Cox's Algorithm: strengths and weaknesses with varying composition digital images

Emma Tierney, Thomas Newe, Tom Coffey

Data Communications Security Group,
Department of Electronic and Computer Engineering,
University of Limerick, Limerick
IRELAND

Abstract – **Starting in the early 90's, numerous digital watermarking schemes were introduced to combat the unauthorised copying of digital multimedia on the web. Initially techniques were proposed which manipulated data in the spatial domain to embed the secret information. However Cox et al. progressed to use a spread spectrum technique, which alters components in the frequency domain of an image to hide the secret data.**
In this paper Cox's second-generation watermarking algorithms for embedding and detection are discussed. A number of images are watermarked and then used with a benchmarking tool called Stirmark. This tool performs various attacks on the watermarked images. The results obtained show the strengths and weaknesses of using Cox's algorithm with varying composition images as a cover media.

Keywords – **Digital watermarking, benchmarking, robustness, evaluation.**

1. INTRODUCTION

With the introduction of the World Wide Web in the late 80's and then the first Internet connections being sold commercially in the early 90's, a new era in computer communications was born. With millions of users coming online every month, there came the inevitable misuse or unauthorised duplication of digital multimedia such as still images, video and audio files.

To address these issues related to intellectual property and copyright protection, digital watermarking emerged as a commercial application from the age-old art of steganography *(meaning hidden writing).*

Starting in the mid 90's, numerous watermarking systems were proposed such as [1], [2], [3], each one intending to embed various forms of secret information within digital media without degrading its quality in any way.

There are three main characteristics that should be present for a watermarking scheme to be effective, imperceptibility - robustness - capacity. *Imperceptibility*, meaning impossible or difficult to perceive by the mind or senses. In the case of watermarking, the marked image and the original should be perceptually indistinguishable. *Robustness*, meaning powerfully built or sturdy. After a marked image has been attacked with compression, geometric transformations, filtering etc., the mark should be intact and reliably decodable. *Capacity*, meaning the maximum amount that can be contained. Approximately 100 bits is the accepted size of a watermark. The least number of bits, the more robust the mark is. However this increased robustness requires a stronger embedding strength, which in turn increases the visual degradation of the image. The conflict between embedding strength and visual quality becomes apparent at this stage. These are the tradeoffs involved in designing an effective watermarking system.

The paper is structured as follows. Section II introduces the reader to the idea of second generation watermarking and the methods that Cox et al. used to realise their algorithm. Section III details the algorithm where both embedding and detection mechanisms are presented. Section IV describes the attacks used against Cox's technique using the Stirmark benchmarking tool. Section V presents the results and a discussion on the findings of the benchmarking attacks.

2. SECOND-GENERATION WATERMARKING

The watermark embedding algorithm of Cox et al. [4] took an idea from communications

theory and consequently viewed the frequency domain of the image as a communications channel, and the watermark as a signal to be transmitted through it. He used the spread spectrum technique to 'transmit' the watermark over the 'communication channel', the image.

Spread spectrum communication is the transmission of a narrowband signal over a much larger one so that the signal energy present is imperceptible. Equating this theory to how information is secretly transferred as mentioned above, the watermark is spread over each frequency bin of the transformed image and it is so minute that it goes undetected. Once the location and content of these signals are known at the verification point, they can be concentrated back into a single signal to represent the watermark. Destruction of the signal would require noise of high amplitude to be added to each frequency bin, which would visually degrade the image and render it useless.

The Discrete Cosine Transform is the method used in this algorithm to find the perceptually significant locations for watermark embedding.

