Data Embedding Using Phase Dispersion

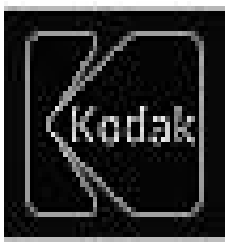
Chris Honsinger and Majid Rabbani
Imaging Science Division
Eastman Kodak Company
Rochester, NY USA

Abstract

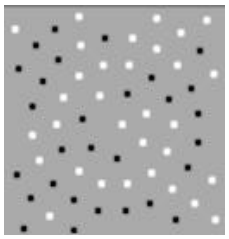
A method of data embedding based on the convolution of message data with a random phase carrier is presented. The theory behind this method is reviewed and it is shown that the technique can be used to hide both pictorial and non-pictorial data. The details of the procedures used for carrier design, message template optimization, message extraction optimization, block synchronization, and rotation and scale correction are discussed. Finally, the algorithm's benchmark results using Stirmark are presented.

1. Introduction

An important aspect of our technique is that it can be used to embed either a grayscale iconic image or binary data. Examples of iconic images are trademarks, corporate logos or other arbitrary small images. Since the algorithm performance generally decreases as the message energy increases, it is preferable to use the edge maps of an icon as a message as shown in Figure 1a. When embedding binary data, the one and zero bits are represented by positive and negative delta functions that are placed in predefined and unique locations across the message image (referred to as the *message template*) as shown in Figure 1b. Common examples of binary data are the 32-bit representations of URL's or copyright notices.



(a) Example of a 128x128 binary iconic message



(b) Example of a 64-bit binary message comprising of 1's (white) and 0's (black)

Figure 1. Examples of iconic and binary message images

The following notation is adopted throughout this paper. The original image is represented by the two dimensional array, $I(x,y)$ the watermarked image (the image containing the embedded data) by $I'(x,y)$, the message image by $M(x,y)$, the carrier image by $C(x,y)$, and the message template by $T(x,y)$. With these definitions, the message embedding process is defined by the following equation:

$$I'(x,y) = \alpha(M(x,y) * C(x,y)) + I(x,y) \qquad \text{Eq. (1)}$$

where the symbol, $*$, represents cyclic convolution and α is an arbitrary constant chosen to make the embedded message simultaneously invisible and robust to common processing. From Eq. (1) it is clear

that there are no restrictions on the message image and its pixel values can be either grayscale or binary. It is well known that space domain cyclic convolution of two signals is equivalent to multiplying their magnitudes while adding their phases in the Fourier domain. For a message that is a single delta function, the effect of convolution with a carrier with random Fourier phase and uniform magnitude is to distribute the energy of the delta function over space.

The basic extraction process is straightforward and consists of correlating the watermarked image, $I'(x,y)$, with the same carrier image, $C(x,y)$, used to embed the image. Denoting the extracted message by $M'(x,y)$, and using Eq. (1):

$$M'(x,y) = I'(x,y) \otimes C(x,y) = \alpha(M(x,y) * C(x,y)) \otimes C(x,y) + I(x,y) \otimes C(x,y), \quad \text{Eq. (2)}$$

where the symbol $\otimes$ represents cyclic correlation. The cyclic correlation of two signals in the space domain is equivalent to multiplying their magnitudes while subtracting their phases in the Fourier domain. Thus, for a carrier with uniform Fourier amplitude, $C(x,y) \otimes C(x,y) = \delta(x,y)$, where $\delta(x,y)$ is the Dirac delta function. Noting that the operations of convolution and correlation commute, Eq. (2) reduces to:

$$M'(x,y) = \alpha M(x,y) * (C(x,y) \otimes C(x,y)) + I(x,y) \otimes C(x,y) = \alpha M(x,y) + I(x,y) \otimes C(x,y). \quad \text{Eq. (3)}$$

That is, the process of correlating the watermarked image with the carrier image results in a scaled version of the message image plus some noise due to the cross correlation of the original image with the carrier. Ideally, the cross correlation of the carrier image with itself should be a delta function, while the cross correlation of the carrier with the original image should be zero.

