Multimedia Rights Protection
Digital Image Watermarking Techniques
James Padgett

Supervisor: Mr. Bill Whyte
MSc Distributed Multimedia Systems
Session: 2001/2002

The candidate confirms that the work submitted is their own and the appropriate credit has been given where reference has been made to the work of others.

I understand that failure to attribute material which is obtained from another source may be considered as plagiarism.

(Signature of student) _______________________________
Summary

Digital Image Watermarking is a solution to the problem of multimedia copyright violation. It is the insertion of a unique identifier into a digital image so that the owner or authorised user can legally use the image without breaking copyright law. The law and application demand is driving the industry forward to develop a secure robust watermarking solution.

The report discusses the relevant image processing theory involved in digitally watermarking an electronic image. It examines some cutting edge algorithms utilising the Discrete Cosine Transformation to embed the watermark within the frequency domain. Also discussed are the properties of the Human Visual System and ways in which they can be exploited to produce a more robust / less perceptible watermark.

The report goes on to determine the effects of varying the watermark embedding strength and the watermark length of the current industry standard watermarking algorithm by Cox (1995). It is subjected to 8 experiments which simulate the type of attacks which an image can expect to be subjected to if present within the Internet domain. These attacks involve JPEG compression, image scaling, dithering, printing and rescanning and several collusion attacks. A software implementation of the algorithm is used to test the technique on a standard reference image and used to measure the effects of the attacks. The experimental results are provided to support the conclusions of the report.

A complete image protection solution is suggested which provides the current best available protection that a digital image can expect. Avenues for further research are suggested which centre on the exploitation of weaknesses in the Human Visual System to produce a more strongly embedded watermarking scheme. Other areas for research involve the use of the Direct Wavelet Transform in preference to the DCT.
Acknowledgements

- Thanks to Bill Whyte for the academic and help and advice given throughout the project.
- Thanks to my tutor Eric Atwell for the pastoral advice given throughout the MSc course.
- I would like to thank the support of my parents, without which it would not have been possible to complete this MSc.
# Multimedia Rights Protection – Digital Image Watermarking Techniques

## Contents

Summary ........................................................................................................... ii
Acknowledgements ......................................................................................... iii
Contents .......................................................................................................... iv

1. Introduction ............................................................................................... 1  
   1.1 The Law ................................................................................................. 1  
   1.2 Encryption Systems ............................................................................... 1  
   1.3 Watermarking Applications .................................................................. 2  
   1.4 Watermarking Principles ...................................................................... 3  

2. Watermarking Theory ............................................................................... 4  
   2.1 Embedding ............................................................................................ 5  
   2.2 Extraction .............................................................................................. 11  
   2.3 Detection ............................................................................................... 11  
   2.4 Perceptual Watermarking – The Human Visual System ...................... 12  
   2.5 Spread Spectrum Watermarking ........................................................... 13  

3. Experimental Attacks ................................................................................ 17  
   3.1 Tool Construction .................................................................................. 17  
   3.2 Effect of Scaling Factor (α) on Image quality ........................................ 19  
   3.3 Effect of Watermark Length (n) on Image quality ................................ 19  
   3.4 Experiment 1: Image Scaling ............................................................... 20  
   3.5 Experiment 2: JPEG Compression ....................................................... 20  
   3.6 Experiment 3: Clipping ......................................................................... 22  
   3.7 Experiment 4: Dithering ....................................................................... 23  
   3.8 Experiment 5: Digital / Analogue – Analogue / Digital Conversion ...... 24  
   3.9 Experiment 6: Uniqueness of Watermark ........................................... 24  
   3.10 Experiment 7: Multiple Watermarks ............................................... 24  
   3.11 Experiment 8: Collusion .................................................................... 25  

4. Results ........................................................................................................ 26  
   4.1 Experiment 1: Image Scaling ............................................................... 27  
   4.2 Experiment 2: JPEG Compression ....................................................... 28  
   4.3 Experiment 3: Clipping ......................................................................... 30  
   4.4 Experiment 4: Dithering ....................................................................... 31  
   4.5 Experiment 5: Digital / Analogue – Analogue / Digital Conversion ...... 32  
   4.6 Experiment 6: Uniqueness of Watermark ........................................... 33
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.7</td>
<td>Experiment 7: Multiple Watermarks</td>
<td>34</td>
</tr>
<tr>
<td>4.8</td>
<td>Experiment 8: Collusion</td>
<td>35</td>
</tr>
<tr>
<td>5.</td>
<td>Discussion</td>
<td>36</td>
</tr>
<tr>
<td>6.</td>
<td>Conclusion</td>
<td>40</td>
</tr>
<tr>
<td>7.</td>
<td>Future Work</td>
<td>42</td>
</tr>
<tr>
<td>8.</td>
<td>Evaluation of Objectives</td>
<td>43</td>
</tr>
<tr>
<td>9.</td>
<td>References</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>Appendix A: Personal Experience</td>
<td>47</td>
</tr>
<tr>
<td></td>
<td>Appendix B: Objectives and Deliverables</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>Appendix C: Interim Report Header</td>
<td>51</td>
</tr>
</tbody>
</table>
1. Introduction

The rapid uptake of affordable broadband internet access and the spread of Peer-2-Peer file sharing services have increased the potential for misuse and theft of copyrighted digital multimedia data. One area where this problem is most acute is digital image data. The problems of misuse and theft exist due to the very nature of digital multimedia data; when a reproduction is made it is identical to the original in every way. The ease with which it can be copied and the speed with which it can be illegally distributed add up to make multimedia rights protection a pressing concern for today’s Internet. This problem is preventing the creators / producers of such multimedia content from distributing their work. What is needed is a robust, secure and reliable multimedia rights protection technology.

1.1 The Law

The realisation that a problem exists has been acknowledged by both the United States government and the European Union commission. In the US the Digital Millennium Copyright Act (DMCA 1998) and in the EU the European Union Copyright Directive (EUCD 2001) both aim to replace sketchy digital multimedia copyright law in their respective countries. Both make it illegal to circumvent any copyright protection measure, however the EUCD goes further, making it illegal to produce a tool or service that could be used to circumvent the copyright protection.

1.2 Encryption Systems

Public key encryption systems such as the RSA algorithm (Litterio 1999) have traditionally been used to protect copyrighted data in transit between trusted parties (Ó Ruanaidh 1996). This system is only effective when the trusted parties do not lose, redistribute or pirate the unencrypted data. If the data has numerous recipients there will be no way of proving which of them was responsible for the loss of integrity. Any digital watermarking technology would, when used in conjunction with such a public key encryption system, form a complete multimedia rights protection system.

This report focuses on the current state of the art digital watermarking techniques as applied to digital image data. Digital image watermarking involves the insertion into an image a serial number, logo or unique code that will remain for the lifetime of the image. The unique code typically shows ownership of the image and may include relevant copyright information such as the issuer and possibly the purchaser of the image. Multiple watermarks may also be embedded to show, for
example, the creator and the purchaser, or to track ownership. A typical watermark is usually a function of the information that is to be embedded and can be related to a public / private key watermark generation system. These measures are all designed to prevent misuse or theft of the image.

1.3 Watermarking Applications

Although this report focuses on digital image watermarking as a method to protect the multimedia rights of the creator or owner of a work; image watermarking has numerous applications. Fingerprinting, copy protection, image authentication and steganography (data hiding) are all applications that have benefited from digital image watermarking.

Fingerprinting seeks to track illegal copies of the same image by embedding different copies of the watermark into each distributed copy of that image. Such watermarks provide information regarding the purchaser rather than the producer / creator and help to identify the origin of any pirated images. It is synonymous with a software serial number; however such a system is vulnerable to deception attacks such as collusion.

Watermarking for copy protection prevents illegal copying of an image by indicating the copy status of the image. This may be “copying never” or “copy once”; the ability to copy the image depends upon the status of the watermark. The success of copy protection watermarking depends heavily on whether the system is open, such as a desktop pc or proprietary, such as a DVD player. Open systems are more prone to abuse; given their very nature it is easy to alter software readers, making any watermarking scheme ultimately breakable.

Image authentication aims to reveal alterations to an image with the use of a “fragile” watermark that is highly sensitive to tampering. Alterations to an image with such a watermark can be easily discovered by comparison with the original image. Such an application is well suited to the media world where image modification is often used to heighten the impact of a story.