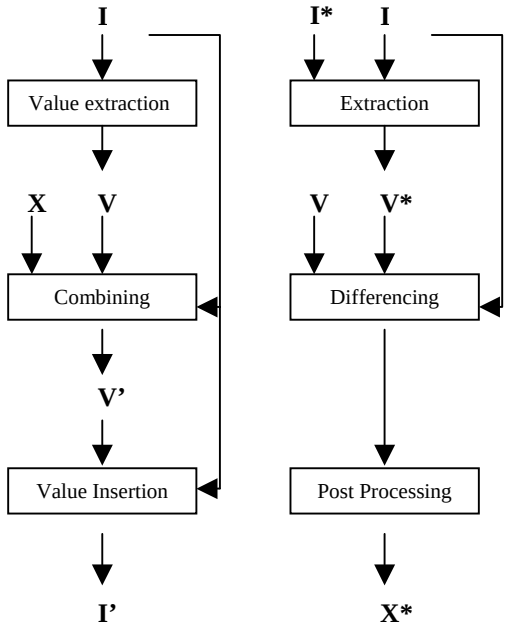
2.1 The Discrete Cosine Transform

The discrete cosine transform (DCT) helps separate the image into parts or spectral sub-bands, of differing importance with respect to the image's visual quality [5]. It transforms a signal or image from the spatial domain to the frequency domain, similar to the discrete Fourier transform. With an input image, A, the coefficients for the output "image," B, are:

$$B(k_1,k_2) = \sum_{i=0}^{N_1-1}\sum_{j=0}^{N_2-1} 4\cdot A(i,j)\cdot\cos\left[\frac{\pi\cdot k_1}{2\cdot N_1}\cdot(2\cdot i+1)\right]\cdot\cos\left[\frac{\pi\cdot k_2}{2\cdot N_2}\cdot(2\cdot j+1)\right]$$

The input image is N2 pixels wide by N1 pixels high; A(i,j) is the intensity of the pixel in row i and column j; B(k1,k2) is the DCT coefficient in row k1 and column k2 of the DCT matrix. All DCT multiplications are real. This lowers the number of required multiplications, as compared to the discrete Fourier transform. The DCT input is an 8 by 8 array of integers. This array contains each pixel's grey scale level; 8 bit pixels have levels from 0 to 255. The output array of DCT coefficients contains integers; these can range from -1024 to 1023. For most images, much of the signal energy lies at low frequencies; these appear in the upper left corner of the DCT. The lower right values represent higher frequencies, and are often small - small enough to be neglected with little visible distortion.

3. THE ALGORITHM



From each image **I**, extract a sequence of DCT coefficient values $\mathbf{V} = \mathbf{v_1}\ldots,\mathbf{v_n}$ into which to embed the watermark $\mathbf{X} = \mathbf{x_1},\ldots,\mathbf{x_n}$. The IDCT is then calculated from the resulting sequence **V'** and the newly watermarked image **I'** is created.

Image **I'** may then be attacked producing a new image **I***. Given **I** and **I***, a corrupted watermark **X*,** maybe extracted and compared to **X** for statistical significance. **X*** is found by first extracting a set of values $\mathbf{V^*} = \mathbf{v^*}_1,\ldots,\mathbf{v^*}_n$ from **I*** (using information about **I**) and then generating **X*** from **V*** and **V**. For testing purposes a correlation factor can indicate how alike **X** and **X*** actually are. A factor of one will indicate that the original watermark and the extracted watermark are exactly the same and 0 meaning they are totally different.

3.1 Embedding Strength

Different watermark embedding strengths can be applied to the image by exploiting knowledge of the masking phenomena in the human visual system. Perceptual masking is where certain regions of an image are obstructed by perceptually more prominent information in another part of the image.

A scaling parameter α can be used to determine the strength at which the watermark is

embedded to different parts of the image. The formulae for watermark insertion are as follows:

(1) $v'_i = v_i + \alpha x_i$
(2) $v_i = v_i(1 + \alpha x_i)$
(3) $v'_i = v_i(e^{\alpha x i})$

Equation 1 is always invertible, and equations 2 and 3 are invertible if v_i is not equal to 0.