Experimentally, we have found that tiling the original image and embedding the same message in each tile independently improves the robustness of the algorithm. For the results reported in this paper, the message image is assumed to be 128x128 and the original image is tiled into 128x128 blocks and the message is repeatedly embedded in each tile. During the extraction process, the 128x128 tiles are aligned and summed before applying the cross correlation. The basic operations of message embedding and extraction are depicted in Figs (2), and (3), respectively.

For imaging applications with severe quality loss, such as small images printed using ink-jet printers on plain paper, a weighting factor that depends on the estimated signal to noise ratio is calculated and applied to each extracted message element before summation.

2. Carrier Design Considerations

The design of the carrier is a tradeoff between the visual transparency of the embedded message, the optimum extracted signal quality, and robustness to image processing tasks. For a general carrier, according to Eq. (3), the extracted message is given by:

$$M'(x,y) = \alpha M(x,y) * p(x,y) + I(x,y) \otimes C(x,y), \quad \text{Eq. (4)}$$

where $p(x,y)$ denotes the autocorrelation function of the carrier and acts as a point spread function (psf) to filter the original message. On one hand, in order to improve the extracted signal quality, the function $p(x,y)$ should be as close to a delta function as possible. This can be accomplished by setting the Fourier magnitude of the carrier image to a constant at all frequencies. On the other hand, the contrast sensitivity function (CSF) of the human visual system falls off rapidly with increasing spatial frequency. Hence, for visual transparency, most of the carrier energy should be concentrated in high frequencies. Finally, another factor to be considered is robustness to both friendly and malicious attacks. Considering the fact

that the power spectrum of typical imagery falls off as the inverse of the square of spatial frequency, concentration of the carrier energy in high frequencies creates very little overlap between the image and the embedded message in the frequency domain. This makes the embedding scheme vulnerable to simple frequency domain processing such as low-pass filtering.