Perhaps the oldest application for image watermarking is steganography or data hiding. It is the art of hiding and transmitting data within a digital image whilst trying to conceal the existence of the data. In essence the data is camouflaged to hide its existence and conceal the fact that it is being sent. It differs from cryptography in that cryptography scrambles the data being transmitted to protect the information within the data; rather than camouflaging its existence. Cryptography makes
no secret of the fact that data is being transmitted and uses codes and cipers to encrypt the data. Steganography tries to conceal the existence of the data.

1.4 Watermarking Principles

Digital image watermarks can either be visible, such as an embossed logo superimposed upon the image or invisible, embedded imperceptibly within the image data. In the case of invisible watermarking the watermark must have a number of qualities if it is to be successful in protecting the creator’s multimedia rights. Barnet (1999) and Swanson (1998) highlight some of the most important qualities.

**Imperceptibility.** The watermark itself should not be visible by the human visual system (HVS) and should not degrade the image quality.

**Unambiguous.** Once detected the watermark should unequivocally identify the rightful copyright holder.

**Robustness.** The image once watermarked should be robust to three types of attacks common to digital images: signal processing, geometric deformation and deception. Attempts to remove the watermark should result in the gradual degradation of image quality, such that if the watermark is wholly removed then the image will be useless to the attacker.

**Signal Processing.** Such attacks focus on operations commonly carried out on digital images such as: linear / nonlinear filtering, additive noise (Gaussian), JPEG compression, resampling (histogram functions) and Digital-to-Analogue and Analogue-to-Digital conversion (printing and rescanning)

**Geometric Deformation.** Image deformation involves any operation which introduces a distortion to the image: translation, rotation and scaling (temporal or spatial). It may also include changes to pixel/s.

**Deception.** Counterfeiting a watermark by forgery or collusion using multiple copies of a valid watermarked image. These may be combined to produce a forged watermarked image which on examination would return a false positive identification of the owner, ie. The watermark is valid but it does not identify the rightful owner.
2. Watermarking Theory

There are 3 main books offering a reasonably high level explanation of the subject of digital image watermarking by Katzenbeisser (1999), Cox (2001) and Johnson (2001). The first deals mainly with steganography and only briefly touches on digital watermarking. The second is wholly related to digital watermarking, offering a high level comprehensive review of the current state of the art techniques with example code. The third is mainly a steganography text but offers some incite into the types of attack that a watermark might come up against. They should be read only to gain background knowledge of the subject; specific detail on algorithms should be sort from the papers published by the algorithm creators.

There are several good review papers detailing current industry standard algorithms without going into to much depth. The papers are those by Nikolaidis (1999) and Podilchuk (2001) and should be read before any in depth study regarding particular algorithms is attempted.

No standard terminology has been coined for the subject of digital watermarking. Most algorithms after 1996 share a common model given by Pfitzmann (1996), however this relates more to steganographic research rather than digital image watermarking. This report will not follow the terminology as given by Pfitzmann (1996) but will aim to use uniform terminology throughout.

There are four stages to an image watermarking algorithm; they are watermark generation, embedding, extraction and detection. The techniques used to embed the watermark can be classified into two groups: those operating in the spatial domain and those operating in a transform domain, such as the frequency domain. Note that spatial domain image watermarking such as that by van Schyndel (1994), Langelaar (1996) and Wolfgang (1996) is not covered in this report, but provide valuable alternatives to transform domain watermarking. Their work centres on using Least Significant Bit (LSB) insertion and weaknesses within the Human Visual System (HVS) to embed the watermark.

Watermarking algorithms can also be classified according to the way in which the watermark is extracted from the image. When the original unmarked image needs to be referenced during the extraction stage, the algorithm is said to be non-blind. When only some details from the original unmarked image are required the algorithm is said to be semi-blind. Finally, when no reference to the original image is needed the algorithm is said to be blind.
The original image for the purposes of the practical work of this report is a standard watermarking reference image called Lena (Rosenberg 2001), Figure 1. Originally from the centre-fold of the November 1972 Playboy magazine, she is ideal for the application of watermarking due to the spread of frequencies within the image. The low frequency area by her shoulder and high frequency of her hair make watermark embedding difficult.

2.1 Embedding

Figure 2 is the process of embedding the watermark within the image.

Although there are many formats in which digital images can be stored, two of the most popular in image processing circles are the JPEG standard (Wallace 1991), and the Portable Greymap standard (PGM). For this report the PGM format will be used in all practical experiments with the JPEG standard being used for the lossy compression attack.
When a digitized image is stored, its spectral colour components are represented by pixels (Crane 1997).

![Figure 3 Example of how number of pixels affects quality of image (a) 512 x 512; (b) 32 x 32](image)

Each pixel represents an area of the image and is represented by a pixel value usually within an integer range. Figure 3 illustrates how the number of pixels can affect the quality of the image. As pixel count increases so does file size.

![Figure 4 Example of how bit depth affects quality of image (a) 6bits; (b) 1bit](image)

Figure 4 illustrates how the number bits per pixel can affect the quality of the image. As the number of bits per pixel increases so does file size.

Figure 1 (Lena) shows a 512 x 512 greyscale image with 8 bits per pixel (8bpp). The image is 512 pixels wide by 512 pixels long, and the entire picture is represented by 262144 pixels. Lena is an 8bpp greyscale image meaning each pixel has a value between 0 – 255.

Size of file in bytes: $\frac{512 \times 512 \times 8}{8 \times 1024} = 256KBytes$

A typical example (Rabbani 1999) of a localized 8 x 8 block of pixel values for the Lena image might be:
Each value represents a greyscale point on the 256 value scale between black and white. Assuming this is our original image to be watermarked, using Figure 2 as a guide, the image must be transformed using the Direct Cosine Transformation (DCT). The DCT is a Fourier domain (Bhaskaran 1995) based transform used in the JPEG standard (Wallace 1991), (RAO 1996).

The 2D DCT:

$$H(u,v) = \frac{2}{\sqrt{MN}} C(u)C(v) \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} h(x,y) \cos \left( \frac{(2x+1)u\pi}{2M} \right) \cos \left( \frac{(2y+1)v\pi}{2N} \right)$$

and the 2D iDCT:

$$h(x,y) = \frac{2}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} C(u)C(v)H(u,v) \cos \left( \frac{(2x+1)u\pi}{2M} \right) \cos \left( \frac{(2y+1)v\pi}{2N} \right)$$

where

$$C(\gamma) = \begin{cases} 
\frac{1}{\sqrt{2}} & \text{for } \gamma = 0 \\
1 & \text{for } \gamma > 0 
\end{cases}$$

and

- $u, v =$ pixel coordinates for the transformed block
- $x, y =$ pixel coordinates of original image block
- $H(u, v) =$ value of transform domain coefficient at coordinate given by $u, v$
- $h(x, y) =$ value of spatial domain pixel coefficient at coordinate given by $x, y$
- $M =$ Number of rows in block to be transformed
- $N =$ Number of columns in block to be transformed

The 2D DCT given above is the standard algorithm cited when a 2D array of pixel values needs to be transformed. The DCT is itself a popular area for research with the development centred on fast...
and computationally efficient algorithms such as the one developed by Lee (1984) which uses 2 eight point 1D DCT algorithms for the rows and columns. This leads to a lower level of complexity and a faster compute time. Once transformed the spatial image block $h(x, y)$ given above is represented by a transform (frequency) block:

$$H(u, v) = \begin{bmatrix}
1260 & -1 & -12 & -5 & 2 & 2 & -3 & 1 \\
-23 & -17 & -6 & -3 & 0 & 0 & -1 & \\
-11 & -9 & -2 & 2 & 0 & -1 & -1 & 0 \\
-7 & -2 & 0 & 1 & 1 & 0 & 0 & 0 \\
-1 & -1 & 1 & 2 & 0 & -1 & 1 & 1 \\
2 & 0 & 2 & 0 & -1 & 1 & 1 & -1 \\
-1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\
-3 & 2 & -4 & -2 & 2 & 1 & -1 & 0
\end{bmatrix}$$

The upper left coefficient in this transform block is called the DC coefficient and is proportional to the average brightness within the spatial domain image block from which it was calculated. All the other coefficients are referred to as the AC coefficients and represent the different frequency components present within the spatial image block. The coefficients in the top left correspond to the low frequency components, whilst those in the bottom right represent the high frequency components. The low frequency coefficients are those which are perceptually most significant within the image. They are the most sensitive to alteration and can destroy the fidelity of the image if they are altered beyond their perceptual capacity. The high frequency coefficients represent those which are perceptually least significant. Despite this they are not totally without purpose and add fine detail to the image. The reader is reminded that the DCT example given is for one localised 8 x 8 block within the Lena image. This has to be repeated for N 8 x 8 blocks for the entire image to complete a full DCT transformation, as happens in the JPEG image compression standard.

