Capítulo 2

Esteganografia e suas Aplicações

Eduardo Pagani Julio, Wagner Gaspar Brazil, Célio Vinicius Neves Albuquerque

Universidade Federal Fluminense
Departamento de Ciência da Computação - Centro Tecnológico
{ejulio,wbrazil,celio}@ic.uff.br

Abstract

Steganography derives from the greek words stegano and graphy, where stegano means to hide, mask and graphy means to write. So, steganography is the art of cover writing. Along history, people has tried various forms to hide information within various media, searching in some form, to provide more privacy to their comunications. Some usual approaches to inserting messages into images include techniques such as: overwritting the least significant bit, as well as filtering, masquerading and and transformation algorithms. Each of these techniques can be applied to images with different levels of success. The goal of this course is to explore some steganography techniques since these techniques can be used to protect communications. Besides covering well-known techniques, we intend to show some of the applications and the applicability of steganography as an alternative to cryptographic methods.

Resumo

Esteganografia deriva do grego, onde estegano significa esconder, mascarar e grafia significa escrita. Logo, esteganografia é a arte da escrita encoberta. Durante toda a história, as pessoas buscam inúmeras formas de esconder informações dentro de outros meios, para, de alguma forma, obter mais privacidade para seus meios de comunicação. As abordagens mais comuns de inserção de mensagens em imagens incluem técnicas de: inserção no bit menos significativo, filtragem e mascaramento e algoritmos de transformações. Cada uma destas técnicas pode ser aplicada a imagens, com graus variados de sucesso. O objetivo deste curso é explorar as técnicas de esteganografia de maneira que

possam ser usadas na proteção das comunicações. Além disso, deseja-se mostrar as aplicações e a aplicabilidade da esteganografia como uma opção aos métodos de criptografia mais conhecidos.

2.1. Introdução

A segurança digital é uma área com grande potencial para pesquisa e desenvolvimento. Sistemas de detecção de intrusão, anti-vírus, *proxies* e *firewalls* ultimamente aparecem muito na mídia em geral e estão se tornando ferramentas de uso doméstico. É cada vez maior o número de pessoas que tentam a todo custo ludibriar as defesas para ter acesso a um dos bens mais preciosos da sociedade moderna: a informação. Por outro lado, existem outras pessoas que buscam o desenvolvimento e o estudo de técnicas para proteção das comunicações. As ferramentas e técnicas que provêem a segurança da informação são inúmeras e a criptografia está entre elas há milhares de anos.

Um dos ramos da criptografia é a esteganografia. De origem grega, a palavra significa a arte da escrita escondida (estegano = esconder e grafia = escrita). A esteganálise por sua vez é a arte de detectar mensagens escondidas nos mais diversos meios de comunicação. A esteganografia inclui um amplo conjunto de métodos e técnicas para prover comunicações secretas desenvolvidos ao longo da história. Dentre as técnicas se destacam: tintas invisíveis, micropontos, arranjo de caracteres (*character arrangement*), assinaturas digitais e canais escondidos (*covert channels*) (PETITCOLAS; ANDERSON; KUHN, 1999) (PETITCOLAS; KATZENBEISSER, 1999) (JOHNSON; JAJODIA, 1998).

As aplicações de esteganografia incluem identicação de componentes dentro de um subconjunto de dados, legendagem (*captioning*), rastreamento de documentos e certificação digital (*time-stamping*) e demonstração de que um conteúdo original não foi alterado (*tamper-proofing*). Entretanto, como qualquer técnica, a esteganografia pode ser usada correta ou incorretamente. Há indícios recentes de que a esteganografia tem sido utilizada para divulgar imagens de pornografia infantil na Internet (MORRIS, 2004) (HART; ASHCROFT; DANIELS, 2004), além das mensagens de redes terroristas como a Al-Qaeda.

2.1.1. Terminologia

Há um interesse cada vez maior, por diferentes comunidades de pesquisa, no campo da esteganografia, marcas dágua e seriação digitais. Com certeza, isso leva a uma certa confusão na terminologia. A seguir, encontram-se alguns dos principais termos utilizados nestas áreas e ilustrados na Figura 2.1:

- dado embutido ou *embedded data* - é o dado que será enviado de maneira secreta, normalmente em uma mensagem, texto ou figura;
- mensagem de cobertura ou *cover-message* - é a mensagem que servirá para mascarar o dado embutido. Esta mensagem de cobertura pode ser de áudio (*cover-audio*), de texto (*cover-text*) ou uma imagem (*cover-image*);
- estego-objeto ou *stego-object* - após a inserção do dado embutido na mensagem de cobertura se obtém o estego-objeto;

- estego-chave ou *stego-key* - adicionalmente pode ser usada uma chave para se inserir os dados do dado embutido na mensagem de cobertura. A esta chave dá-se o nome de estego-chave;

- número de série digital ou marca *fingerprinting* - consiste em uma série de números embutidos no material que será protegido a fim de provar a autoria do documento.

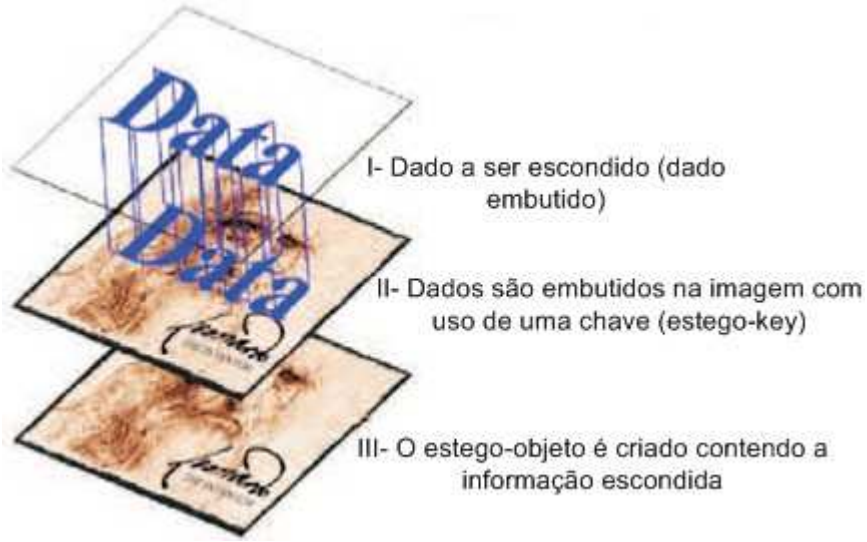


Figura 2.1: Escondendo uma imagem (PETITCOLAS; ANDERSON; KUHN, 1999).

