in the first and fourth (or third and second) quadrants. (a) 2-D impulse train rotated  $45^\circ$  and its Fourier transform. (b) The impulse train in (a) after being scaled along the  $y$ -axis. (c) Impulse train of (b) rotated by an angle  $\omega$  and its Fourier transform. (d) Geometry of the peaks: Fourier transform from (c) showing the four nearest impulses to the origin.

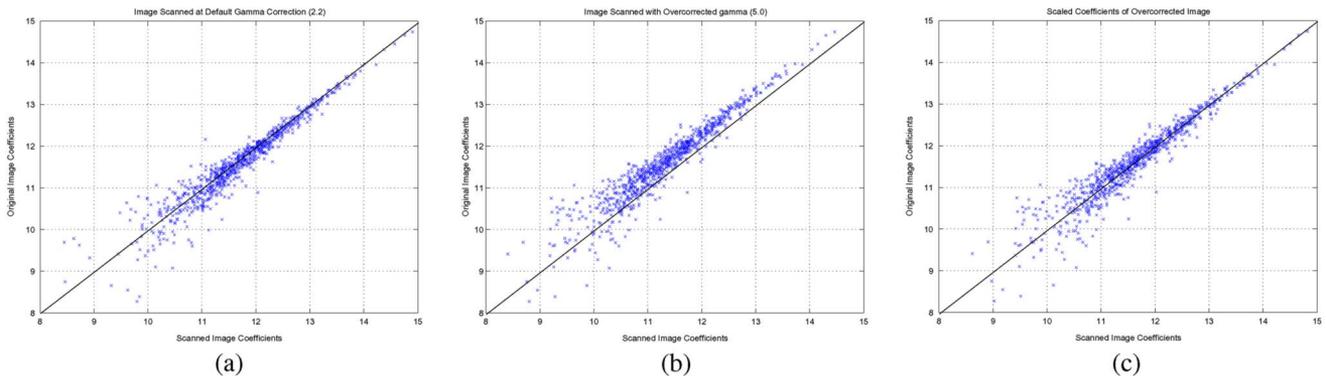


Fig. 10. Effect of gamma correction: Logarithm of low-frequency DFT coefficient magnitudes of original  $512 \times 512$  peppers image is plotted against those of the same image after printing and scanning.  $1/N^2$  scaling has not been applied in computing the DFT. It can be seen that the plot is spread around the  $x = y$  line for the gamma correction of (a). If the image is overcorrected at the scanner (b), the response shifts. However, a plot spread around  $x = y$  can be achieved by scaling the coefficients (c). (a) Image scanned with a gamma correction of 2.2. (b) Image scanned with a gamma correction of 5.0 (overcorrection). (c) The coefficients of the overcorrected scanned image of (b) are scaled by 1.023.

compensation was simulated on the images assuming a monitor gamma of 2.2. The simulated values were obtained by reading an image, applying gamma of 2.2 (raising to  $\gamma_m = 2.2$  after normalizing), digitizing, and then applying different gamma compensation values (raising to power  $1/\gamma_c$  after normalizing). The log-magnitude coefficients of the Fourier spectrum of the simulated images were plotted against the original ones and the best linear fit that passes through the origin was calculated. The slope of this regressor determines the best scaling factor to use.

The scaling factors for an actual process were determined in the same manner: the simulated images were replaced by actual scanned ones and different gamma compensation values were applied at the time of scanning. Fig. 11 plots the scaling factors versus gamma compensation values for the actual print-scan process and for the simulated process. We see that the actual print-scan plot closely follows that of the simulated one. This validates our gamma compensation model and also provides a scaling factor which can be used at the time of decoding. Note