Figure 5 Image with (a) DC coefficient only; (b) DC plus 3 AC coefficients; (c) All coefficients
To illustrate the effect of the higher frequency coefficients, Figure 5 shows the effect of (a) an image which has been subjected to the DCT but only the DC coefficient for each 8x8 block has been used to reconstruct the image, with all AC coefficients set to zero. This is equivalent to very aggressive image compression. In (b) only the DC coefficient and the first 3 AC coefficients (zig-zag pattern) of each 8x8 block are retained and (c) all coefficients are retained. Notice how the fine detail increases as more of the high frequency coefficients are used.

Given $H(u, v)$ the watermark $W(k)$ is used to modulate the chosen transform block coefficients given some formulae described by the given watermarking algorithm. A few examples are:

$$H'(u, v) = H(u, v) + W(k)$$

$$H'(u, v) = H(u, v) + \alpha W(k)$$

$$H'(u, v) = H(u, v) + \alpha H(u, v)W(k)$$

where

$\alpha = $ Scaling Factor

The scaling factor $\alpha$ depends primarily on the properties of the HVS and should alter depending on whether the coefficient is in the low, middle or high frequency range. By tailoring the scaling factor makes it easier to place a stronger watermark within the transform block without causing perceptible distortion to the spatial domain image. Which coefficients are modulated depends upon the watermarking algorithm used; more detailed explanation of specific algorithms will be given later. For now it is adequate to say that some or all of the coefficients may be altered.

After the watermark $W(k)$ is combined with the transform block it is inverse transformed using the inverse Discrete Cosine Transformation (iDCT). This is a similar process to the DCT but transforms the transform domain block back into the spatial domain. Combining the watermark is simply the addition of noise to the image, except the noise is not random and serves a valid purpose. Although one of the qualities of a successful watermark is imperceptibility, the addition of noise to the transform domain coefficients will alter the pixel values in the spatial domain and so alter the image. If the algorithm is good then the changes introduced to the pixel values will be imperceptible to the HVS. The reconstructed spatial domain block is now:
Introduction of the watermark has lead to the following errors within the block:

\[
\hat{h}(x, y) = \begin{bmatrix}
144 & 146 & 149 & 152 & 154 & 156 & 156 & 156 \\
148 & 150 & 152 & 154 & 156 & 156 & 156 & 156 \\
155 & 156 & 157 & 158 & 158 & 157 & 156 & 155 \\
160 & 161 & 161 & 162 & 161 & 159 & 157 & 155 \\
159 & 163 & 164 & 163 & 162 & 160 & 158 & 156 \\
163 & 163 & 164 & 164 & 162 & 160 & 158 & 157 \\
160 & 161 & 162 & 162 & 162 & 161 & 159 & 158 \\
158 & 159 & 161 & 161 & 162 & 161 & 159 & 158 
\end{bmatrix}
\]

\[
e(x, y) = \begin{bmatrix}
-5 & -2 & 0 & 1 & 1 & -1 & -1 & -1 \\
-4 & 1 & 1 & 2 & 3 & 0 & 0 & 0 \\
-5 & -1 & 3 & 5 & 0 & -1 & 0 & 1 \\
-1 & 0 & 1 & -2 & -1 & 0 & 2 & 4 \\
-4 & -3 & -3 & -1 & 0 & -5 & -3 & -1 \\
-2 & -2 & -3 & -3 & -2 & -3 & -1 & 0 \\
2 & 1 & -1 & 1 & 0 & -4 & -2 & -1 \\
4 & 3 & 0 & 0 & 1 & -3 & -1 & 0 
\end{bmatrix}
\]

Such errors are imperceptible to the HVS given that the changes in greyscale indicated are between -5 to 3. Shown below is the original image pixel block \( h(x, y) \) and the watermarked image pixel block \( \hat{h}(x, y) \). There is little visible difference between them; given that these are tiny 8x8 pixel blocks, the differences would be imperceptible in the context of a 512x512 pixel image such as Lena.
2.2 Extraction

Extraction of the embedded watermark from the host image is a similar process to that of embedding. It is essentially the same process in reverse and depending whether the algorithm being used is blind / semi-blind / non-blind; determines the steps needed to extract the watermark. If the algorithm is non-blind then it is made easier if the original unmarked image and the secret key used to embed the watermark are both known. Such an algorithm is often referred to as a private key system given that the key used to embed the watermark is secret. Blind algorithms conversely are often referred to as public key systems given they have no reference to either the original unmarked image or the secret key. Systems such as these reference a public key to verify the integrity of the watermark; however, work by Kalker (1998) proves such systems have a number of inherent problems.

Given a new possibly corrupted host image $\hat{h}^*(x, y)$, it is assumed that the image has been subjected to a number of attacks similar to those described in the introduction. By comparison with the original image $h(x, y)$, the watermark $W(k)$ and using knowledge of the algorithm used, a corrupted watermark $W^*(k)$ can be determined. Comparison of $\hat{h}^*(x, y)$ and $h(x, y)$ takes place in the transform domain as this is where the watermark was embedded; giving $H(u, v)$ and $\hat{H}^*(u, v)$.

2.3 Detection

After the corrupted watermark $W^*(k)$ has been extracted from the new host image $\hat{h}^*(x, y)$ it is compared with the original watermark $W(k)$ embedded in the original image $h(x, y)$, to obtain our watermarked image $\hat{h}(x, y)$.

Comparison of $W^*(k)$ with $W(k)$ can be made via one of many correlation measures found in statistical mathematics such as Gosset’s t-distribution or Fisher’s z-transform. Whatever method is used depends upon the type of answer that is required from the detection system. A simple system might return a basic YES/NO answer as to whether the original watermark has been detected in the new host image $\hat{h}^*(x, y)$. More complicated systems might return a normalised value between 0 – 1; 0 being no match between the two watermarks and 1 being an exact match. This system can be adapted to a YES/NO system by using a threshold value. If the result is above this value, then the watermark is detected, if below, then there is no watermark.
There are problems with any detection system; they are false positives and false negatives (Therrien 1989). A false positive occurs when the detection process returns a positive result indicating that a watermark is present, when in reality there isn’t one. A false negative result occurs when the detection process returns a negative result indicating that no watermark present, when in fact there is one. If digital image watermarking is to have any credibility in the world of multimedia rights protection then errors such as these cannot be tolerated.

2.4 Perceptual Watermarking – The Human Visual System

There have been many models constructed of the Human Visual System such as those by Watson (1993), (Watson 1993a) and Wolfgang (1999) which have suggested that the eye is sensitive to both frequency and luminance. Watson proposed a block-based DCT perceptual model similar to the JPEG compression standard. It uses a modified quantisation matrix to take into account known properties of the HVS. The model uses a sensitivity function, 2 masking components and a pooling component.

**Frequency Sensitivity** An image is divided up into 8x8 pixel blocks $c_{x, y, z}$ and each is DCT transformed as explained in chapter 2.1. For each of the 8x8 pixel blocks a sensitivity table is defined. The sensitivity table $t[i, j]$ is an 8x8 block containing values representing the amount by which the corresponding DCT coefficients of $c_{x, y, z}$ can be altered before a difference can be noticed within the spatial image block. A smaller value in $t[i, j]$ indicates that the corresponding coefficient in $c_{x, y, z}$ can be altered less. A typical table of values would contain small numbers in the top left of the block, and large numbers in the bottom right. This follows with what has been discussed in chapter 2 in that the lower frequencies in the top left of the DCT transform block are perceptually the most significant, offering the least capacity for any watermark payload. Whilst the higher frequency in the bottom right of the DCT transform block are perceptually least significant, offering the greatest capacity for any watermark payload.