A esteganografia pode ser dividida em dois tipos: técnica e lingüística. O primeiro tipo se refere às técnicas utilizadas quando a mensagem é fisicamente escondida, como por exemplo escrever uma mensagem em uma tábua de madeira e cobri-la com cera, como faziam alguns povos na antigüidade. A esteganografia lingüística se refere ao conjunto de técnicas que se utilizam de propriedades lingüísticas para esconder a informação, como por exemplo *spams* e imagens.

Os sistemas de marcação visam proteger a propriedade intelectual sobre algum tipo de mídia (eletrônica ou não). Estes sistemas de marcação são conhecidos também como *watermarking* (marca d'água). Apesar de aparecerem quase sempre em conjunto com esteganografia, os sistemas de marcação não pertencem ao ramo da esteganografia. Ambos pertencem a uma área de pesquisa conhecida como **ocultamento da informação** ou ***information hiding***.

O sistema de marcação tipo marca d'água se refere a métodos que escondem informações em objetos que são robustos e resistentes a modificações. Neste sentido seria impossível remover uma marca d'água de um objeto sem alterar a qualidade visual do mesmo. Por outro lado a esteganografia se propõe a esconder uma informação em uma imagem de cobertura. Se a imagem for destruída ou afetada a mensagem é perdida. Uma outra diferença clara entre esteganografia e técnicas de marca d'água é que enquanto o dado embutido da esteganografia nunca deve ficar aprente, a marca d'água pode ou não aparecer no objeto marcado, dependendo da aplicação que se queira atender.

Neste sentido pode-se classificar os sistemas de marcação segundo de acordo com a sua robustez e a sua aparência. Segundo sua robustez, podem ser classificados como:

- **robustos** - são aqueles em que mesmo após a tentativa de remoção a marca permanece intacta;
- **frágeis** - são os sistemas em que qualquer tentativa de modificação na mídia acarreta a perda da marcação. É muito útil para verificação de cópias ilegais. Quando se copia um objeto original, a cópia é feita sem a marca.

Já quanto a sua aparência, os sistemas de marcação podem ser classificados como:

- **de marcação imperceptível** - são os sistemas onde a marca encontra-se no objeto ou material, porém não é visível diretamente;
- **de marcação visível** - neste sistema a marca do autor deve ficar visível para comprovar a autoria visualmente. Um bom exemplo deste sistema são as marcas d´água em cédulas de dinheiro e em selos.

2.1.2. Aspectos Históricos

A esteganografia é uma arte antiga. Suas origens remontam à antiguidade. Os gregos já a utilizavam para enviar mensagens em tempos de guerra (KAHN, 1996). Nas "Estórias de Herodotus", existem muitas passagens mostrando o uso da esteganografia. Em uma estória, um mensageiro se disfarçou de caçador para enviar uma mensagem ao rei escondendo-a dentro de uma lebre. Como o mensageiro estava disfarçado, passou despercebido pelos portões do palácio e o rei pôde receber a mensagem.

Mensagens também foram enviadas através de escravos de confiança. Alguns reis raspavam as cabeças de escravos e tatuavam as mensagens nelas. Depois que o cabelo crescesse, o rei mandava o escravo pessoalmente com a mensagem (KAHN, 1996). Ninguém suspeitaria onde a mensagem se encontrava, a menos que soubesse exatamente onde procurar. Neste caso o segredo com a localização da mensagem deveria ser mantido. Outro exemplo de esteganografia na Grécia antiga era furar buracos em livros acima das letras que formavam a mensagem desejada. Quando o destinatário recebesse o livro poderia procurar pelos buracos sobre as letras para reconstruir as mensagens. Para quem não soubesse do código, o livro pareceria ter apenas seu conteúdo escrito pelo autor.

Os chineses e egípcios também criaram seus métodos de esteganografia na idade antiga. Os chineses escreviam mensagens em finas folhas de papel de seda que eram depois enroladas como uma bola e cobertos com cera. Esta bola era então escondida em algum lugar do corpo ou engolida para prevenir sua detecção. Os egípcios usavam ilustrações para cobrir as mensagens escondidas. O método de escrita egípcio conhecido como hieróglifo era uma técnica comum para esconder mensagens. Quando um mensageiro egípcio era pego com um hieróglifo que continha algum código, o inimigo não suspeitava e a mensagem podia ser entregue sem problemas ao destinatário.

Durante a idade média, a esteganografia foi mais estudada e desenvolvida. Em 1499, um monge chamado Tritheimius escreveu uma série de livros chamados "Steganographia" (Figura 2.2) nos quais ele descreveu várias técnicas diferentes. Uma delas, desenvolvida na idade média, foi a grade de Cardano (KAHN, 1996). Criada por Girolamo Cardano, a grade era uma lâmina que randomicamente definia retângulos. A quantidade e o posicionamento dos retângulos era o segredo da grade. O remetente escrevia as palavras da mensagem secreta nos retângulos. Depois a grade era removida e o remetente preenchia os espaços remanescentes com letras ou palavras para criar a mensagem que seria enviada (mensagem de cobertura). Uma vez entregue a mensagem, o destinatário colocaria a grade, que era a mesma do emissor, sobre o papel ou superfície que continha a mensagem e podia lê-la sem problemas, lendo os caracteres que estariam dentro dos retângulos.

Figura 2.2: Exemplar de *"Schola Steganographica"* publicado em 1680 (PETITCOLAS; KATZENBEISSER, 1999).