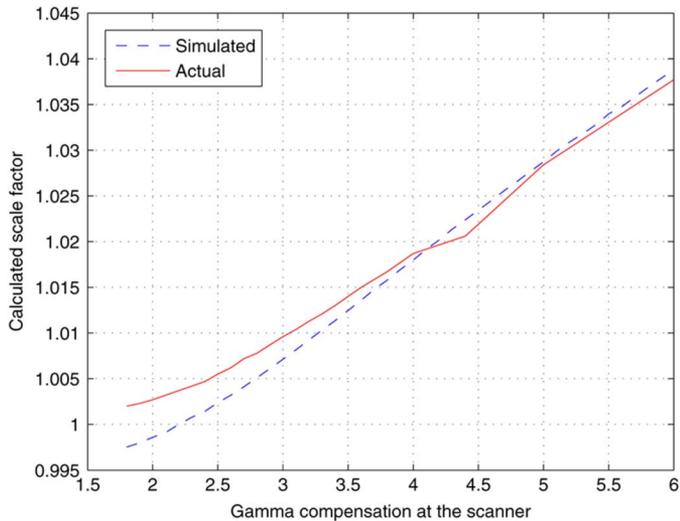


Fig. 11. Scale factor versus gamma compensation for actual print-scan and simulated processes.

that, in practice, we need not be precise in determining the scaling factor. Due to the error-correction capability (ECC) of the decoder, any scaling factor that is close to this calculated value works.

### C. Decoding

Once the image is automatically derotated and the gamma compensation is corrected using the above algorithms, it is then used to demodulate and decode the embedded information. Readers are referred to our prior work [23] for a detailed discussion on decoding for the employed coding framework. Here, we just provide an overview.

The receiver takes the DFT of the image coefficients and scans the coefficients in the same order as the encoder. It employs the same threshold criteria as the encoder to estimate the locations where data have been embedded. Hard-decision decoding of the embedded channel symbols is performed. This is because it is difficult to quantify the statistics of the print-scan attack. For those coefficients that do not pass the threshold test, an erasure is passed to the channel decoder. Finally, the sum-product algorithm [26] is used to decode the hidden information bits leading to error-free recovery of the hidden data despite the strong attacks. The use of powerful channel codes provides robustness to the embedded data against a variety of other attacks as well.

## VI. RESULTS

We now present the performance of our embedding schemes in this section. Note that the setup for evaluating the hiding techniques remains the same as that in our experimental setting (Section III). Images with hidden data are printed and the digital scanned image is fed to a receiver that decodes the hidden data after undoing the rotation using the automated algorithm of Section V-A. We have evaluated the hiding schemes for several images and for many different printers. Note that when scanning at higher resolutions (300 samples per inch or more), the choice

TABLE II  
NUMBER OF INFORMATION BITS HIDDEN ALONG WITH RA CODE PARAMETERS USED FOR VARIOUS  $512 \times 512$  IMAGES FOR THE PRINT-SCAN ATTACK. THE IMAGES WITH A LISTED NUMBER OF HIDDEN BITS ALSO SURVIVE ATTACKS SUCH AS  $3 \times 3$  GAUSSIAN FILTERING,  $4 \times 4$  MEDIAN FILTERING, HEAVY JPEG COMPRESSION ( $QF = 10$ ), 17 ROW AND FIVE COLUMNS REMOVAL, AND ASPECT RATIO CHANGE (BY  $0.8 \times 1.00$ )

Image	# of bits hidden	RA code rate ( $1/q$ )	# of coeff. in band
Peppers	250	1/4	870
Baboon	475	1/6	2450
Bridge	250	1/7	1560
Man	500	1/5	2450
Couple	300	1/6	1560

TABLE III  
COMPARISON OF NUMBER OF INFORMATION BITS HIDDEN IN VARIOUS  $512 \times 512$  IMAGES IN TWO SCENARIOS: (I) AUTOMATIC DEROTATION AT THE DECODER, AND (II) CAREFUL MANUAL PLACING OF THE IMAGE PRINTOUT ON THE FLATBED SCANNER

Image	Number of bits hidden	
	Auto. derotation	Manual placing
Peppers	250	225
Baboon	475	350
Bridge	250	200
Man	500	400
Couple	300	275

of scanner does not make much difference in the performance of the embedding schemes. In the presented results, we assume that the original image size is known to the decoder.<sup>4</sup>

In Sections VI-A and VI-C, we present the maximum number of bits that can be hidden and recovered perfectly for five selected sample images, for each hiding scheme. These images were chosen based on varying detail and texture content so as to study their embedding capacities. In the experiments, the number of bits embedded into the images is increased (in steps of 25 bits), until we fail to recover the hidden data. The bits reported in Tables II, III, and V are the number of bits that can be embedded in that particular image with perfect recovery after scanning. We next present a prototype large-scale system (Section VI-D) in which same parameters are used to embed in a set of 30 images. We also present results of psycho-visual experiments here. We now start the presentation of the results with the SELF hiding scheme.