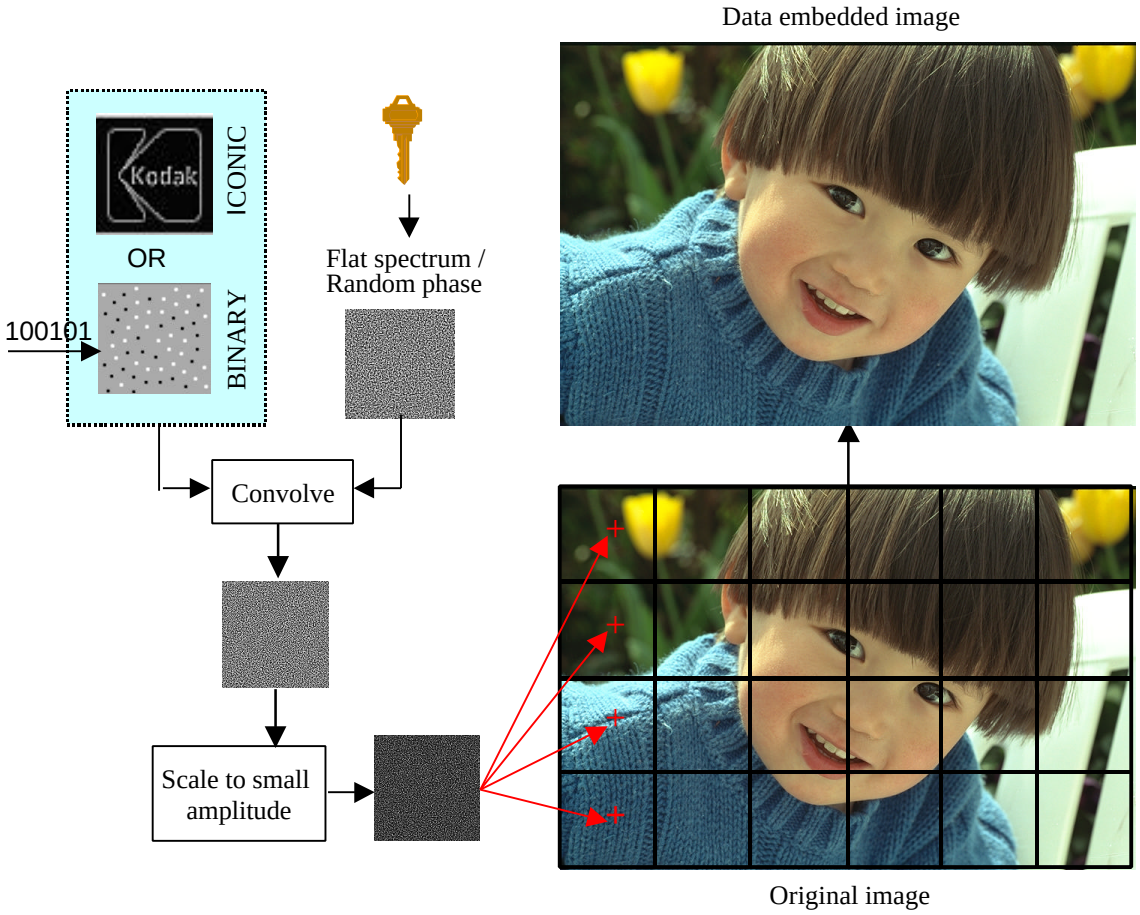


Figure 2. The message embedding process

Our carrier has been designed with all these considerations in mind. More specifically, the phase of the carrier is generated using a psuedo-random number generator with a user-specified key. The magnitude is set to zero at zero frequency (DC value) and is gradually increased with increasing spatial frequency up to about 1/5 of Nyquist frequency. As shown in Figure 4, for frequencies greater than 1/5 of Nyquist, the carrier envelope is derived from the Contrast Sensitivity Function[3] (CSF) data. The CSF provides a measure of the sensitivity of the average observer to changes in contrast at a given spatial frequency. For a prescribed set of viewing conditions, the reciprocal of the CSF can be used to determine the carrier magnitude needed at a given frequency to bring the embedded signal just below the threshold of delectability by an average viewer.

The CSF models generally take into account the observer viewing distance, background noise, device dot density, and color component wavelength among other factors. Use of these CSF parameters can be particularly advantageous when optimizing the embedding algorithm for a specific application. Figure 3 shows a two-dimensional plot of the carrier magnitude in the frequency domain (dark implies a small magnitude) for three different carrier designs based on CSF data. All three examples use the same set of viewing conditions except for the viewing distance that has been set to 4, 8, and 12 inches, respectively.

This demonstrates the wide variation that is possible in carrier magnitudes based on varying the viewing distance alone.

The effect of a carrier with a nonuniform Fourier magnitude on the autocorrelation function $p(x,y)$ is to produce small sidelobes around the peak occurring at zero displacement. In our experience, the impact of destructive processes such as compression, error diffusion, printing and scanning on the quality of the recovered message is far greater that the loss in bandwidth due to the non-ideal correlation function. In fact, as long as the sidelobes are confined to half of the minimum separation distance between the delta functions in the message template, the sidelobe interference may be considered to be negligible.