Os primeiros experimentos com tintas invisíveis também começaram na idade média. Giovanni Porta escreveu vários livros de história natural. Dentro destes livros estavam receitas de tintas secretas que poderiam ser usadas para escrever sobre a pele humana e outras superfícies. Este tipo de tinta foi desenvolvido e usado mais tarde no fim dos anos de 1700 e foi a chave para comunicações secretas.

Tintas invisíveis também foram muito usadas em esteganografia nos tempos mais modernos e são utilizadas até hoje. Estas tintas foram utilizadas por espiões durante a primeira e a segunda grande guerra com o desenvolvimento de reagentes químicos específicos para cada tinta. Textos eram escritos em jornais, revistas ou livros com tintas invisíveis para serem passados de forma segura até seus destinatários. Uma outra utilização era escrever a mensagem com tinta invisível sobre um papel, cortá-lo em alguns pedaços e depois rejuntá-los no destinatário (KAHN, 1996).

Outros métodos modernos de esteganografia incluem cifradores nulos e micro pontos. Cifradores nulos são mensagens nas quais certas letras devem ser usadas para formar a mensagem e todas as outras palavras ou letras são consideradas nulas. Para o uso do cifrador nulo, ambos os lados da comunicação devem usar o mesmo protocolo de uso das letras que formam a mensagem. Por exemplo, usar sempre a primeira letra de cada palavra para compor a mensagem. Este método é difícil de implementar, pois a mensagem de cobertura deve ter algum sentido, do contrário um inimigo desconfiará e

quebrará o código. Um exemplo de um código utilizando cifrador nulo é mostrado abaixo (JOHNSON, 1998).

"News Eight Weather: tonight increasing snow. Unexpected precipitation smothers eastern towns. Be extremely cautious and use snowtires especially heading east. The highways are knowingly slippery. Highway evacuation is suspected. Police report emergencies in downtown ending near Tuesday".

Usando as primeiras letras de cada palavra o texto que aparece é:

"Newt is upset because he thinks he is president".

A técnica de mMicro-pontos é também uma outra forma de esteganografia usada atualmente. Um micro-ponto é uma fotografia da mensagem secreta que deve ser entregue. Com a tecnologia avançando rapidamente, é possível tirar uma foto de uma mensagem e reduzi-la a uma fotografia circular de 0,05 polegadas ou 0,125 cm de diâmetro. Esta minúscula fotografia é então colada em um sinal de pontuação de uma frase ou no "pingo"de uma letra "i"de uma outra mensagem qualquer que será entregue. Somente aqueles que sabem onde procurar o micro-ponto poderão detectar sua presença.

Atualmente, novas técnicas de esteganografia são produzidas para serem utilizadas nos novos meios de comunicação. Por exemplo, hoje em dia muitos artistas e gravadoras estão utilizando a marca d'água para proteger suas obras. Com o crescente aumento da pirataria e de *sites* na Internet onde se pode baixar filmes, músicas e vídeos, esta técnica tem se mostrado uma aliada na proteção dos direitos autorais. O uso de esteganografia em software tem um grande potencial, pois pode esconder dados em uma infinidade de mídias. Nas técnicas que utilizam o último bit de um byte para esconder mensagens, uma mensagem de 64Kbytes pode ser escondida em uma figura de 1024 x 1024 em tons de cinza ou imagens coloridas. Esta e outras novas técnicas, representam o estado da arte da esteganografia atual e são apresentadas a seguir.

2.2. Estado da Arte

As imagens são a mídia de cobertura mais popular para esteganografia e podem ser armazenadas em um formato bitmap direto (como BMP) ou em um formato comprimido (como JPEG). Imagens de palheta de cores estão normalmente no formato GIF. O ocultamento de informações é realizado ou no domínio espacial ou no domínio de freqüência. Em termos de esquemas de inserção, vários métodos (como substituição, adição e ajuste) podem ser usados. Uma abordagem de ajuste é a QIM (*Quantization Index Modulation*), que usa diferentes quantizadores para transportar diferentes bits dos dados secretos (SULLIVAN et al., 2004).

As abordagens mais comuns de inserção de mensagens em imagens incluem técnicas de insercão no bit menos signicativo, técnicas de filtragem e mascaramento e algoritmos e transformações. Cada uma destas técnicas pode ser aplicada à imagens, com graus variados de sucesso. O metodo de inserção no bit menos signicativo é provavelmente uma das melhores técnicas de esteganografia em imagem (PETITCOLAS; ANDERSON; KUHN, 1999) (WAYNER, 2002).

2.2.1. Requisitos para Sistemas Esteganográficos

Os três requisitos mais importantes que devem ser satisfeitos para qualquer sistema esteganográfico são:

- segurança - a fim de não levantar suspeita, enquanto tenta criar uma blindagem contra um algorítmo de descoberta, o conteúdo escondido deve ser invisível tanto perceptivelmente quanto por meios estatísticos (BUCCIGROSSI; SIMONCELLI, 1999). Algumas definições baseadas em informações teóricas para um sistema seguro perfeito assumem conhecimento detalhado das estatísticas da cobertura e exigem recursos computacionais ilimitados. Estas condições não são estritamente encontradas em aplicações esteganográficas reais. Por exemplo, relativo a conhecimento estatístico, pode-se estimar estatísticamente um conjunto particular de sinais freqüentemente utilizados por um certo grupo de pessoas e estabelecer um modelo para descoberta. Mas tais modelos não tem sentido se o erro de estimação excede a extensão de modificações causadas por inclusão. Além disso, a complexidade computacional de qualquer ferramenta de esteganografia útil não pode ser infinitamente grande. Em termos de praticidade, um sistema pode ser considerado seguro, ou esteganograficamente forte (DUDA; HART; STORK, 2000), se não for possível descobrir a presença de stego-conteúdo usando qualquer meio acessível;

- carga útil - diferentemente de marca d'água, que precisa embutir somente uma quantia pequena de informações de direitos autorais, a esteganografia é direcionada à comunicação escondida e portanto normalmente exige capacidade de inclusão suficiente. Os requisitos para capacidade significativa de dados e segurança são freqüentemente contraditórios. Dependendo dos argumentos de aplicação específica, um compromisso deve ser buscado;

- robustez - embora robustez contra ataques não seja uma prioridade importante, como em marcas d'água, ter a capacidade de resistir a compressão é certamente desejável, pois a maioria das imagens JPEG coloridas são comprimidas antes de serem colocadas on-line.