### A. Surviving Print-Scan With Automatic Derotation

Fig. 12 shows three sample images at various stages of embedding, attack, and decoding. The embedded bits can be recovered from the images after they are printed and scanned, even when the images get rotated during the scanning process. For example, Fig. 12(a) and (b) shows the original man image and the composite image with 500 bits embedded. Fig. 12(c) shows the printed-and-scanned image which was rotated during the scanning process. Fig. 12(d) shows the automatically derotated image (using the algorithm proposed in Section V-A). Fig. 12(e) shows the image after the background is automatically cropped. Similarly, Fig. 12(f)–(j) shows the baboon image

<sup>4</sup>Note that when the original image size is not known, it is possible to use the printer halftoning pattern to compute the original scale of the image (when the printer’s lpi is known to the decoder). In general, a particular system can be designed based on the information assumed to be available to the decoder.

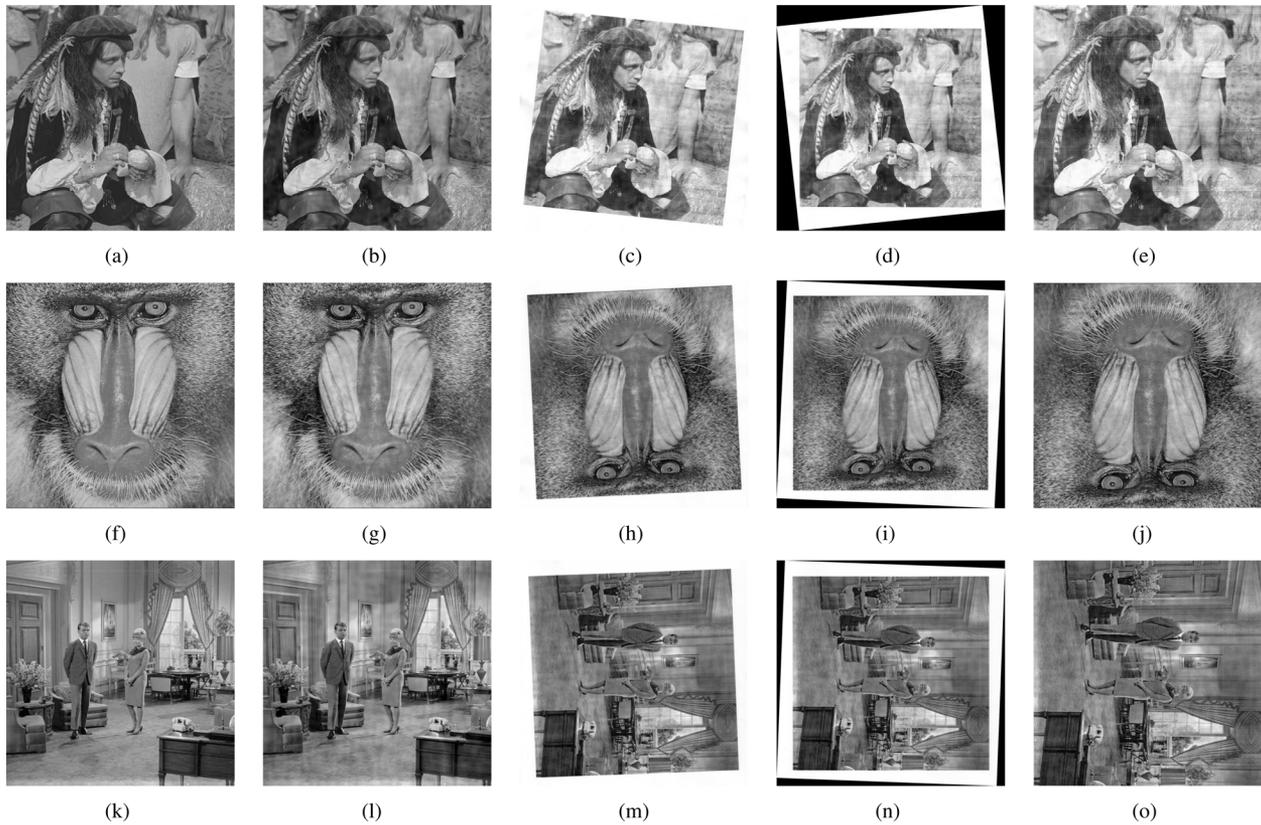


Fig. 12. Images at various stages of embedding, attack, and decoding. All of the embedded bits have been recovered successfully at the decoder. Decoding is successful even when the derotated image is inverted (second example) because of the symmetry of the embedding process. When rotation is  $90^\circ$  or  $270^\circ$ , the decoder conducts another trial with a  $90^\circ$  rotation. (a) Original  $512 \times 512$  man image. (b) Image with 500 bits hidden. (c) Printed and scanned image. (d) Automatically derotated image. (e) The derotated image cropped automatically. (f) Original  $512 \times 512$  baboon image. (g) Image with 475 bits hidden. (h) Printed and scanned image. (i) Automatically derotated image. (j) The derotated image cropped automatically. (k) Original  $512 \times 512$  couple image. (l) Image with 300 bits hidden. (m) Printed and scanned image. (n) Automatically derotated image. (o) The derotated image cropped automatically.

example, and Fig. 12(k)–(o) shows the intermediate-stage images for the couple image.

Table II shows the number of information bits hidden for various  $512 \times 512$  images along with the RA code rate and number of candidate embedding coefficients. The listed number of bits was perfectly recovered after the images were printed and scanned with varying degrees of rotation.