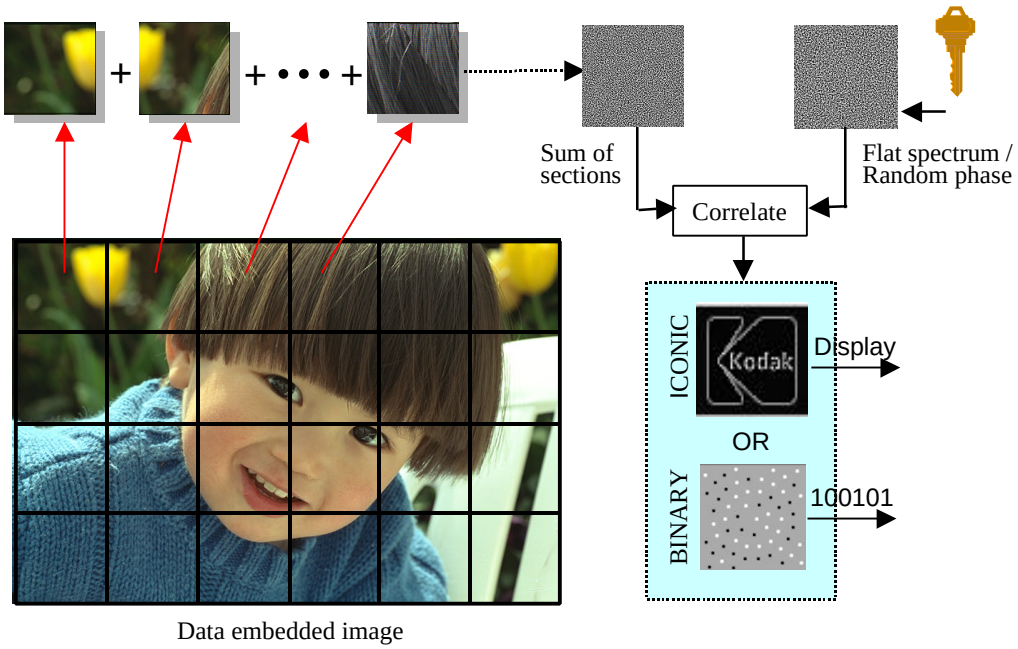


Figure 3. The basic message extraction process

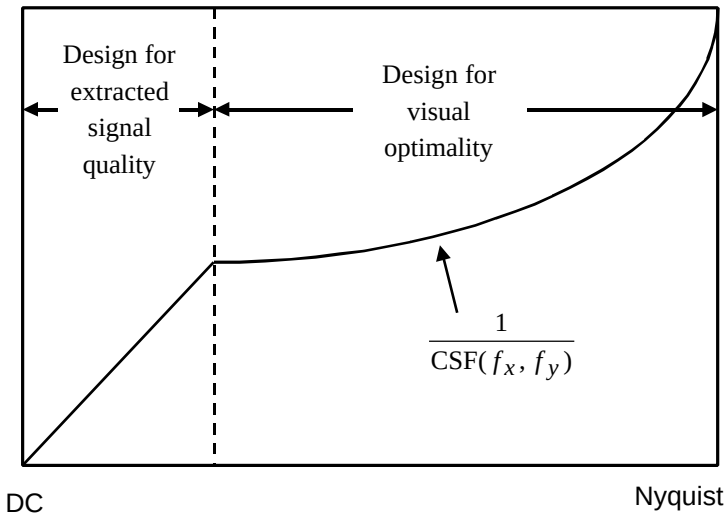
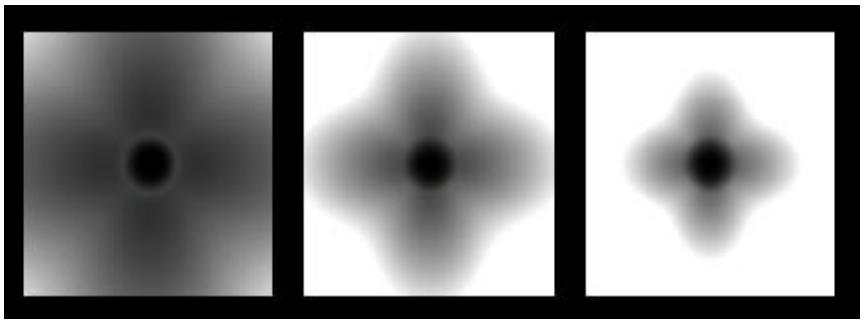


Figure 4. Plot of the carrier magnitude as a function of spatial frequency.