2.2.2. LSB

Estas técnicas são baseadas na modificação dos bits menos significativos (*Least Significant Bit*) dos valores de pixel no domínio espacial. Em uma implementação básica, estes pixels substituem o plano LSB inteiro com o stego-dados. Com esquemas mais sofisticados em que locais de inclusão são adaptativamente selecionados, dependendo de características da visão humana, até uma pequena distorção é aceitavel. Em geral, a inclusão de LSB simples é suscetível a processamento de imagem, especialmente a compressão sem perda.

Técnicas baseadas em LSB podem ser aplicadas a cada *pixel* de uma imagem codificada em 32bits por *pixel*. Estas imagens possuem seus pixels codificados em quatro bytes. Um para o canal alfa (alpha transparency), outro para o vermelho (red), outro para o verde (green) e outro para o azul (blue). Seguramente, pode-se selecionar um bit (o menos signicativo) em cada byte do pixel para representar o bit a ser escondido sem

causar alteracões perceptíveis na imagem. Estas técnicas constituem a forma de mascaramento em imagens mais difícil de ser detectada pois podem inserir dados em pixels não sequenciais, tornando complexa a detecção (POPA, 1998) (PETITCOLAS; ANDERSON; KUHN, 1999) (WAYNER, 2002).

2.2.3. Filtragem e Mascaramento

As técnicas de esteganografia baseadas em filtragem e mascaramento são mais robustas que a insercão LSB. Estas geram estego-imagens imunes a compressão e recorte. No entanto, são técnicas mais propensas a detecção (WAYNER, 2002). Ao contrário da inserção no canal LSB, as técnicas de filtragem e mascaramento trabalham com modificacões nos bits mais significativos das imagens. As imagens de cobertura devem ser em tons de cinza porque estas técnicas não são eficazes em imagens coloridas (POPA, 1998). Isto deve-se ao fato de que modificacões em bits mais significativos de imagens em cores geram muitos artefatos tornando as informações mais propensas a detecção.

Estas técnicas são semelhantes a marca d'água visível em que valores de pixel em áreas mascaradas são aumentados ou diminuídos por um pouco de porcentagem. Reduzindo o incremento por um certo grau faz a marca invisível. No método de retalhos (*patchwork*), pares de remendos (*patches*) são selecionados pseudo-aleatoriamente. Os valores de pixel em cada par são aumentados por um valor constante pequeno em um remendo e diminuídos pela mesma quantia no outro.

2.2.4. Algoritmos e Transformações

As técnicas de esteganografia baseadas em algoritmos e transformações conseguem tirar proveito de um dos principais problemas da inserção no canal LSB que é a compressão. Para isso são utilizadas: a transformada de Fourier discreta, a transformada de cosseno discreta e a transformada Z (GONZALEZ; WOODS, 2002).

Sendo embutido no domínio de transformação, os dados escondidos residem em áreas mais robustas, espalhadas através da imagem inteira e fornecem melhor resistência contra processamento de sinal. Configuram-se como as mais sosticadas técnicas de mascaramento de informações conhecidas (POPA, 1998), embora sofisticacão nem sempre implique em maior robustez aos ataques de esteganálise. A inclusão de dados apresentados no domínio de transformação é amplamente usada para marca d'água robusta.

De forma geral, estas técnicas baseadas em algoritmos e transformações aplicam uma determinada transformação em blocos de 8x8 pixels na imagem. Em cada bloco, devem ser selecionados os coeficientes que são redundantes ou de menor importância. Posteriormente, estes coeficientes são utilizados para atribuir a mensagem a ser escondida em um processo em que cada coeficiente é substituído por um valor pré-determinado para o bit 0 ou 1 (POPA, 1998).

Para melhor entendimento do funcionamento destas técnicas, é explicada a seguir a transformada de cosseno discreta (DCT) que é muito utilizada nas compressões dos padrões JPEG e MPEG.

Transformada de Cosseno Discreta

A transformada de cosseno discreta (DCT - *Discrete Cosine Transform*) é uma transformada matemática baseada em cossenos, muito utilizada em processamento digital de imagens e compressão de dados. O valor da função da DCT de um vetor *p* de *pixels* de comprimento *n* é:

$$G_f = \frac{1}{2} C_f \sum_{t=0}^{n-1} p_t \cos\left(\frac{(2t+1)f\pi}{2n}\right), \tag{2.1}$$

$$onde: C_f = \begin{cases} \frac{1}{\sqrt{2}}, & f = 0 \\ 1 & f > 0 \end{cases} \text{para } f = 0, 1, ..., n-1.$$

A matriz dessa transformada é composta de vetores ortonormais, sendo por isso uma matriz de rotação. Na compressão de dados, esta transformada é muito utilizada pois transfere a maior parte da informação contida para os primeiros elementos do vetor, otimizando o armazenamento (para compressão sem perdas) e facilitando a quantização dos valores (para compressão com perdas), conforme mostrado na Figura 2.3.

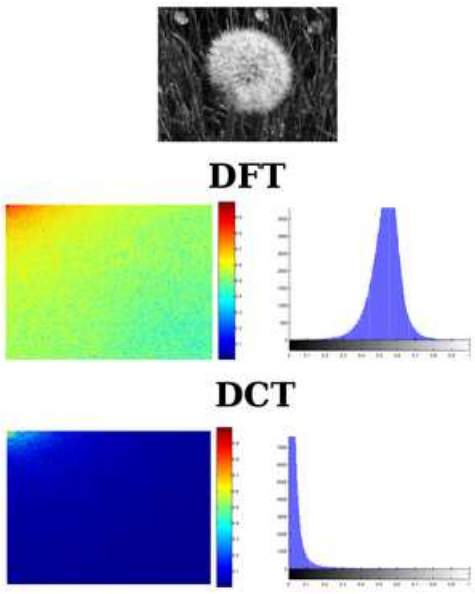


Figura 2.3: Comparação entre a Transformada de Fourier discreta e a DCT: pode se observar o acúmulo dos coeficientes mais significativos no canto superior direito da imagem da DCT, proporcionando melhor capacidade de compressão.