Table III compares the number of information bits hidden in various  $512 \times 512$  images with automatic derotation at the decoder and with careful manual placing of the image on the flatbed of the scanner to avoid rotation. It can be seen that more information bits can be hidden when automatic derotation is performed at the decoder compared to careful manual placing without automatic derotation. It shows that the automatic derotation outperforms the best human effort at preventing rotation.

### B. Other Attacks

The images with data hidden using SELF hiding scheme also survive several other attacks included in *Stirmark* [13] (e.g., Gaussian or median filtering, rows and/or columns removal, heavy JPEG compression, and aspect ratio change). The number of bits listed in Tables II and III survives these attacks as well. In Table IV, we show the percentage of errors encountered against various attacks for an uncoded transmission. This gives us an idea of the amount of protection needed via ECCs to deal with

TABLE IV  
PERFORMANCE OF THE PROPOSED SELF HIDING SCHEME  
AGAINST VARIOUS ATTACKS

Images	# bits hidden	Attacks: Overall error percentage					
		Print-Scan	JPEG compr. QF=10	$3 \times 3$ Gaussian filter	$4 \times 4$ Median filter	17 rows 5 cols removed	Aspect ratio change $0.8 \times 1.0$
Barbara	367	7.63%	1.77%	0%	2.72%	2.45%	0.27%
Man	1076	15.75%	8.59%	0.09%	3.86%	5.62%	0.09%
Couple	364	10.03 %	4.81%	0%	1.64%	1.24%	0.55 %

those errors. It can be seen that the print-scan process is most severe among all of the attacks. Hence, a system with sufficient redundancy to survive the print-scan process would also work against all other attacks. This is consistent with our observation that the images that are designed to survive the print-scan process using the SELF hiding scheme survive all of the attacks listed in Table IV. Note that these attacks were carried out independently of the print-scan process.

It should be noted that much less data can be hidden against the *Stirmark* random bending attack. For example, 73 bits are hidden in the Peppers image (without the channel coding) and received with 20% error. Note that this performance may still be good for watermarking applications, where the watermark sequence is known to the decoder and can be correlated with the hidden data to detect the watermark. So far, we have discussed the performance of the SELF embedding scheme (for

TABLE V  
DQIM EMBEDDING IN PHASE: NUMBER OF INFORMATION BITS HIDDEN  
ALONG WITH RA CODE PARAMETERS USED FOR VARIOUS  $512 \times 512$   
IMAGES FOR THE PRINT-SCAN ATTACK

Image	# of bits hidden	RA code rate (1/q)	# of coeff. in band
Peppers	125	1/5	576
Baboon	275	1/6	1444
Bridge	250	1/6	1444
Man	225	1/7	1444
Couple	150	1/6	784

hiding in magnitude spectrum). We now move on to the DQIM hiding method, which embeds data into the phase spectrum of the images.