**Luminance Masking** refers to the fact that a DCT coefficient has a greater capacity for change without being notice in blocks with a higher average brightness. Given that the average brightness of the block is proportional to the DC term, then blocks with high DC coefficient values offer a greater capacity for watermarking. Therefore it follows that brighter areas of an image offer more capacity for watermarking. Each coefficient in the sensitivity table $t[i, j]$ is altered to take account of this. All values within $t[i, j]$ increase proportionally as the average brightness of $c_{x, y, z}$ increases; giving $tL[i, j, z]$, the luminance masked threshold table.
**Contrast masking** is the detectability of one signal in the presence of another signal. It is the reduction in visibility of a change in one frequency due to the energy present in that frequency. In the case of an image and a watermark, the image acts as the background signal which reduces the visibility of the watermark due to the energy present within the image signal. If there is a lot of energy present within a certain frequency then any change to that frequency introduced through the addition of a watermark will be less. The luminance masked threshold \( t_{i,j,z} \) is modified to take account of this. Contrast masking affects the capacity of the coefficients to accept watermark information, giving the masking thresholds \( s_{i,j,z} \). These threshold values are used in conjunction with the 8x8 DCT block values \( c_{x,y,z} \) to achieve a perceptual watermarking algorithm. The threshold values are synonymous with the scaling factor \( \alpha \). The difference is the scaling factor alters according to the perceptual model to achieve the maximum capacity for each of the coefficients to accept the watermark without changing the fidelity of the image.

### 2.5 Spread Spectrum Watermarking

The algorithm proposed by Cox (1995) and later, (Cox 1997) details a secure, robust spread spectrum algorithm for the watermarking of multimedia files. The technique can be used to watermark images as well as audio and video. The algorithm is one of the most prominent in the watermarking community and despite its age is still one of the most robust against attack. Watermarking takes place in the frequency domain, but unlike many block based DCT algorithms (Barni 1998) the entire image is subjected to one large DCT. The paper argues that the watermark must be embedded in the perceptually most significant frequencies of the image for it to maintain the qualities required of a successful algorithm.

The watermark \( W(k) \) length \( n \) is a sequence of pseudo-random real numbers with zero mean and a variance of 1:

\[
W(k) = k_1, ..., k_n
\]

A normally distributed watermark is chosen as it is robust against collusion attacks especially those which average several watermarked copies of an image to produce an unwatermarked image.
$W(k)$ is embedded within the $n$ largest magnitude AC coefficients of $H(u, v)$, which is the full image DCT of the image $h(x, y)$; obtaining $\hat{h}(x, y)$, the watermarked image. The formula for modulating the chosen AC coefficients (of $H'(u, v)$) is given by:

$$H'(u, v) = H(u, v) + \alpha W(k)$$

where

$\alpha = $ Scaling Factor

The scaling factor is the amount by which the AC coefficient can be altered without the perceptual quality of the image being affected. It is different for each coefficient, and generally the low frequency coefficients (those at the top left of the frequency block) have the least capacity for absorbing the watermark energy. Embedding the watermark within the low frequency coefficients is the most beneficial; it is these frequencies which are the most sensitive to tampering and any attack on them will most likely destroy the image. The algorithm is described as spread spectrum because by choosing the $n$ highest magnitude AC coefficients as the ones to modulate, the algorithm is almost guaranteeing that they will be spread throughout the entire transform domain block. In doing so, the watermark will be present within the low frequency coefficients given that these are typically of the highest magnitude; but also within the middle and high frequencies of the image. Take $H(u, v)$, the 8x8 block example used in section 2.1:

$$H(u, v) = \begin{bmatrix}
1260 & -1 & -12 & -5 & 2 & 2 & -3 & 1 \\
-23 & -17 & -6 & -3 & -3 & 0 & 0 & -1 \\
-11 & -9 & -2 & 2 & 0 & -1 & -1 & 0 \\
-7 & -2 & 0 & 1 & 1 & 0 & 0 & 0 \\
-1 & -1 & 1 & 2 & 0 & -1 & 1 & 1 \\
2 & 0 & 2 & 0 & -1 & 1 & 1 & -1 \\
-1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\
-3 & 2 & -4 & -2 & 2 & 1 & -1 & 0
\end{bmatrix}$$

Supposing a watermark of length 9 is to be embedded within this 8x8 image, then the coefficients marked in bold are the 9 highest magnitude AC coefficients and will be used as the modulate coefficients. Their distribution within the block is centred mainly in the low frequency spectrum of the image, however one is located in the middle to high frequency range and so the watermark has achieved the spread throughout the spectrum of the image block. This example is exaggerated, but
in a 512 x 512 image block and using a watermark of length 1000, the highest magnitude coefficients will be more widely distributed throughout the frequencies.

Cox’s paper suggests that “…as the number of altered components (coefficients) are increased the extent to which they must be altered decreases.” This suggests that an image has a given capacity for absorbing a watermark without it affecting the visual appearance of that image. There will always be a trade off between the scaling factor and watermark length due to the differing modification tolerance of the transform coefficients. It is the purpose of this report to examine this trade off and establish if there is an optimum value of scaling factor $\alpha$ and length $n$ of watermark $W(k)$ for the Cox algorithm when exposed to a suite of common image attacks. Cox (1995) leaves this question unanswered and as such the writer of this report can find no literature that has tested the hypothesis. Values of $\alpha = 0.1$ and $n = 1000$ are used by Cox, leaving an interesting opportunity for further investigation.

Extraction of a corrupted watermark $W^*(k)$ is achieved by comparison of the watermarked image $\hat{h}(x, y)$ and the possibly corrupted new host image $\hat{h}^*(x, y)$. Given that the formula for altering the AC coefficients is reversible, the availability of the original image and knowledge of the algorithm, a possibly corrupted watermark $W^*(k)$ can be found.

Comparison of the corrupted watermark and the original watermark is achieved using the normalised correlation for pseudo-random sequences:

$$\text{Sim}(W(k), W^*(k)) = \frac{W^*(k)W(k)}{\sqrt{W^*(k)W^*(k)}}$$

The two watermarks, length $n$, are treated as a vector with $n$ dimensions. In this way a normalised dot product of the 2 vectors is taken to establish their similarity. By applying a threshold $\tau$ to the similarity test, a decision can be made as to whether the watermarks match:

$$\text{Sim}(W(k), W^*(k)) \geq \tau$$

Then the two watermarks match and the watermark which was in the watermarked image $\hat{h}(x, y)$, is also present within the possibly corrupted host image $\hat{h}^*(x, y)$. However if:

$$\text{Sim}(W(k), W^*(k)) < \tau$$
Then the two watermarks do not match and there is no watermark within $\hat{h}(x, y)$. 
3. Experimental Attacks

The intension of the report is to provide a deeper analysis of the Cox (1995) algorithm than was given by the authors when the paper was first published. More specifically, what is the effect of changing the values of the scaling factor $\alpha$ and the watermark length $n$? A series of experimental attacks will be used, similar to those used in the original paper. These attacks aim to simulate what happens to a watermarked image when published on the World Wide Web.

3.1 Tool Construction

The frequency domain based tools created to insert the watermark within the images are an extension to source code produced by Meerwald (2001) and with help from Kelley (1995), Kelley (1996) and Liberty (2000), recompiled using a custom makefile. The chosen operating system is Linux due to the availability of:

- GCC, the C/C++ compiler / linker
- a Java Image Processing API (Efford 2000) for the implementation of the image processing functions needed to simulate the image attacks

And due to the authors preference for this operating system given much of the software is open source and freely available and due to the helpful book by Hawkins (2002).

There are four executable files used in the watermarking process, each corresponding to the 4 stages as set out in chapter 2. They are watermark generation, embedding, extraction and detection. The four watermark programs have the following usage.

Generation:

```
./gen_cox_sig [-a alpha] [-m mean] [-n length] [-d variance] [-o file]
```

- `a alpha`: Specify the coefficient scaling factor $\alpha$.
- `m mean`: Specify the mean values of the Pseudo-Random watermark number sequence
- `n length`: Specify the length of the watermark, e.g the number of coefficients to be modulated
- `d variance`: Specify the variance of the Pseudo-Random watermark number sequence
- `o file`: Name of the output file containing the signature file
Embedding:

```
./wm_cox_e [-s file] [-o file] file
```

- `s file` Name of signature file containing the watermark sequence.
- `o file` Name of the .pgm output file for the watermarked image.
- `file` Name of the .pgm input file being watermarked, e.g. lena.pgm.