A recuperação dos dados transformados pode ser feita com a operação inversa, chamada de IDCT (*Inverse Discrete Cosine Transform*), que é dada pela fórmula:

$$p_t = \frac{1}{2} \sum_{j=0}^{n-1} C_f G_j \cos\left(\frac{(2t+1)j\pi}{2n}\right), \text{para } t = 0, 1, ..., n-1. \tag{2.2}$$

Em compressão de imagens e vídeos a maioria dos padrões usa a transformada discreta de cosseno do vetor *p* com o tamanho *n* = 8 (JPEG e MPEG).

Sabendo que os pixels de uma imagem tem correlação com seus vizinhos nas duas dimensões da imagem, e não apenas em uma dimensão, a DCT para ser usada na compressão de imagens também deve ser uma transformada bidimensional. A fórmula para uma matriz (ou seja uma imagem) *p* de tamanho *n* x *n* é:

$$G_{ij} = \frac{1}{\sqrt{2n}} C_i C_j \sum_{x=0}^{n-1} \sum y = 0^{n-1} p_{xy} \cos\left(\frac{(2y+1)j\pi}{2n}\right) \cos\left(\frac{(2x+1)i\pi}{2n}\right), \text{para } 0 \leq i, j \leq n-1, \quad (2.3)$$

$$onde\, C_f = \begin{cases} \frac{1}{\sqrt{2}}, & f = 0 \\ 1, & f > 0 \end{cases}.$$

Essa transformada pode ser considerada como uma rotação (ou duas rotações consecutivas, uma em cada dimensão), ou ainda como uma base ortogonal em um espaço vetorial de *n* dimensões. A recuperação dos dados transformados pode ser feita usando a transformação inversa, conhecida como IDCT bidimensional:

$$p_{xy} = \frac{1}{4} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \cos\left(\frac{(2x+1)i\pi}{2n}\right) \cos\left(\frac{(2y+1)j\pi}{2n}\right). \quad (2.4)$$

Analogamente à transformada unidimensional, a transformada bidimensional resulta em uma matriz onde os coeficientes mais significativos se acumulam no canto superior esquerdo (início da matriz) e os demais coeficientes são de pequeno valor podendo ser mais facilmente armazenados ou mesmo quantizados para proporcionar uma compressão com perdas.

Apesar de serem relativamente fáceis de implementar em qualquer linguagem de programação, a compressão de imagens demanda um grande poder de processamento e por isso precisa ser otimizada ao máximo. O uso da DCT em imagens grandes, apesar de apresentar ótimos resultados, exige um processamento muito grande. Por isso na prática a estratégia que se adota é dividir a imagem em blocos de tamanho menor (em geral de tamanho 8x8 pixels, como no JPEG), levando a uma primeira otimização:

- otimização 1 - a imagem a ser tratada deve ser dividida em blocos menores facilitando a computação das transformadas. Outra justificativa para esta abordagem é que, apesar de existir bastante correlação com os vizinhos próximos, existe pouca ou nenhuma correlação entre pontos distantes de uma mesma imagem. Os ganhos de processamento com esta abordagem suplantam em muito as perdas em termos de compressão.

O cálculo das funções de cosseno, por ser uma função transcendental, também exige bastante poder de processamento. Verificando a fórmula da DCT pode-se pré-calcular todos os valores de cosseno a serem utilizados e, depois disto, apenas realizar operações aritméticas de soma e multiplicação, o que leva à segunda otimização:

- otimização 2 - os cossenos utilizados devem ser pré-calculados e armazenados, realizando-se assim apenas operações aritméticas ao se calcular a fórmula da transformada.

Com um pouco de esforço algébrico, pode-se provar que a somatória dupla da fórmula da DCT bidimensional na Equação 3 corresponde ao produto matricial CPC^T, onde P é a matriz 8x8 representando o bloco de imagem a ser comprimido, C é a matriz definida por:

$$C_{ij} = \begin{cases} \frac{1}{\sqrt{8}}, & i = 0 \\ \frac{1}{2}\cos\left(\frac{(2j+1)i\pi}{16}\right), & i > 0 \end{cases} \tag{2.5}$$

e C^T é a sua transposta. Essa multiplicação matricial exige menor número de multiplicações e somas que a fórmula original, reduzindo ainda mais o tempo de execução da transformada. E isso leva à terceira otimização:

- otimização 3 - aplicação da transformada de cosseno discreta sob a forma matricial CPC^T para reduzir ainda mais o número de operações.

Uma última otimização é a utilização de aritmética de ponto fixo (número fixo de casas decimais). Esta técnica aproveita o fato de que muitos computadores executam as instruções de ponto fixo com mais rapidez do que as de ponto flutuante, acelerando assim o cálculo da transformada. Entretanto, esta técnica introduz uma quantização forçada, mas que no contexto da compressão de dados pode ser desprezada.

- otimização 4 - uso de aritmética de ponto fixo para aproveitar a maior velocidade desse tipo de cálculo na maioria dos computadores.

Ao aplicar a DCT, os coeficientes mais significativos se acumulam no início do vetor (ou matriz) dos dados, ficando o restante com valores muito pequenos e carregando pouca informação. Este tipo de distribuição já é suficiente para que uma técnica de redução de redundância (como os algoritmos LZ77, LZ78 ou LZW) ou uma codificação otimizada (como codificação de Huffman ou codificação aritmética) produzam melhores resultados do que na imagem ou nos dados originais. Entretanto, por se trabalhar sempre com uma precisão finita nas representações numéricas utilizadas, tem-se uma perda nos dados. Portanto, mesmo sem aplicar nenhuma forma de quantização, a compressão usando transformada de cosseno discreta é uma compressão com perdas.