Equation 1 is unsuitable when the values of **V** vary widely because, for instance, if $v_i = 10$ adding 100 could have drastic implications on the image where as adding 100 to 10^6 would go unnoticed. Equations 2 and 3 are less susceptible to such varying changes in scale. For example, (2) could be changed to $v_i = v_i(1 + \alpha_i x_i)$ where α_i can have multiple scaling values.

A single value for α may not be applicable for perturbing all values of v_i, since certain spectral components can exhibit more or less tolerance to modification that others. For example, a large α_i means that one can alter a v_i by a large factor and it can perceptually go unnoticed.

4. THE ATTACKS

Stirmark version 3.1.79[6] was used as a benchmarking tool to attack the watermarked images.

Stirmark was first published in 1997 as "a generic tool for simple robustness testing of image watermarking algorithms" [7]. It was developed by Fabien Petitcolas as part of his PhD and is used widely in the watermarking community as a benchmarking tool. Several versions were introduced in the years after, improving the original attacks and adding new attacks to its list.

Given a watermarked image as input, Stirmark will generate a number of modified images, which can be used to verify if a watermark can still be detected. Image alterations that can be found in the benchmarking suite are:

- Cropping
- Flip
- Rotation
- Rotation-Scale
- FMLR, sharpening, Gaussian filtering
- Random bending
- Linear transformations
- Aspect ratio
- Scale changes
- Line removal
- Colour reduction
- JPEG compression

A subset of these attacks was chosen and they are discussed in the paragraphs that follow.

Stirmark 3.1.79 applies distortions to the images using a command line interface. The goal of Stirmark is to prevent a watermark from being detected by attacking/degrading it, while at the same time attempting to maintain image quality.

4.1 JPEG Compression

JPEG (pronounced "jay-peg") is a standardized image compression mechanism [8]. It is designed for compressing either full-colour or grey-scale images of natural, real-world scenes. It is "lossy," meaning that the decompressed image isn't quite the same as the one you started with. The procedure basically gets rid of redundant pixels in the image to make it smaller. There are lossless image compression algorithms, but JPEG achieves much greater compression than is possible with lossless methods.

A useful property of JPEG is that by adjusting compression parameters one can vary the degree of lossiness. This means that the image-maker can trade off file size against output image quality.

Stirmark gives results for compression factors of 90% down to 10%, of the original images' size.

4.2 Median Filters

The median filter is a sliding-window spatial filter, which replaces the centre value in the window with the median of all the pixel values in the window. An example of median filtering of a single 3x3 window of values is shown below.

6	2	0
3	97	19
10	4	3

Unfiltered values

*	*	*
*	4	*
*	*	*

Median filtered

In order: 0, 2, 3, 3, **4**, 6, 10, 15, 97

Center value (previously 97) is replaced by the median of all nine values (4). This illustrates one of the celebrated features of the median filter: its ability to remove 'impulse' noise (outlying values, either high or low).

Testing Stirmarks' median filter option, the images were passed through a 2 x 2, 3 x 3 and 4 x 4 filter.

4.3 Convolution Filters

Convolution filters, also called user-defined filters, use a matrix of values around (and including) a kernel pixel to modify (or filter) an image according to a mathematical formula. Such a filter gives different weights to the individual matrix points. The grey value of each point is multiplied by the corresponding point in the matrix. The colour (or grey level) of the target point is determined by summing all these values for each pixel.

$$F = \left(\sum_{i=1}^{49} P_i C_i \right) \div D + B$$

Where F is the filtered value of the target pixel, P is a pixel in the grid, C is a coefficient in the matrix, D is the divisor and B is the Bias, 49 it being a 7 x 7 matrix.

Blurring or smoothing filers are known as low-pass filters. To implement this type of filter, you add a contribution from the neighbouring pixels and reduce the contribution from the pixel itself. You can modify the filter to blur more by increasing the number of pixels that are sampled in the matrix (e.g. use a 5x5 or 7x7 matrix rather than a 3x3 one) and by increasing the contribution for the pixels surrounding the central one.