### C. DQIM Hiding in Phase

For our DQIM hiding in phase method, we are able to embed several hundred bits against the print-scan attack. Table V shows the number of information bits hidden for various  $512 \times 512$  images along with the RA code rate and number of candidate embedding coefficients. Here too, all of the embedded bits are recovered after the print-scan attack. The volume of embedding depends on the host image, which turns out to be less than that of the SELF hiding scheme for embedding in the magnitudes. This is especially true for images such as Peppers and Couple that have many smooth regions within, so that a smaller candidate embedding bands must be used in order to preserve the perceptual quality. Since DFT phase is known to have more information about the image than the magnitudes [27], it is that much more difficult to embed data in the phase spectrum without inducing much perceptual distortion.

### D. Prototype Large-Scale System

In the previous sections, we present results for specific images with data hidden using parameters tuned for those images. The goal was to understand the embedding capacity of the algorithms for the images. We now present a prototype system, in which same parameters are used to embed data into a set of 30 images. The SELF hiding scheme is used to hide 100 bits with rate  $1/4$  RA code, and 380 coefficients in the candidate embedding band. In the experiments conducted, we could recover the hidden data from all of the images after printing and scanning.

In order to assess the perceptual quality of the hidden images, we conducted psycho-visual experiments. Distortion metrics, such as PSNR, have been used in the literature, but are known to be ineffective. In the absence of good distortion metrics, the only way to ascertain the perceptual quality of images is through psycho-visual experiments. Thirty-five subjects were shown a slide show of 30 original images used in the experiment. After that, they were asked to distinguish between original and doctored images. Half of the images had 100 bits embedded and others were originals. The results are as follows. On average, 62.10% of composite images were missed (identified as innocent), and 27.74% were false positives (original images identified as doctored). The standard deviation for the percentage missed and false alarm was 19.20 and 16.70, respectively. It can be seen that overall, the subjects made mistakes almost 90% of time (sum of missed and false alarm is 89.84%). Though this is

not a perfect result, we believe that this is an excellent performance.<sup>5</sup>

Before we leave this section, we would like to caution the readers regarding the results presented here. Not all images can have enough embedding capacity to hide 100 bits in a perceptually transparent manner. For some smooth images, much fewer bits could be hidden without inducing noticeable perceptual distortion.

## VII. CONCLUSION

We have successfully demonstrated print-scan resilient data hiding methods with potential applications, such as document authentication and image copyright protection. The robustness of the methods is based on three key components of our approach: choice of embedding strategy based on analytical and experimental modeling of the print-scan process, the use of powerful turbo-like channel codes, and automated algorithms for derotation and correcting gamma compensation at the receiver. In analytical modeling, we get around the complexity involved by dividing the print-scan operation into simpler subprocesses and identifying the bottlenecks, which are then studied in detail.

There is still much left for future investigation. One could focus on some specific printers and scanners, and analyze the nonlinear transformations in more detail so as to design hiding schemes with higher capacities. Another interesting avenue for future work is to leverage the inverse halftoning literature for reducing the effect of colored noise. This way, we could possibly improve the embedding capacity by using the mid (or high) frequency coefficients along with the low-frequency ones for hiding.

## ACKNOWLEDGMENT

We thank the anonymous reviewers for their careful reading and insightful comments, which greatly helped improve the clarity of the paper. We would also like to thank J. Byun for helpful discussions on derotation techniques.

## REFERENCES

- [1] C. Y. Lin and S. F. Chang, "Distortion modeling and invariant extraction for digital image print-and-scan process," presented at the Int. Symp. Multimedia Information Processing Dec. 1999.
- [2] J. K. O. Ruanaidh and T. Pun, "Rotation, scale and translation invariant spread spectrum digital image watermarking," in *Signal Process.*, May 1998, vol. 66, no. 3, pp. 303–317.
- [3] V. Solachidis and I. Pitas, "Circularly symmetric watermark embedding in 2-D DFT domain," *IEEE Trans. Image Process.*, vol. 10, no. 11, pp. 1741–1753, Nov. 2001.
- [4] P. Bas, J.-M. Chassery, and B. Macq, "Geometrically invariant watermarking using feature points," *IEEE Trans. Image Process.*, vol. 11, no. 9, pp. 1014–1028, Sep. 2002.
- [5] J. Rosen and B. Javidi, "Hidden images in halftone pictures," *Appl. Opt.*, vol. 40, no. 20, pp. 3346–3353, Jul. 2001.
- [6] M. S. Fu and O. C. Au, "Data hiding watermarking in halftone images," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 477–484, Apr. 2002.
- [7] S. V. Voloshynovskiy, O. Koval, F. Deguillaume, and T. Pun, "Visual communications with side information via distributed printing channels: Extended multimedia and security perspectives," in *Proc. SPIE: Security, Steganography, and Watermarking of Multimedia Contents VI*, San Jose, CA, Jan. 2004, pp. 428–445.