Entretanto, a forma mais comum e que gera melhores resultados, é a aplicação de uma operação de quantização nos dados gerados pela transformada, e apenas o armazenamento dos dados quantizados. Essa quantização permite uma maior eficiência das técnicas de codificação e eliminação de redundância utilizada. Algumas formas de quantização normalmente utilizadas com a DCT são:

- eliminação dos componentes menos significativos - determina-se um patamar de valor ou mesmo de posição na matriz de resultados da transformada, e elimina-se ou substitui-se esses valores por 0;
- divisão inteira dos valores por um coeficiente de quantização fixo - assim pode-se usar menos dígitos, ou bits, para se representar os valores;
- divisão inteira por uma matriz de coeficientes de quantização - esta técnica é a empregada pela maioria dos padrões de compressão de dados, pois é mais flexível e permite que se ajuste a matriz a qualidade desejada da imagem.

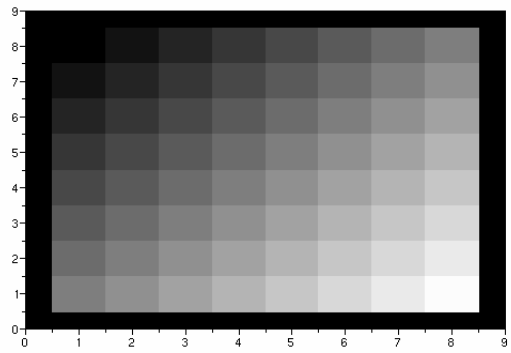
O padrão JPEG usa esta última técnica, e a tabela de coeficientes utilizada deve ser gravada junto com o arquivo comprimido da imagem. A escolha das matrizes no padrão JPEG pode ser da seguinte forma:

1. Uso das tabelas padronizadas de quantização fornecidas pelo padrão JPEG; ou
2. Uso de uma tabela de quantização Q personalizada, em geral calculada com uma fórmula simples que pode ser parametrizada para melhor ou pior qualidade de imagem. Uma fórmula bem comum é a seguinte, que usa um valor inteiro R como parâmetro:

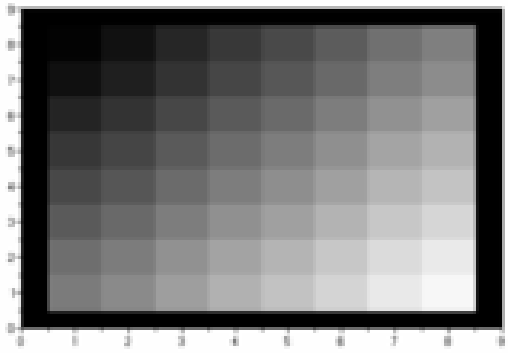
$$Q_{ij} = 1 + (i + j) \times R \tag{2.6}$$

Ainda no padrão JPEG, os coeficientes quantizados são separados (o coeficiente mais significativo de cada bloco 8x8 é separado dos demais para efeito de maior compressão) e comprimidos usando-se uma combinação de RLE (*Run Length Encoding*) e codificação de Huffman. O padrão prevê também a compressão através de uma variante das codificações aritméticas, chamada de codificação QM. Entretanto, a codificação QM, assim como a maioria das codificações aritméticas está protegida por patentes, e é preciso de uma licença do detentor das patentes para ser utilizada. Esta restrição das patentes fez com que a maioria dos compressores de JPEG utilize apenas a codificação de Huffmann, ignorando o uso do QM.

A Figura 2.4 apresenta alguns exemplos de imagens transformadas usando DCT (de tamanho 8x8 pixels, ampliadas para maior clareza), quantizadas com a tabela recomendada pelo padrão JPEG, e destransformadas para recompor a imagem descomprimida. Note que as imagens onde as transições de tons são mais suaves proporcionam uma melhor recomposição da imagem.

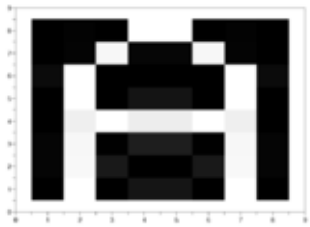


(a) Degradê cinza sem passar por DCT

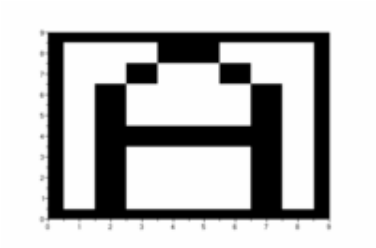


(b) Degradê cinza após DCT

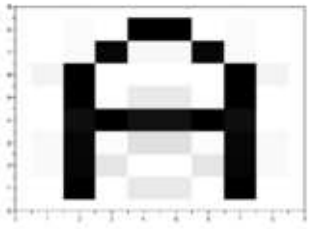
(c) Letra com fundo preto sem passar por DCT



(d) Letra com fundo preto após passar por DCT



(e) Letra com fundo branco sem passar por DCT



(f) Letra com fundo branco após passar por DCT

Figura 2.4: Efeito da DCT em imagens.

Essa característica de suavizar as bordas, que pode ser notada nas imagens da Figura 2.4, é o que faz o DCT ser amplamente utilizado em compressão de fotos, pois nesse tipo de imagem, a presença de bordas e mudanças abruptas é mais rara. Para a compressão de desenhos e textos escaneados, esta técnica não é tão boa pois "borra" ligeiramente as bordas das linhas retas, como pôde ser visto nos dois últimos conjuntos de imagens.