4 inches 8 inches 12 inches

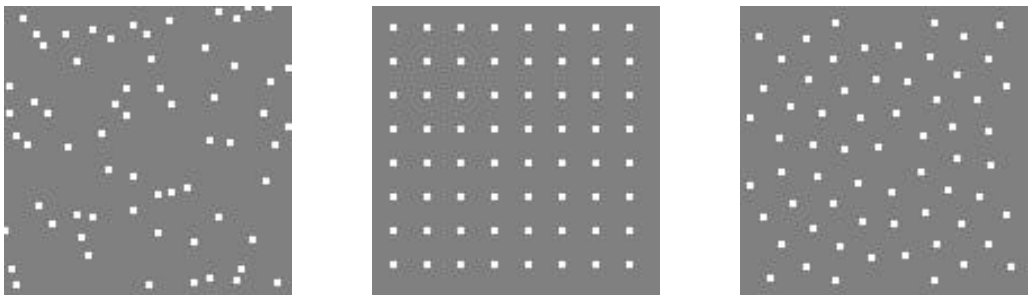
Threshold Viewing Distance →

Figure 5. Variation of Fourier magnitude of the carrier as a function of viewing distance (Device dpi=300)

3. Message Template Design

As mentioned earlier, the process of embedding binary information consists of representing the one and zero bits by positive and negative delta functions that are placed in predefined and unique locations within the message image. The message template, $T(x,y)$, is the image resulting from placing a positive delta function at every message location. Figure 6 contains three examples of different message templates that contain a payload of 64 bits.

The design of an optimal message template is guided by two requirements. The first one is maximizing the quality of the extracted signal, which is achieved by placing the message locations maximally apart to minimize the interference from the sidelobes of the carrier autocorrelation function. For example, the message template shown in Fig. (6a) is a poor choice for this purpose because some of the message locations are too close to one another. The second requirement is that the embedded message be recoverable from a cropped version of the watermarked image.



(a) (b) (c)

Figure 6. Examples of three different message template for a payload of 64 bits, (a) random lattice, (b) rectangular lattice, (c) circular lattice with random angular displacement

Consider a case where the watermarked image has been cropped such that the tiles in the cropped image have their origin displaced by a distance $(\Delta x, \Delta y)$ with respect to the tiles in the original image. From Eq. (2) it can be easily shown that the extracted message from the cropped image, $M''(x,y)$, is a cyclically shifted version of the extracted message from the uncropped image, $M'(x,y)$, i.e.,:

$$M''(x, y) = M'(x - \Delta x, y - \Delta y). \quad \text{Eq. (5)}$$

It should be noted that although the original message is not known at the time of extraction, the message template is assumed to be known. Hence, by insuring that all the cyclic shifts of the message template are unique, the amount of the shift can be unambiguously determined. In mathematical terms, the message template should ideally be designed so that its autocorrelation is a delta function. For a message template designed according to this criterion,

$$T(x - \Delta x, y - \Delta y) * T(x, y) = \delta(\Delta x, \Delta y), \quad \text{Eq. (6)}$$

and the amount of displacement (Δx, Δy) can be found by correlating the extracted message template with the original message template and recording the location of the peak. For example, the message template in Fig. 6b that uses a regular 2-D lattice is a poor choice for this purpose since its autocorrelation has multiple peaks spaced at regular intervals. In practice, it is impossible to make the autocorrelation of the message template a delta function as sidelobes will always be present. Instead, the message template is designed so that the amplitude of its maximum sidelobe is minimized. Since for a given data capacity (payload) the design of the message template is done offline and only once, computationally intensive optimization techniques such as simulated annealing can be used to insure maximum separation and zero autocorrelation. For the results presented in Section 6, the message template chosen was similar to the one shown in Fig. (3c) except that it had a payload of 74 bits instead of 64 bits. It consisted of concentric circles with equal increments in radius and random angular displacement.

4. Rotation/Scale Detection And Correction

The ability to handle rotation and scale is a fundamental requirement of robust data embedding techniques. Almost all applications involving printing and scanning will result in some degree of scaling and rotation. Some published algorithms rely on an additional calibration signal to correct for rotation and scale that taxes the information capacity of the embedding system. Our approach, however, uses the autocorrelation of the watermarked image[4] and hence does not require a calibration signal. In addition, it can be applied to any embedding technique where the embedded image is repeated in tiles. It can also be implemented on a more local level to confront low order geometric warps.