<sup>5</sup>The subjects tended to view the images very carefully before deciding. Also, since a slide-show of the original images was shown to the users, the test is not completely blind. This is why we believe that users making mistakes 90% of time is very good.

- [8] A. K. Mikkilineni, G. N. Ali, P.-J. Chiang, G. T. C. Chiu, J. P. Allebach, and E. J. Delp, "Signature-embedding in printed documents for security and forensic applications," in *Proc. SPIE: Security, Steganography, and Watermarking of Multimedia Contents VI*, San Jose, CA, Jan. 2004, pp. 455–66.
- [9] K. Solanki, U. Madhow, B. S. Manjunath, and S. Chandrasekaran, "Estimating and undoing rotation for print-scan resilient data hiding," presented at the ICIP, Singapore, Oct. 2004.
- [10] K. Solanki, U. Madhow, B. S. Manjunath, and S. Chandrasekaran, "Modeling the print-scan process for resilient data hiding," in *Proc. SPIE Security, Steganography, and Watermarking of Multimedia Contents VII*, Mar. 2005, vol. 5681, pp. 418–429.
- [11] K. Solanki, O. Dabeer, U. Madhow, B. S. Manjunath, and S. Chandrasekaran, "Robust image-adaptive data hiding: Modeling, source coding and channel coding," in *Proc. 42nd Annu. Allerton Conf. Communications, Control, and Computing*, Oct. 2003.
- [12] B. Chen and G. W. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE Trans. Inf. Theory*, vol. 47, no. 4, pp. 1423–1443, May 2001.
- [13] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Attacks on copyright marking systems," in *Proc. Workshop Information Hiding*, 1998, pp. 219–239.
- [14] D. Kacker, T. Camis, and J. P. Allebach, "Electrophotographic process embedded in direct binary search," *IEEE Trans. Image Process.*, vol. 11, no. 3, pp. 243–257, Mar. 2002.
- [15] A. Vongkumhae, J. Yi, and R. B. Wells, "A printer model using signal processing techniques," *IEEE Trans. Image Process.*, vol. 12, no. 7, pp. 776–783, Jul. 2003.
- [16] G. Sharma, "Targetless scanner color calibration," *J. Imaging Sci. Technol.*, vol. 44, no. 4, pp. 301–307, Jul./Aug. 2000.
- [17] E. H. B. Smith, "Characterization of image degradation caused by scanning," *Pattern Recognit. Lett.*, vol. 19, no. 13, pp. 1191–1197, 1998.
- [18] R. Ulichney, *Digital Halftoning*. Cambridge, MA: MIT Press, 1987.
- [19] M. Mese and P. P. Vaidyanathan, "Look-up table (LUT) method for inverse halftoning," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1566–1578, Oct. 2001.
- [20] X. Zixiang, M. T. Orchard, and K. Ramchandran, "Inverse halftoning using wavelets," *IEEE Trans. Image Process.*, vol. 8, no. 10, pp. 1479–1483, Oct. 1999.
- [21] P. Moulin and A. Briassouli, "A stochastic QIM algorithm for robust, undetectable image watermarking," presented at the ICIP, Singapore, Oct. 2004.
- [22] T. D. Kite, B. L. Evans, and A. C. Bovik, "Modeling and quality assessment of halftoning by error diffusion," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 909–922, May 2000.
- [23] K. Solanki, N. Jacobsen, U. Madhow, B. S. Manjunath, and S. Chandrasekaran, "Robust image-adaptive data hiding based on erasure and error correction," *IEEE Trans. Image Process.*, vol. 13, no. 12, pp. 1627–1639, Dec. 2004.
- [24] N. Jacobsen, K. Solanki, U. Madhow, B. S. Manjunath, and S. Chandrasekaran, "Image adaptive high volume data hiding based on scalar quantization," in *Proc. IEEE Military Comm. Conf.*, Anaheim, CA, Oct. 2002.
- [25] D. Divsalar, H. Jin, and R. J. McEliece, "Coding theorems for turbo-like codes," in *Proc. 36th Allerton Conf. Communications, Control, and Computing*, Sep. 1998, pp. 201–210.
- [26] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.
- [27] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Reading, MA: Addison-Wesley, 1992.