Para demonstrar a capacidade de compressão proporcionada, usa-se a matriz que gerou a imagem em degradê e mostra-se aqui todos os passos da compressão usando DCT. Primeiro tem-se a matriz original a seguir. Pode-se ver que essa matriz possui vários valores distintos, não alcançando bons resultados apenas com a eliminação das repetições:

$$\begin{pmatrix} 1. & 19. & 37. & 55. & 73. & 91. & 109. & 127. \\ 19. & 37. & 55. & 73. & 91. & 109. & 127. & 145. \\ 37. & 55. & 73. & 91. & 109. & 127. & 145. & 163. \\ 55. & 73. & 91. & 109. & 127. & 145. & 163. & 181. \\ 73. & 91. & 109. & 127. & 145. & 163. & 181. & 199. \\ 91. & 109. & 127. & 145. & 163. & 181. & 199. & 217 \\ 109. & 127. & 145. & 163. & 181. & 199. & 217 & 235. \\ 127. & 145. & 163. & 181. & 199. & 217 & 235. & 253. \end{pmatrix} \quad \textbf{(1)}$$

Quando se aplica o DCT, tem-se a matriz seguinte. Esta matriz já tem vários valores zerados, que podem ser eliminados na compressão por alguma técnica de remoção de repetições, como RLE. A matriz a seguir está arredondada em duas casas decimais.

$$\begin{pmatrix} 1016. & -327.99 & 0. & -34.29 & 0. & -10.23 & 0. & -2.58 \\ -327.99 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ -34.29 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ -10.23 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ -2.58 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix} \quad \textbf{(2)}$$

O passo seguinte é aplicar a quantização. Nesse momento pode-se ver que o número de posições zeradas aumenta ainda mais, e os valores restantes são todos relativamente pequenos, podendo ser representados em um arquivo com número menor de bits do que os valores maiores do arquivo original:

$$\begin{pmatrix} 63. & -30. & 0. & -2. & 0. & 0. & 0. & 0. \\ -27. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ -2. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix} \textbf{(3)}$$

Após a destransformação da matriz quantizada, usando IDCT, observa-se que os valores quase não mudaram em relação ao arquivo original, com a maior diferença entre eles na posição (2,8) que é de 6 (em um máximo de 256), ou seja, menos de 3%.

$$\begin{pmatrix} 4. & 18. & 39. & 57. & 74. & 93. & 113. & 128. \\ 17. & 32. & 52. & 71. & 88. & 106. & 127. & 141. \\ 37. & 52. & 72. & 91. & 107 & 126. & 146. & 161. \\ 56. & 70. & 91. & 109. & 126. & 144. & 165. & 179. \\ 73. & 87. & 108. & 126. & 143. & 161. & 182. & 196. \\ 91. & 106 & 126. & 145. & 161. & 180. & 200. & 215 \\ 111. & 125. & 146. & 164. & 181. & 200. & 220 & 235. \\ 124. & 139. & 159. & 178. & 195. & 213. & 234. & 248. \end{pmatrix} \textbf{(4)}$$

Para imagens onde as variações dos tons são graduais, a técnica de DCT mostra excelentes resultados, e por isso é adotada nos padrões mais usados hoje em dia.

O padrão MPEG usa para a compressão de áudio uma variante da DCT conhecida como MDCT (*Modified Discrete Cosine Transform*). Esta transformada é bastante parecida com a transformada de cosseno unidimensional, e sua fórmula é:

$$S_i = \sum_{k=0}^{n-1} x_k \cos\left(\frac{\pi}{2n}\left[2k+1+\frac{n}{2}\right](2i+1)\right), i = 0,1,..,\frac{n}{2}-1. \tag{2.7}$$

E a sua inversa, conhecida como IMDCT é dada por:

$$x_k = \sum_{i=0}^{n/2-1} S_i \cos\left(\frac{\pi}{2n}\left[2k+1+\frac{n}{2}\right](2i+1)\right), k = 0,1,..,n-1. \tag{2.8}$$

Maiores detalhes podem ser obtidos em (SALOMON, 2000).

2.2.5. Técnicas de Espalhamento de Espectro

Na técnica de espalhamento de espectro (como o espalhamento de freqüência), os dados escondidos são espalhados ao longo da imagem de cobertura. Uma stego-chave é usada para selecionar randomicamente os canais de freqüência. A *White Noise Storm* é uma ferramenta popular usando esta técnica. Em outra pesquisa (MARVEL; BONCELET; RETTER,

1999), dados embutidos como objeto a ser transmitido, a imagem de cobertura é visualizada como interferência em um *framework* de comunicação de cobertura. Os dados embutidos são primeiramente modulados com pseudo ruído e então a energia é espalhada sobre uma faixa de freqüência larga, alcançando somente um nível muito baixo de força de inclusão. Isto é valioso para alcançar a impercepitibilidade.

2.2.6. Técnicas de Esteganografia em Vídeo

Como já foi dito anteriormente, a esteganografia tira proveito de qualquer meio ou tipo de informação para esconder uma mensagem. No mundo digital atual, há grande quantidade de áudio e vídeo circulando principalmente pela Internet. E a esteganografia tira proveito deste vasto domínio de cobertura.

Quando informações são escondidas dentro de um vídeo, normalmente é usado o método da DCT. Sendo assim, esteganografia em vídeo é muito similar a esteganografia em imagens, exceto pelo fato de que as informações são escondidas em cada frame do arquivo de vídeo. Da mesma forma que nas imagens, quanto maior for a quantidade de informação a ser escondida no vídeo, maior será a possibilidade do método esteganográfico ser percebido.

Maiores detalhes sobre esteganografia em vídeos podem ser encontrados em (HARTUNG; GIROD, 1996) (LANGELAAR; LAGENDIJK; BIEMOND, 1997) (QIAO; NAHRSTEDT, 1998) (KALKER et al., 1999) (LINNARTZ; KALKER; HAITSMA, 1999).

2.2.7. Técnicas de Esteganografia em Áudio

Esconder imagens em sinais de áudio é algo desafiante, pois o sistema auditivo humano (SAH) pode trabalhar em uma faixa muito grande de freqüências. O SAH pode captar até um bilhão de potências diferentes de sinais (altura) e até mil freqüências de sinais distintas. A sensitividade a ruído também é muito apurada. Uma perturbação em um arquivo de som pode ser detectada tão baixa quanto uma em 10 mihões de partes ou 80 dB em um ambiente comum. Apesar de ser tão poderoso para captar sinais e freqüências, o SAH não consegue fazer diferenciação de tudo que recebe. Sendo assim, sons mais altos tendem a mascarar sons mais baixos. Além disso, o SAH não consegue perceber um sinal em fase absoluta, somente em fases relativas. Também existem algumas distorções do ambiente muito comuns que são simplesmente ignoradas pelo ouvido na maioria dos casos.