A sharpening filter, also called a high-pass filter, emphasizes discontinuities in the pixel values by subtracting a contribution from the surrounding pixels from an increased central pixel. This means that when the surrounding pixels are dark, the central pixel is increased in brightness, which enhances the edges of an image.

The tables below represent the blurring filter matrix and the sharpening filter matrix used in the Stirmark test that is passed along the image to disrupt the edges.

1	2	1
2	4	2
1	2	1

Blurring matrix

0	-1	0
-1	5	-1
0	-1	0

Sharpening matrix

4.4 Geometric Distortions

Geometric transforms are used to distort images. These transforms modify the spatial relationships between pixels in an image. Geometric transforms are often called "rubber sheet transformations", because they are mainly viewed as the process of printing an image on a rubber sheet and then stretching this sheet in accordance to some predefined set of rules.

In digital image processing, a geometric transformation consists of two basic operations: (1) spatial transformations, which defines the rearrangement of the pixels in the image plane; and (2) grey-level interpolation, which deals with the assignment of grey levels to pixels in the transformed image.

4.4.1 Spatial Transformations

Lets presume that an image *f* with pixel co-ordinates (x,y) undergoes geometric distortion to produce an image *g* with co-ordinates (x',y'). This may be expressed as

$$x' = r(x,y)$$

and

$$y' = s(x,y)$$

where r(x,y) and s(x,y) are the spatial transformations that produced the geometrically distorted image *g*(x',y'). For example, if r(x,y) = x / 2 and s(x,y) = y / 2, the distortion is simply a shrinking of the size of *f*(x,y) by one half in both spatial directions.

One can respond to the geometric distortion attack by simply reversing the initial spatial transform that was applied to the image. As in the case mentioned above, if the pixel position has been divided by two to shrink it, the original size can be retrieved by simply multiplying these values by two.

4.4.2 Grey-level interpolation

During the spatial transformation described above, resulting pixel values can result in being non-integer values. The distorted image *g* is digital therefore its pixel values are defined only as integer values. Thus the non-integer values for x' and y' causes mappings into locations of *g* for which no grey levels are defined. It then becomes necessary to infer what grey-level values should be at these locations using grey-level interpolation.

5. RESULTS AND DISCUSSION

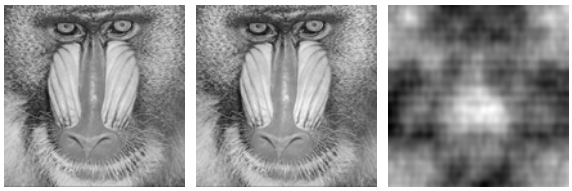
Two images were used in this analysis, lena.pgm, a very smooth image with large areas of similar colour, and baboon.pgm, a "rougher" picture with a lot of variation in shades.

The watermark was embedded and extracted using code Cox's algorithms [9, 10].

Difference images between the original and watermarked image were obtained, to show what affect the watermark had on the image.

a) b) c)

a) Original image, b) watermarked image, c) difference image



a) b) c)

a) Original image, b) watermarked image, c) difference image

5.1 Stirmarks JPEG compression:

The first results show the outcome of the compression testing. Both watermarked images underwent various factors of JPEG compression ranging from reduction to 90% of its size, to 10% of its original size. The graph in Fig 5.1 shows the correlation factor against the rate of compression. Both fair very well with the lowest correlation factor not going below a rate of 9.8 (1 being perfect). However the graph does indicate that "lena" can withstand a greater compression rate as opposed to "baboon". This goes back to the theory behind JPEG compression, where redundant pixels are gotten rid of. "Lena" being a smoother image (a bigger area of redundant pixels) can resist the more intense compression rates.