**Kaushal Solanki** (M'00) received the B.E. degree in electronics engineering from the National Institute of Technology (NIT), Surat, India (formerly REC-Surat), in 2000 and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of California, Santa Barbara (UCSB), in 2001 and 2005, respectively. His research interests include various aspects of information hiding, multimedia security, and computer vision.

Dr. Solanki was awarded a dissertation fellowship from the Department of Electrical and Computer Engineering, UCSB, in 2005. He is the recipient of the IBM student paper award of the 2004 IEEE International Conference in Image Processing (ICIP) for his work on print-scan resilient data hiding.



**Upamanyu Madhow** (F'05), received the B.Sc. degree in electrical engineering from the Indian Institute of Technology, Kanpur, in 1985, and the M.S. and Ph.D. degrees in electrical engineering from the University of Illinois, Urbana-Champaign, in 1987 and 1990, respectively.

From 1990 to 1991, he was a Visiting Assistant Professor at the University of Illinois. From 1991 to 1994, he was a Research Scientist with Bell Communications Research, Morristown, NJ. From 1994 to 1999, he was on the Faculty of the Department of Electrical and Computer Engineering at the University of Illinois. Since 1999, he has been with the Department of Electrical and Computer Engineering at the University of California, Santa Barbara, where he is currently a Professor. His research interests are in communication systems and networking, with a current emphasis on wireless communication, sensor networks, and multimedia security.

Dr. Madhow is a recipient of the National Science Foundation (NSF) CAREER award. He was an Associate Editor for Spread Spectrum of the IEEE TRANSACTIONS ON COMMUNICATIONS, and as Associate Editor for Detection and Estimation of the IEEE TRANSACTIONS ON INFORMATION THEORY and Associate Editor of the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY.



**B. S. Manjunath** (F'05) received the B.E. degree in electronics (Hons.) from Bangalore University, Bangalore, India, in 1985, and the M.E. degree (Hons.) in systems science and automation from the Indian Institute of Science, Bangalore, in 1987, and the Ph.D. degree in electrical engineering from the University of Southern California, Los Angeles, in 1991.

Currently, he is a Professor of Electrical Computer Engineering and Director of the Center for Bio-Image Informatics at the University of California, Santa Barbara. His research interests

include image processing, data hiding, multimedia databases, and bioimage informatics.

Dr. Manjunath was a Recipient of the National Merit Scholarship (1978–1985) and was awarded the university gold medal for the best graduating student in electronics engineering from Bangalore University in 1985. He is a co-editor of the book on *Introduction to MPEG-7* (Wiley, 2002). He was an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING and is currently an Associate Editor of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and IEEE TRANSACTIONS ON MULTIMEDIA.



**Shiv Chandrasekaran**, received the M.Sc. (Hons.) degree in physics from the Birla Institute of Technology and Science (BITS), Pilani, India, in 1987, and the Ph.D. degree in computer science from Yale University, New Haven, CT, in 1994.

He was a Visiting Instructor at North Carolina State University, Raleigh, in the Mathematics Department, before joining the Electrical and Computer Engineering Department, University of California, Santa Barbara, where he is currently a Professor. His research interests are in computational mathematics.



**Ibrahim El-Khalil** (S'01) received the B.E. degree in computer and communications engineering from the American University of Beirut (AUB), Beirut, Lebanon, in 2004, and the M.S. degree in electrical engineering from the University of California, Santa Barbara, in 2006, where he is currently pursuing the Ph.D. degree at the Vision Research Laboratory. His research interests include multimedia data hiding and computer vision.

Dr. El-Khalil received the Merit Scholarship Award from AUB in 2000.