Extraction:

```
./wm_cox_d [-s file] [-o file] [-i file] file
```

- `s file` Name of signature file containing the original watermark sequence.
- `o file` Name of signature file to store recovered watermark.
- `i file` Name of the original input file, e.g. lena.pgm.
- `file` Name of the .pgm output file for the watermarked image.

Detection:

```
./cmp_cox_sig [-s file] file
```

- `s file` Name of signature file containing the original watermark sequence.
- `file` Name of signature file to store recovered watermark.

The files are supplied with the standard reference image lena.pgm, a makefile and the source code needed to recompile the programs. The makefile may need reediting to point to your gcc executable. All files need to be in the same directory. Usage:

```
made && make install && make clean
```
3.2 Effect of Scaling Factor (α) on Image quality

The following pictures detail how image quality is degraded as the scaling factor α is increased. The watermark length n remains at 1000 for all values of α. Much above α = 0.4 the watermark is noticeable within the image; however the higher α values will be used in the experiments to exaggerate the effect of the scaling factor.

<table>
<thead>
<tr>
<th>α = 0.1</th>
<th>α = 0.2</th>
<th>α = 0.3</th>
<th>α = 0.4</th>
<th>α = 0.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>α = 0.6</td>
<td>α = 0.7</td>
<td>α = 0.8</td>
<td>α = 0.9</td>
<td>α = 0.10</td>
</tr>
</tbody>
</table>

3.3 Effect of Watermark Length (n) on Image quality

These pictures detail how image quality is degraded as the watermark length n is increased. The scaling factor α remains at 0.1 for all values of n. Image degradation due to increases in watermark length are less noticeable than increases in scaling factor. This may be as a result of the scaling factor being low or may be due to the fact that as watermark length increases it is embedded in coefficients that are less and less significant, i.e. the higher frequency coefficients.

<table>
<thead>
<tr>
<th>n = 200</th>
<th>n = 400</th>
<th>n = 600</th>
<th>n = 800</th>
<th>n = 1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>n = 1200</td>
<td>n = 1400</td>
<td>n = 1600</td>
<td>n = 1800</td>
<td>n = 2000</td>
</tr>
</tbody>
</table>
3.4 Experiment 1: Image Scaling

The image scaling attack scales the watermarked image to half its size from 512 x 512 pixels to 256 x 256 pixels then subjects the newly rescaled image to a low pass filter and finally rescales it back to 512 x 512 pixels; its original size.

Rescaling to a smaller resolution essentially reduces the quality of the image, with the loss of fine detail due to averaging effects. Such an effect is akin to JPEG compression, resulting in the loss of much of the information from the high frequency spectrum of the image. The low pass filter is introduced to exaggerate this effect, producing a decidedly blocky image when it is rescaled. This attack replicates what typically happens when digital images are embedded in document processors such as MS Word™ and Adobe Acrobat™.

3.5 Experiment 2: JPEG Compression

The JPEG compression attack converts the PGM file to a JPEG JFIF compliant file format with 10% quality and 0% smoothing and then reconverts back to the PGM format.

The conversion between PGM and JPEG formats does little to damage the watermark, indeed the image should contain identical pixel values in either format; both being 8 bpp. The JPEG compression algorithm is a more testing attack on the robustness of the watermark. Once in JPEG JFIF compliant format the image may be lossy compressed; meaning that much of the image file size reduction is achieved with the permanent loss of image detail and quality.
This JPEG compression algorithm is similar to the watermarking scheme by Watson (1993); indeed such a scheme invariably has its roots within the JPEG compression algorithm. The same 8 x 8 block is used here as was used in section 2.

The image is initially divided into 8 x 8 pixel blocks which are then transformed using the DCT:

\[
\]

\[
\Rightarrow \text{ DCT Transform}
\]

\[
H(u, v) = \begin{bmatrix} 1260 & -1 & -12 & -5 & 2 & 2 & -3 & 1 \\ -23 & -17 & -6 & -3 & -3 & 0 & 0 & -1 \\ -11 & -9 & -2 & 2 & 0 & -1 & -1 & 0 \\ -7 & -2 & 0 & 1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 2 & 0 & -1 & 1 & 1 \\ 2 & 0 & 2 & 0 & -1 & 1 & 1 & -1 \\ -1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\ -3 & 2 & -4 & -2 & 2 & 1 & -1 & 0 \end{bmatrix}
\]

Once the block has been transformed, the coefficients (DC and AC) are normalised and quantised using a normalisation array \(Q(u, v)\). \(Q(u, v)\) contains 8 bit integer values and controls the magnitude of quantisation that each of the transform coefficients is subjected and ultimately controls the compression rate. The values of \(Q(u, v)\) are set by the JPEG standard and there are many available depending on the degree of compression required:

\[
Q(u, v) = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}
\]
Quantisation of $H(u, v)$ leads to a quantised DCT block $\hat{H}^*(u, v)$ given by:

$$\hat{H}^*(u, v) = \text{round}\left(\frac{H(u, v)}{Q(u, v)}\right)$$

Each coefficient in $\hat{H}^*(u, v)$ is divided by the corresponding coefficient in $Q(u, v)$ and rounded to the nearest integer. As a result many of the AC coefficients are set to zero, as evidenced in the current example.

As can be seen, most of the coefficients that are set to zero are in the perceptually least significant middle and high frequencies, which is why for a watermark to be robust it must reside within the perceptually most significant – low frequency areas. Many of the coefficients holding the watermark information could potentially be destroyed by such an attack. JPEG compression is often carried out on digital images to reduce their size prior to use on websites; therefore any watermark algorithm has to be able to withstand such an attack.

### 3.6 Experiment 3: Clipping

Clipping simulates a break apart and recombine attack which is one of the sternest tests a watermark can undergo. A section of watermarked image is first clipped and the rest discarded. The missing parts of the image are recombined from the un-watermarked image to restore the image. Paint Shop Pro 7.04 was used to clip and reassemble the images in this experiment.

A clipping attack has the effect of removing watermark information in all frequencies of the image, low to high. No matter how strongly the watermark is
embedded within the transform domain coefficients clipping will always remove a percentage of the
watermark. Clipping removes the watermark from those frequencies which have been clipped, and
this could be from either the low or high frequency spectrum. If it is a high frequency area then
much of the watermark would not remain in the clipped and reconstructed image. If it is a low
frequency area then much of the watermark may remain.

3.7 Experiment 4: Dithering

Dithering, for this attack, is the reduction of the number of bits
per pixel to 1, black and white. The image detail is provided by
the density of black pixels within a given region. Such an attack
influences the entire frequency space of the image and it will be
interesting to see how the watermark stands up to this attack.
Dithering was performed in Paint Shop Pro 7.04 using the Floyd-
Steinberg dithering algorithm with a non weighted palette. The
Floyd-Steinberg dithering algorithm uses a diffusion error
reduction method, to reduce the colour depth. When the reduction is from 8bpp to 1bpp the effect is
similar to thresholding. Suppose we set the threshold value to be 128; all pixels below that are set to
0 and all those above to 255. A problem arises when a pixel value is very close to the threshold
value, such as 127 or 128. In the first case the pixel would be set to 0, in the second it would be set
to 255; however both pixels would have an error, \( \varepsilon \) of 127. The Floyd-Steinberg error diffusion
algorithm (Efford 2000) distributes this error amongst the neighbouring pixels. Assuming that the
algorithm is scanning left to right and top to bottom, the pixel error is divided thus amongst the
neighbouring pixels:

If \( h(x, y) \) is the pixel undergoing thresholding, then \( h(x+1, y) = h(x+1, y) + 7(\varepsilon)/16 \) and the other
pixels below undergo a similar change. A non-weighted palette means the pixel values are not set
nearer to black or white before dithering.
3.8 Experiment 5: Digital / Analogue – Analogue / Digital Conversion

This is the printing and rescanning of the watermarked image using the same settings for printing and scanning for each test conducted. The watermarked image is printed using an Epson C80 inkjet printer at 1440 dots per inch (dpi) on plain paper and rescanned at 1400 dots per inch (dpi) using a Visioneer 4800 USB scanner. After rescanning the image had to be resized to 512 x 512 pixels; to enable it to be read by the extraction program.