Consider the autocorrelation function of a watermarked image that has not been rotated or scaled. At zero displacement, a very large peak is expected due to the image correlation with itself. However, since the embedded message is repeated at each tile, lower magnitude correlation peaks are also expected at regularly spaced horizontal and vertical intervals equal to the tile dimension. It can be shown that rotation and scale affect the relative position of these peaks in exactly the same way that they affect the image. An example is shown in Figs. (7) and (8). Since the energy of the original image is much larger than that of the embedded message, the autocorrelation of the original image can mask the detection of the periodic peaks. To alleviate this problem, the watermarked image is processed prior to the computation of the autocorrelation function. Furthermore, the resulting autocorrelation function is high-pass filtered to amplify the peak values.

The first procedure, referred to as moment normalization, sets the local mean of the watermarked image to zero and its local standard deviation to a target value σ_d. Denoting the pixel value at location (x,y) of the watermarked image by $\nu(x,y)$, and its local mean and local standard deviation (e.g., computed over a 3x3 or 5x5 neighborhood) by $\mu(x,y)$ and $\sigma(x,y)$, respectively, the pixel value is modified according to the relationship

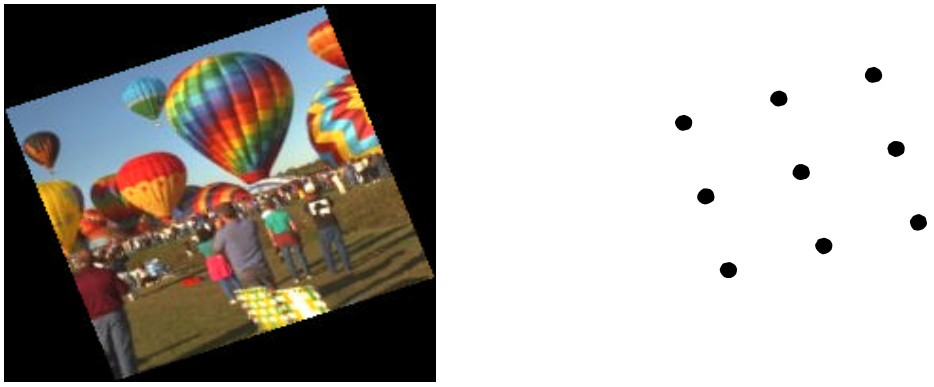
$$n'(x,y) = \frac{s_d}{s(x,y)}\left[n(x,y) - m(x,y)\right]. \qquad \text{Eq. (7)}$$

The effect of this operation is to remove the high amplitude, low frequency coherent noise. In flat areas when $\sigma(x,y) \to 0$, the modified pixel value is set to a value taken from a random number generator with a standard deviation equal to σ_d.

Next, the autocorrelation function of the processed image is computed. Finally, the resulting autocorrelation function is processed with a high-pass filter with a frequency response that linearly increases with increasing spatial frequency. A typical output resulting from these processing steps (see Figure 9) will contain peaks that need little further processing. In our implementation, the autocorrelation function is computed using FFT in frequency domain, so the high-pass filtering operation can be carried out at the same time.

(a) (b)

Fig. 7. (a) A watermarked image without rotation and scale; (b) autocorrelation of the image in (a)



(a) (b)

Fig. 8. (a) A watermarked image that has undergone rotation and scale; (b) autocorrelation of the image in (a)

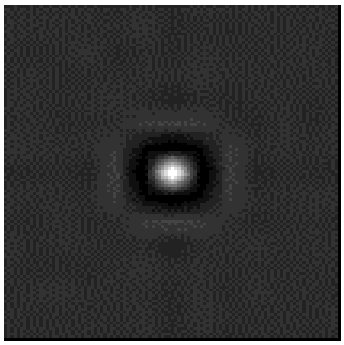


Figure 9: An actual autocorrelation peak (enlarged) demonstrating high signal to noise ratio

5. Stirmark Results

One way to assess the effectiveness of a data embedding algorithm is by using the StirMark[5,6] algorithm benchmark. Briefly, the Stirmark benchmark exposes an embedding algorithm to various types of adverse processing such as rotation/scale, JPEG compression, shearing, warping and other processes that a robust data embedding technique would be required to handle. More information may be obtained by visiting: *http://www.cl.cam.ac.uk/~fapp2/steganography*. The algorithm reported in this paper scored a value of 0.98 using StirMark 3.0.