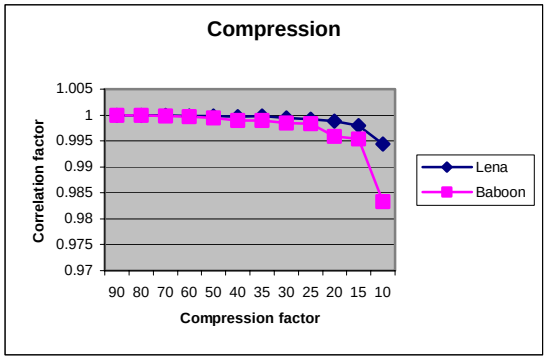


Fig 5.1. Correlation versus Compression Factor

Although one image appears to do better than the other in this test, visual quality of both images (observation) is severely reduced as the size of the image is reduced to 10%. The trade off between size and quality becomes apparent at this stage.

5.2 Stirmarks median filtering:

The median filtering test uses filters of size 2x2, 3x3 and 4x4. These filters are passed along both images. Fig 5.2 shows the results that were obtained.

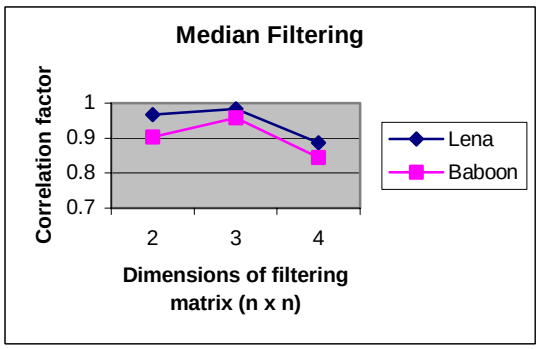


Fig 5.2. Correlation versus Filter size

As can be seen odd length filters give better results, as even length filters get the median of the middle two pixels in the area they cover and unnecessary image blurring may occur. It doesn't happen with odd filter lengths because the neighborhood has a clear center. Odd length filters give a more accurate result.

5.3 Stirmarks convolution filters:

As mentioned previously the Gaussian blurring filter is a low pass filtering process. Again its results are near perfect Fig 5.3.1, as low pass filtering does not tend to affect the watermarked pixels.

However we cannot say the same for the sharpening filter Fig 5.3.2, which is a high pass filter. This process affects the perceptually significant parts of the image i.e. the low frequency regions where the watermark is embedded.

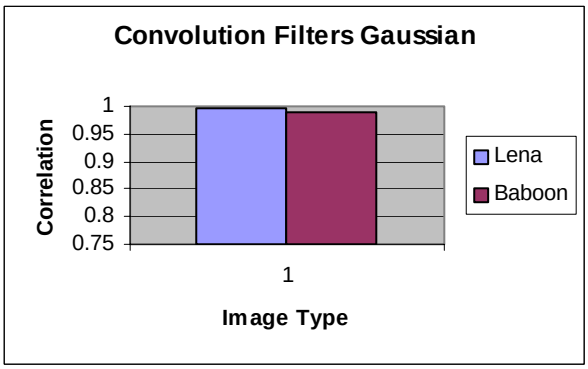


Fig 5.3.1 Correlation rates of images after Gaussian filtering

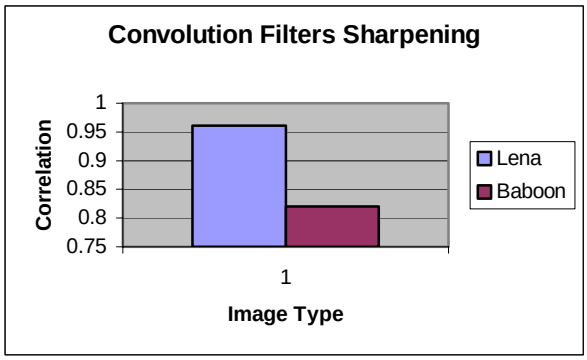


Fig 5.3.2 Correlation rates of images after sharpening filter

5.4 Stirmarks geometric distortions:

The results for the geometric distortion tests are more dramatic than the previous tests. In this test, each pixel value is transformed from its original position after it has been watermarked. This leaves detection much more difficult. Fig 5.4 displays the results.