3.9 Experiment 6: Uniqueness of Watermark

This attack tests the watermarking algorithm for false positive and false negative detector responses. The watermarked image is tested in its non-attacked state to determine if the detector is working correctly. This test is valid as many of the images in the Internet domain may not be corrupted and may just have been taken and used in its original watermarked form. The detector has to be able to correctly detect the watermark and prove that it does not suffer from false positive or false negative results.

3.10 Experiment 7: Multiple Watermarks

The image is watermarked with five successive watermarks which has the effect of adding noise to those coefficients which are used to embed the watermarks. Adding multiple watermarks in this way will have a cumulative affect on image quality and fidelity; decreasing both.

This attack aims to break the integrity of the valid watermark by adding noise to those frequencies which contain the watermark. It is unlike collusion in that it has a larger impact on image fidelity.
3.11 Experiment 8: Collusion

The watermarked image is averaged with another anonymously watermarked image to create a single image containing some elements of each of the separately watermarked images.

Such collusion attacks aim to break the integrity of the valid watermark by adding noise to those frequencies which contain the watermark. This attack is useful as the disruption to the watermark comes with little impact on the image fidelity given that the two images being averaged look identical.
4. Results

For each of the experiments performed the first graph represents Scaling Factor ($\alpha$) vs. Normalised Detector Confidence Measure and shows how the scaling factor affects the strength of the watermark. The second graph represents Watermark $W(k)$ Length ($n$) vs. Normalised Detector Confidence Measure and shows how the watermark length (hence the number of coefficients modulated) affects the robustness of the algorithm. For the second graphs the black line represents the experiment conducted with a constant scaling factor value of 0.1, whilst the purple line represents the experiment conducted with a constant scaling factor value of 0.3. This was only carried out for experiments 1, 2 and 4 to check the accuracy of the results obtained using scaling factor = 0.1. The Normalised Detector Confidence Measure has a scale of 0 – 1; 0 corresponding to no correlation between the watermark embedded within the original image and that extracted from the corrupted image. 1 corresponds to a perfect correlation between the watermark embedded within the original image and that extracted from the corrupted image.

First graph for each experiment:

- Scaling Factor ($\alpha$) vs. Normalised Detector Confidence Measure
- Black Line => $n = 1000$

Second graph for each experiment:

- Watermark $W(k)$ Length ($n$) vs. Normalised Detector Confidence Measure
- Black Line => $\alpha = 0.1$
- Purple Line => $\alpha = 0.3$
4.1 Experiment 1: Image Scaling
4.2 Experiment 2: JPEG Compression

4.2.1 10% Quality 0% Smoothing
4.2.2 5% Quality 0% Smoothing

![Graph 1](image1)

![Graph 2](image2)
4.3 Experiment 3: Clipping

![Graph showing normalised confidence vs. scaling factor (alpha)]

![Graph showing normalised confidence vs. watermark length (n)]
4.4 Experiment 4: Dithering

![Graph 1: Scaling Factor vs. Normalised Confidence](image1)

![Graph 2: Watermark Length vs. Normalised Confidence](image2)
4.5 Experiment 5: Digital/Analogue – Analogue/Digital Conversion
4.6 Experiment 6: Uniqueness of Watermark
4.7 Experiment 7: Multiple Watermarks

![Graph 1: Normalised Confidence vs Scaling Factor (Alpha)]

![Graph 2: Normalised Confidence vs Watermark W(K) Length (n)]
4.8 Experiment 8: Collusion
5. Discussion

The aim of this report was not to prove the effectiveness of the spread spectrum watermarking algorithm of Cox (1995); this has already been proved in the original paper. The aim is to determine whether there is an ideal value for the watermark length $n$ and the scaling factor $\alpha$.

**Experiment 1:** Virtually all the applications that use watermarking call for the algorithm to be robust to geometric deformations such as image scaling. The best response from Cox’s algorithm came using an $\alpha$ value of between 0.1 and 0.3. Above this there was little increase in confidence from the detector, and given that an $\alpha$ value above 0.5 produced an image in which the watermark was visible; the optimum appears to be 0.3.

The detector response to variations in the watermark length $n$ was not what was expected. The expected result was that detector confidence would increase gradually as the watermark length increased. In actuality the result showed a peak confidence at approximately $n = 200$, after which confidence dropped off as $n \to 2000$. To confirm this, the test was conducted again with a scaling factor of 0.3, results for which are shown by the purple line.

Scaling the image to 50% of its original size, applying a low pass filter, then rescaling to its original size has the effect of removing all the high frequency transform domain coefficients, and is similar to JPEG compression. Increasing the watermark length $n$ gave an opposite result to the one expected due to how the Cox algorithm is implemented. The watermark length $n$ affects the number of transform domain coefficients that will be altered. As the watermark length increases the watermark begins to spread further into the high frequency transform domain coefficients as all the low frequency coefficients are used up. These are the very coefficients that offer least robustness to attack and can be altered significantly and even set to zero as is the case with the JPEG compression attack. As a result the watermark present in this section of the image spectrum is completely destroyed. When the watermark sequence is extracted from the corrupted image, the algorithm references the unwatermarked image to determine the location of the $n$ highest magnitude transform domain coefficients to embed the watermark. If the watermark is of sufficient length these may fall within the high frequency spectrum. The extraction algorithm references the same coefficients within the watermarked and corrupted images to determine a possibly corrupted watermark; for the high frequency coefficients their values may be completely different. The part of the watermark that is embedded within the high frequency spectrum of the image will give a lower correlation than the part of the watermark that is embedded within the low frequency; thus lowering the overall
confidence measure of the detector. Indeed, as the watermark length increases, the effect will become more pronounced.

**Experiment 2:** JPEG compression is the most common attack on a digital image. Past watermarking schemes such as that by Koch (1995) were designed specifically for robustness to JPEG compression. In general the decoder response was lower for 5% quality than 10% quality due to the nature of JPEG compression. As the compression rate increases, the number of transform domain coefficients which are set to zero also increases, leading to the loss of image quality data and subsequently the embedded watermark.

Detector response to a change in scaling factor increased as $\alpha$ increased for both 10% and 5% quality tests. There was a slower response for the 5% quality test, with an optimum scaling factor of $0.3 - 0.5$.

A change in the watermark length $n$ produced similar results to experiment 1 such that decoder confidence dropped after $n = 200$ for both the 5% and 10% quality test. Given the similarity of the JPEG compression experiment with the image scaling experiment this can be explained by the loss of image quality information from the high frequency spectrum of the image and the subsequent loss of the watermark from within this region.

**Experiment 3:** The clipping attack resulted in the lowest overall response from the detector for all the experiments that were run. A scaling factor of 0.2 gave the best confidence response from the detector, however the results above $\alpha = 0.2$ did not follow the trend of scaling factor increase obtained for the other experiments. Above $\alpha = 0.2$ the confidence of the detector decreased negligibly suggesting that the embedding strength has little effect on the robustness of the algorithm to clipping attacks. A clipping attack has the effect of removing watermark information in all frequencies of the image, low to high. No matter how strongly the watermark is embedded within the transform domain coefficients clipping will always remove a percentage of the watermark.

Similar results were obtained from the detector for the watermark length $n$, which had an optimum value of $n = 200$. After this value the confidence of the detector decreased negligibly until $n = 1600$, when it increased again.

Both the scaling factor and the watermark length results show that neither has much of an effect on the robustness of the algorithm to a clipping attack. Clipping removes the watermark from those
Multimedia Rights Protection – Digital Image Watermarking Techniques

frequencies which have been clipped, and this could be either the low or high frequency spectrum. The confidence of the result depends upon whether the clipped area containing the watermark is a high or low frequency area. If it is a high frequency area then using Cox’s algorithm, much of the watermark would not remain in the clipped and reconstructed image. If it is a low frequency area then much of the watermark may remain; however the frequency of the retained watermarked area of the image also has an effect.

**Experiment 4:** The detector response to the dithering experiment was also unexpected. Dithering is an aggressive attack which reduces the pixel depth to 1 bpp meaning that each pixel can be either black or white. Pixel density provides the image detail rather than a discrete greyscale value.