The flowchart of the algorithm used in the StirMark benchmark is shown in Fig. (10). This flowchart has been designed to optimize the speed of extraction. A 128x128 uniform amplitude carrier (no CSF modification) adjusted to a PSNR of 38dB (as required by Stirmark) was used. The message template was designed based on concentric circles with random angular displacements and a payload of 74 bits, consisting of 64 message bits (as required by Stirmark) and 10 parity bits for error detection. Referring to Fig. (10), the basic extraction process (as shown in Fig. (3)) is executed first. This is referred to in the flowchart as the "Quick Scan". If all the bits are successfully extracted, the process is terminated. Successful extraction is defined as a case when the extracted parity bits detect no error and also the average signal to noise ratio in the extraction process exceeds a certain threshold. If the extraction is not successful, the image is tested for the presence of affine transformations as outlined in Section 4 (e.g., rotation/scale) and corrected accordingly. If the extraction is still unsuccessful, the algorithm performs weighted extraction in order to further improve the signal to noise ratio.

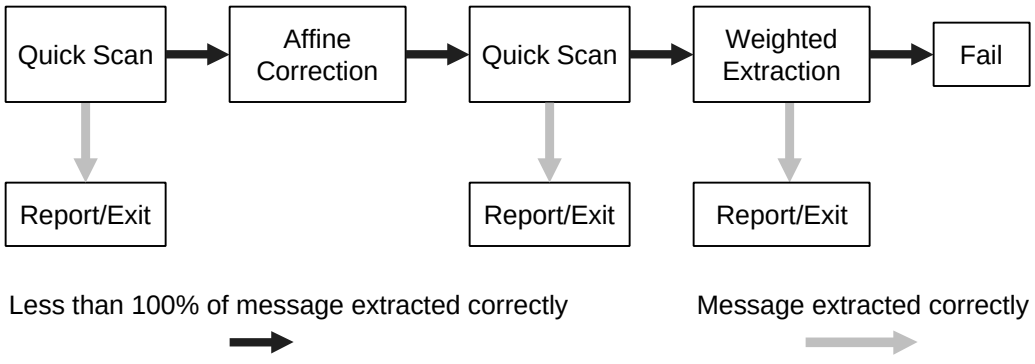


Figure 10: Flowchart of the algorithm used in the StirMark benchmark

References

1. S. J. Daly, J. R. Squilla, M. Denber, C. W. Honsinger, J. Hamilton, “Method for embedding digital information in an image”, U.S. Patent 5,859,920, 1999.
2. C. W. Honsinger, Majid Rabbani, “Method for generating an improved carrier for the data embedding problem”, U.S. Patent 6,044,156, 2000.
3. S. J. Daly, “Method and apparatus for hiding one image or pattern within another”, U.S. Patent 5,905,819, 1999.
4. C. W. Honsinger, S. J. Daly, “Method for detecting rotation and magnification in images”, U.S. Patent 5,835,639, 1998.
5. Fabien A. P. Petitcolas, Ross J. Anderson, Markus G. Kuhn. Attacks on copyright marking systems, in David Aucsmith (Ed), Information Hiding, Second International Workshop, IH'98, Portland, Oregon, U.S.A., April 15-17, 1998, Proceedings, LNCS 1525, Springer-Verlag, ISBN 3-540-65386-4, pp. 219-239.
6. Fabien A. P. Petitcolas and Ross J. Anderson, Evaluation of copyright marking systems. In proceedings of IEEE Multimedia Systems (ICMCS'99), vol. 1, pp. 574--579, 7--11 June 1999, Florence, Italy.