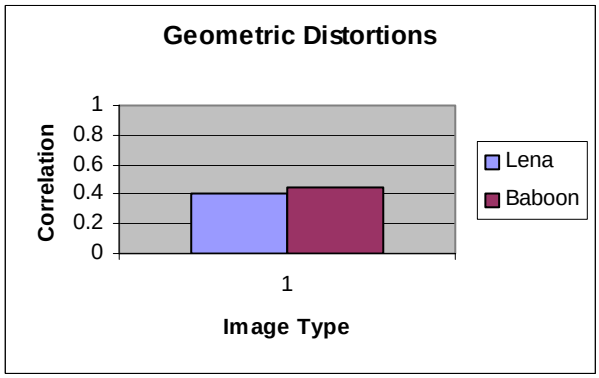


Fig 5.4 Results of geometric distortion of both images

The response however, is still well above the accepted correlation figure, which is generally 0.2. At 0.4, enough of the watermark can be extracted for it to be identified.

6. CONCLUSION

Cox's second-generation algorithm for watermark embedding and extraction has been described in this paper. Two images were watermarked using this algorithm and attacked using Stirmark. Stirmark is a benchmarking tool used to test robustness of algorithms. It disrupts various aspects of the images' composition and tries to make the watermark imperceptible.

A number of results were found after attempts were made to retrieve the marks and these were graphed as seen above.

Two different image types were used in this experiment. Lena, is a very 'smooth' image with a lot of areas of similar colour and Baboon, a very 'rough' image. It was found from the tests that were carried out that smooth images 'take' Cox's watermark embedding methods better than the more rough images. This is because he uses the DCT coefficients to embed the mark. These coefficients are the most perceptually significant areas of an image, i.e. large areas of similar colour, and so are more receptive to changing pixels values. This becomes apparent with Stirmarks convolution filter attacks, which involves low pass and high pass filtering. The smoother image has more low frequencies areas, higher spectral energy, and so they are not affected as much by low pass filtering as the rougher images are.

In summary it can be stated that Cox's algorithm works best with 'smooth' images, such as Lena, given that its design is based on the spread spectrum technique.

REFERENCES:

[1] O. Bruyndonckx, Jean-Jacques Quisquater, and Benoit M. Macq. Spatial method for copyright labelling of digital images. In IEEE Workshop on Nonlinear Signal and Image Processing '95, Thessalonika, Greece, pages 456-459, 1995.

[2] Marco Corvi and Gianluca Nicchiotta. Wavelet-based image watermarking for copyright protection. In Scandinavian Conference on Image Analysis SCIA '97, Lappeenranta, Finland, June 1997.

[3] Rakesh Dugad, Krishna Ratakonda, Narebdra Ahuja. A new wavelet-based scheme for watermarking images. In Proceedings of the IEEE International Conference on Image Processing, ICIP '98, Chicago, IL, USA, October 1998.

[4] Ingemar J. Cox, Joe Kilian, Tom Leighton, and Talal G. Shamoon. Secure spread spectrum watermarking for multimedia. In Proceedings of the IEEE ICIP '97, volume 6 pages 1673-1687, Santa Barbara, California, USA, 1997.

[5] http://www.ece.purdue.edu/~ace/jpeg-tut/jpgdct1.html

[6] http://www.cl.cam.ac.uk/~fapp2/

[7] 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.

[8] http://www.faqs.org/faqs/jpeg-faq/part1/

[9] http://www.cosy.sbg.ac.at/~pmeerw/

[10] Peter Meerwald, Digital Image Watermarking in the Wavelet Transform Domain, Masters Thesis, Department of Scientific Computing, University of Salzburg, Austria, January 2001.