An optimum scaling factor of 0.2 produced a 90% confidence response from the detector. Using a higher scaling factor produced no significant increase in the confidence measure but did start to degrade the image quality.

An optimum watermark length of 200 produced a confidence measure of approximately 95% after which the confidence of the detector dropped as $n \to 2000$.

**Experiment 5:** The digital to analogue and analogue to digital conversion results were unexpected. The printing of the watermarked images was conducted on ordinary plain paper at 1440 dots per inch (dpi) and rescanning was carried out at a resolution of 1400 dpi.

The confidence of the detector given changes to the scaling factor showed a definite trend, but differed from the results typically seen from earlier experiments. Detector response to the smaller $\alpha$ values was low but increased steadily for the higher values. Despite the fact higher values of $\alpha$ produce visible distortions in the watermarked images; when the images came to be printed the distortions were significantly less noticeable and became hardly visible after rescanning. Therefore, where the image is being watermarked to prevent a digital to analogue and analogue to digital conversion, a higher scaling factor may be used despite the immediate digital form of the image being distorted.

A watermark length of 200 produced the best confidence response from the detector, giving approximately 90%. As the watermark length increased, the confidence measure decreased becoming unstable after $n = 1000$. A printing and rescanning attack will destroy the fine detail within a digital image, much of which will be present within the high frequency spectrum of the
image. This loss of detail will destroy a significant amount of the watermark information present within the high frequency transform domain coefficients. This explains why detector confidence decreases as watermark length increases for this experiment.

**Experiment 6:** The uniqueness of watermark experiment is not really an attack on the image. No signal processing, geometric or collusion attacks are imposed upon the image. The experiment is designed to verify that a watermark can be detected correctly from an uncorrupted copy of the watermarked image.

The results for all values of scaling factor and watermark length produced a 99% confidence or above from the detector. The watermark length results confirm that a longer watermark length does not give any benefit to the robustness of the watermark and is worse for most of the experiments performed. As the watermark length increases the detector has to statistically match two longer watermark vectors. The chances of a match decrease as the vector increases in length, increasing the probability of a false negative.

**Experiment 7:** The multiple watermark attack produced results that were expected for both the scaling factor and watermark length. The increase in detector confidence grew more slowly with an increase in the scaling factor than compared with the other experiments. An optimum confidence value was achieved using a scaling factor of between 0.3 and 0.5

This was one of the few occasions when increasing the watermark length increased the confidence of the detector. Although the optimum watermark length is still 200 to 400, the confidence increased steadily as $n \to 2000$. The multiple watermark attack does not reduce fine detail or destroy image information within the high spectral regions therefore a longer watermark length should increase the robustness of the watermark.

**Experiment 8:** All scaling factor and watermark length values gave a >95% confidence response from the detector. This attack seeks to disrupt the watermark by averaging images together to form a corrupted image. Due to the loss of fidelity at higher scaling factor values, there is no reason to use an $\alpha$ value greater than 0.3 or a watermark length greater than 200.
6. Conclusion

This report has discussed ways in which a digital image can be watermarked using a whole image Direct Cosine Transformation spread spectrum technique to protect the multimedia rights of the owner. Variations in the scaling factor $\alpha$ (embedding strength) and watermark length $n$ have been applied to test the algorithm originally proposed by Cox (1995) and later (Cox 1997).

The results have shown that the scaling factor has a huge effect on the robustness of the algorithm. As the scaling factor (embedding strength) increases so does the robustness against attack. There is a trade off between image quality of the watermarked image and the embedding strength of the watermark. As can be seen in chapter 3 using a watermark length of 1000 values similar to that used by Cox in his original experiments, the image quality of the watermarked image decreases significantly above a scaling factor of 0.4. For most applications this will be unacceptable, breaking the imperceptibility principle that is vital if the algorithm is to be robust. For other applications this is not a problem, especially the digital to analogue and analogue to digital conversion attack, where the image is printed and rescanned. In this case the imperceptibility principle is less important given that much of the loss in quality due to the watermark is masked when the image is printed and subsequently rescanned.

Experiments into the variation of the watermark length $n$ showed completely the opposite of what was expected. When an image attack involves the loss of fine detail within the image, such as JPEG compression or image scaling, a longer watermark appears to be less robust than if a shorter one were used. Paradoxically, with the detector used here a 90% confidence achieved using a watermark length of 200 is better than a 40% confidence in a watermark length of 1000. Yet more watermarked coefficients may have been detected in the latter case. In the paper by Cox (1997) they state “…longer watermarks may be used for an image that is especially sensitive to large modifications of its spectral components…” This statement, as this report has proved, is not the case. Large modifications to the spectral components were produced in many of the experiments and all proved that in such cases a longer watermark is actually less robust.

The best solution to protect multimedia rights through digital image watermarking is three-fold. The image must be watermarked using a secure, robust and imperceptible transform domain algorithm such as the Cox algorithm tested here. It has been shown that using a scaling factor of 0.3 and a watermark length of 200, results in the optimum performance of the algorithm against attack. To complement this, if the image is being distributed over the Internet, or any unsafe network, the
image should be RSA encrypted (Litterio 1999) to add a further layer of security. Finally, if the image is to be published to the World Wide Web then the use of a small embossed logo discreetly positioned within the image would complete the scheme.
7. Future Work

There is huge scope for future work within the field of digital image watermarking. As Internet use grows and multimedia content becomes richer and more varied there will be demand for a secure, robust and easy to implement technique for protecting the multimedia copyright of the rightful owner.

Of the work that has been carried out in this report there are numerous paths down which future research could go. One such area is the addition of Cox’s algorithm and Watson’s model of the Human Visual System to develop a hybrid model. This model could dynamically calculate the watermark capacity of each transform domain coefficient and modulate the scaling factor $\alpha$ to adjust the embedding strength for each individual coefficient. This would increase the capacity of the image to accept watermark energy and at the same time make it more robust to attack.

The use of the Discrete Cosine Transform to convert from the spatial domain to the frequency domain is just one of many such transform methods. The Discrete Fourier Transform and the Discrete Wavelet Transform are two other popular methods which have been popular for many years. The Discrete Wavelet Transform (DWT) is used in the JPEG2000 standard is becoming more popular within the watermarking community, offering several advantages over the DCT. The DWT is closer to the HVS than the DCT and does not suffer block shaped artefacts which occur within JPEG compressed images. DWT splits image features such as textures and edges more efficiently into the transform domain. This can be more readily exploited than with DCT meaning more watermark energy can be placed within these regions without affecting the quality of the image.

Blind watermarking, where no reference to the original image is made during the detection stage of the algorithm is another area where research will focus. Algorithms such as that by Barni (1998) combine the advantages of spread spectrum with a simple model of the HVS to implement a blind watermarking algorithm.

Fragile watermarking will become important as the media driven world intensifies and the competition to publish an exclusive story and photograph increases. Fragile watermarking will prevent publications from tampering with digital photographs to increase the impact of their stories.
8. Evaluation of Objectives

To understand the relevant image-processing theory

The report has studied in detail the general principles of watermarking, starting with an explanation of the make up of digital images, the use of the DCT to transform the spatial image into the frequency domain, the methods used to embed and extract the watermark and the statistical methods used to compare them.

To review cutting edge developments in digital image watermarking and report on their effectiveness.

The report has reviewed the algorithms by Cox (1995) and Watson (1993) and tested their effectiveness. The algorithm of Cox was tested by varying the scaling factor and watermark length to determine the effect on the robustness of the algorithm.

To implement an effective digital watermarking technique on a set of reference images.

The report implemented the algorithm by Cox to determine what effect the watermark embedding strength and watermark length has on the robustness of the algorithm. The software is attributable to Meerwald (2001) but has been extensively studied by the author so that it could be re-made for the Linux platform.

To suggest new ways in which digital image rights protection could be implemented, either through a new digital image watermarking scheme, or some other method.

The report has determined the optimum performance of Cox’s algorithm and suggested that it could be adapted further using a dynamic system for varying the watermark embedding strength by exploiting features of the HVS. The report has suggested a complete solution to the problem of multimedia rights protection by combining the optimised algorithm by Cox in combination with the RSA encryption and a small visible copyright logo positioned discretely within the image. The RSA encryption algorithm would protect the image between owner and purchaser, whilst the watermarking scheme would protect the image after it has been unencrypted. The logo is optional but it is a belief of the author that such a feature may prevent casual unauthorised use of the image. It will not prevent a determined attack; this is the job of the watermark.
9. References


Appendix A – Personal Experience

This MSc project has been an invaluable lesson in conducting a research project, from the planning and implementation to the research and reading techniques needed to study the topic. Many of these techniques I will take with me in my future career.

The assessors’ comments regarding my interim report suggested that my literature review was too shallow and superficial. It stated that the report should either go for depth in a narrow field or breadth with less detail. I decided to specifically address this complaint by discussing in great detail the general principles of digital watermarking, and also some of the industry standard algorithms and believe I have answered those criticisms.

The project schedule is an important tool for project management, allowing a project plan to be implemented. I would advise future MSc students not to strictly adhere to the timetable set out in the schedule. Keep your plans flexible because unexpected problems can crop up along the way. I found that starting the write up early made the project easier. I was able to write down ideas I had at the time rather than leaving them until the last minute.

I am pleased with the outcome of the report and have learned a great deal about digital image watermarking and image compression standards such as the JPEG compression algorithm and the Human Visual System. Before the project I had no regard for the size of this area of research and have learned that it has a huge following worldwide. I am pleased I have managed to implement and suggest new ways in which existing algorithms can be used to achieve the highest robustness against attack. Not only have I managed this but I have also managed to explain why these settings work and why others do not.

Citation of respected work within the area of research that you are studying is good practise. Not only does it show that you have read and understood the subject, it also helps the reader understand where your arguments are coming from. I would encourage students to make use of existing source code available within the area of research that they are studying. If source code exists, do not be afraid to use it to achieve your goals; there is no point in reinventing the wheel. I found that studying what the original software engineer had done is just as rewarding as if I had written the software myself. Please remember to cite and give credit to the original source code.
The MSc project log is essential in the early stages of the project, but became an annoying distraction during the latter stages of the project, especially when the report was being written up. I would definitely recommend that students keep up to date with their project logs. If references are kept and documented in the log as the project is progressing then a lot of time is saved when the report comes to be written up.

Finally, don’t be afraid to challenge the views of respected researchers within your field of study but you must have valid and coherent arguments to back up your claims. I managed to discover that using a watermark of greater length actually resulted in a less robust algorithm; however I also stated why I thought this was so.
Appendix B – Objectives and Deliverables

School of Computing, University of Leeds
MSC PROJECT OBJECTIVES AND DELIVERABLES

This form must be completed by the student, with the agreement of the supervisor of each project, and submitted to the MSc project co-ordinator (Mrs A. Roberts) via CSO by 21st March 2002. A copy should be given to the supervisor and a copy retained by the student. Amendments to the agreed objectives and deliverables may be made by agreement between the student and the supervisor during the project. Any such revision should be noted on this form. At the end of the project, a copy of this form must be included in the Project Report as an Appendix.

Student: _______________________________________
Programme of Study: _______________________________________
Supervisor: _______________________________________
Title of project: ______________________________________
External Organisation*: _______________________________________
*(if applicable)

AGREED MARKING SCHEME

<table>
<thead>
<tr>
<th>Understand the Problem</th>
<th>Produce a Solution *</th>
<th>Evaluation</th>
<th>Write-Up</th>
<th>Appendix A</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>40</td>
<td>20</td>
<td>15</td>
<td>5</td>
<td>100</td>
</tr>
</tbody>
</table>

* This category includes Professionalism

OVERALL OBJECTIVES:

1. To understand the relevant image-processing theory
2. To review cutting edge developments in digital image watermarking and report on their effectiveness.
3. To implement an effective digital image watermarking technique on a set of reference images.
4. To suggest new ways in which digital image rights protection could be implemented, either through a new digital image watermarking scheme, or some other method.

MINIMUM REQUIREMENTS:

1. To understand the relevant image-processing theory
2. To review cutting edge developments in digital image watermarking and report on their effectiveness.
3. To construct a basic demonstrator of a digital water-marking system

SOFTWARE AND HARDWARE RESOURCES REQUIRED:
The resources I envisage I will need for this project are all available within the university as of Tuesday 19th March 2002. I will request resources where needed as the project progresses.

DELIVERABLE(s):

1. A project report.
2. A demonstrator

Signature of student:  
Signature of supervisor:  
Date: Tuesday 19th March 2002  
Date: Tuesday 19th March 2002

Amendments to agreed objectives and deliverables:
Date Amendment

NOTE:
Students should discuss the requirements of the form with their supervisor prior to submission but it is the student's responsibility to submit it by the due date (not the supervisor's). Before submitting this form it is important that student and supervisor agree that the objectives, minimum requirements, and computer requirements are feasible.

The overall objectives of the project are the overall top-level goals. The project title should reflect these objectives.
The minimum requirements are components of the deliverables that constitute minimal acceptable work in producing a solution to the problem. They should be described in a quantifiable way, and in a way which gives the greatest opportunity to expand upon them and/or deliver them to a higher standard to gain extra credit. (At the mid-project stage you will be required to describe any enhancements that you intend delivering if time permitted or, failing that, you would recommend as future work in this area.). Note that the marks available for delivery of a solution relate specifically to your achievements in relation to the agreed minimum requirements.
Wherever possible we encourage students to use the software and facilities currently provided in the school. Whilst students are welcome to develop software on their home machines, they WILL have to demonstrate it to their supervisor and assessor at a progress meeting within the school. The issue of availability of facilities and the behaviour of software on school machines MUST be addressed at the outset. For instance, our security arrangements may prohibit the execution of programs and this will not be compromised to accommodate a demonstration. Students and supervisors should discuss computer requirements. Requirements for software and tools which are not currently provided by the school have to have a technical justification and sensible requests will be considered. If you are in any doubt about current provisions or of your ultimate software needs please submit the request anyway. If you are doing a background research phase you should state what software you will be considering and give an estimated date by which you will have decided what you would prefer to use. Provide as much information about the request as possible e.g., operating system to work on, version to install, accessibility needs. The school does not guarantee to honour requests for additional facilities.
Appendix C – Interim Report Header Sheet

School of Computing, University of Leeds
MSC PROJECT INTERIM REPORT

All MSc students must submit an interim report on their project to the MSc project co-ordinator (Mrs A. Roberts) via CSO by Wednesday 8th May 2002. Note that it may require two or three iterations to agree a suitable report with your supervisor, so you should let him/her have an initial draft well in advance of the deadline. The report should be a maximum of 10 pages long and be attached to this header sheet. It should include:

- the objectives, deliverables and agreed marking scheme
- resources required
- progress report and project schedule
- proposed research methods
- a draft chapter on the literature review and/or an evaluation of tools/techniques
- the WWW document link for the project log to date

The report will be commented upon both by the supervisor and the assessor in order to provide you with feedback on your approach and progress so far.

The submission of this Interim Report is a pre-requisite for proceeding to the main phase of the project.

<table>
<thead>
<tr>
<th>Student:</th>
<th>James Padgett</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programme of Study:</td>
<td>DMS</td>
</tr>
<tr>
<td>Title of project:</td>
<td>Multimedia Rights Protection – Digital Image Watermarking Techniques</td>
</tr>
<tr>
<td>Supervisor:</td>
<td>Bill Whyte</td>
</tr>
<tr>
<td>External Company (if appropriate):</td>
<td>n/a</td>
</tr>
</tbody>
</table>

**AGREED MARKING SCHEME**

<table>
<thead>
<tr>
<th>Understand the problem</th>
<th>Produce a solution *</th>
<th>Evaluation</th>
<th>Write-up</th>
<th>Appendix A</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>40</td>
<td>20</td>
<td>15</td>
<td>5</td>
<td>100</td>
</tr>
</tbody>
</table>

* This includes professionalism

Signature of student:  
Date: Friday 3rd May 2002
Supervisor's comments on the Interim Report

I believe you are making good initial progress but note the assessor's comment which you should explicitly address in the final report (e.g. as a "mid-stage review" in your project management section).

Assessor's comments on the Interim Report

Could be an interesting project.

I'm not sure that you've allowed enough time in the schedule for implementing your demonstrator.

Your literature review is far too short and superficial! You should have either gone for breadth, briefly discussing many more references to give an overview of the field, or gone for depth, discussing in much greater detail your existing sources to demonstrate that you understand the theory behind watermarking.