As técnicas de esteganografia exploram muitas destas “vulnerabilidades” do ouvido humano, porém sempre têm que levar em conta a extrema sensibilidade do SAH.

Para se desenvolver um método de esteganografia em áudio, uma das primeiras considerações a serem feitas é o ambiente onde o som trafegará entre a origem e o destino. Há pelo menos dois aspectos que devem ser considerados: a representação digital do sinal que será usado e o caminho de transmissão do sinal.

Quanto à representação digital, existem dois parâmetros críticos para a maioria das representações de áudio: método de quantização da amostra e taxa de amostragem temporal. Um dos formatos mais populares para representar amostras de áudio digital com alta qualidade é uma quantização linear de 16 bits chamada Windows Audio-Visual

(WAV) e Audio Interchange File Format (AIFF). Um outro formato popular para áudio de menor qualidade é a escala logarítmica de 8 bits $\mu - law$. Estes métodos de quantização introduzem uma distorção no sinal que é mais evidente no caso da quantização de 8 bits $\mu - law$.

Taxas de amostragem típicas para áudio incluem 8kHz, 9,6 kHz, 10 kHz, 12 kHz, 16 kHz, 22,05 kHz e 44,1 kHz. As taxas de amostragem impactam na esteganografia a medida que impõem uma barreira para a porção usável do espectro de freqüências. Não é possível, por exemplo, introduzir modificações que têm componentes de freqüência acima de 4 kHz se o sinal foi amostrado a uma freqüência de 8 kHz.

A última representação a ser considerada é a que produz perdas através do uso de algoritmos de compressão, tal como o MPEG-AUDIO. Estas representações modificam drastricamente o sinal, preservando somente as caracterísiticas que o ouvido humano pode perceber trabalhando com um modelo psico-acústico. Isso quer dizer que o som resultante será similar ao original, mesmo que o sinal resultante seja totalmente diferente.

Existem muitos meios de transmissão pelos quais um sinal pode passar no caminho do codificador até o decodificador. A primeira classe de meios de transmissão que pode ser considerada é um ambiente digital fim a fim. Neste ambiente, o arquivo de som é copiado de uma máquina para outra e não é modificado. Como resultado, a amostra é exatamente a mesma, tanto no codificador quanto no decodificador. Esta classe é a que menos impõe barreiras aos métodos esteganográficos.

A próxima classe de meios de transmissão é quando um sinal é re-amostrado para uma taxa de amostragem maior ou menor que a original, mas permanece digital. Esta transformação preserva a magnitude absoluta e a fase da maioria dos sinais, mas muda as características temporais do mesmo.

A terceira classe é a que apresenta um sinal que é "tocado" dentro de um dispositivo analógico, transmitido em uma linha analógica razoavelmente sem ruídos e depois re-amostrado (digital-analógico-digital). Neste caso não são preservados a magnitude do sinal, a quantização inicial e a taxa de amostragem. Somente a fase do sinal é preservada.

O último caso é quando um sinal é transmitido pelo ar ("tocado") e depois sofre nova amostragem com um microfone. O sinal estará possivelmente sujeito a modificações não lineares, resultando em mudanças de fase, amplitude, ecos e mudança de componentes.

Sendo assim, a representação do sinal e o caminho de transmissão devem ser considerados na escolha de um método de esteganografia. A taxa de dados é muito dependente da taxa de amostragem e do tipo de som que está sendo codificado. Um valor típico de taxa é 16 bps, mas este valor pode variar de 2 bps a 128 bps.

Codificação *Low-bit*

A codificação no bit menos significativo é a maneira mais simples de embutir dados dentro de outra estrutura de dados. Através da substituição do bit menos significativo de cada amostra por um codificador binário, é possível codificar uma grande quantidade de dados

em um sinal de áudio. De maneira ideal, a capacidade do canal deve ser de pelo menos 1kb por segundo (kbps) por 1kHz, isso é, em um canal sem ruído, a taxa de transmissão será de 8kbps em uma amostra de 8kHz e 44kbps em uma amostra de 44kHz. Como conseqüência desta grande capacidade, ruídos audíveis são sempre introduzidos. O impacto destes ruídos diferem de acordo com o conteúdo do sinal. Em um *stream* de áudio de uma partida de futebol, o barulho da torcida ao fundo mascarará o ruído introduzido pela técnica de codificação *low-bit*, enquanto que em um *stream* de música clássica, o ruído seria audível.

A maior desvantagem deste método é a sua baixa imunidade a manipulação. A informação codificada pode ser destruída pelo ruído do canal, na re-amostragem, entre outros, a menos que esta informação tenha sido codificada utilizando técnicas de redundância. Para serem robustas estas técnicas reduzem a taxa de transmissão de dados normalmente pela metade. Na prática, este método deve ser utilizado somente em ambientes de transmissão digitais (digital-digital).

Codificação em Fase

O método de codificação em fase trabalha substituindo a fase de um segmento inicial de áudio por uma fase de referência que representa os dados a serem escondidos. A fase dos segmentos subseqüentes é ajustada para preservar a fase relativa entre os segmentos 2.5.

A codificação em fase é um dos mais efetivos métodos de codificação em termos de sinal percebido quando comparado com a percepção do ruído. Quando a relação de fase entre cada componente de freqüência é mudada muito drasticamente, uma dispersão de fase será notada. Entretanto se as modificações das fases forem pequenas, uma codificação inaudível é obtida. A codificação em fase trabalha com o fato de que os componentes de fase de um som não são tão perceptíveis pelo ouvido humano quanto é o ruído.

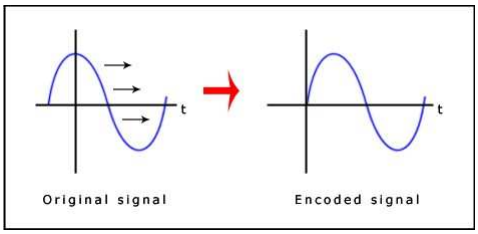


Figura 2.5: Sinal original versus sinal codificado.

Para esconder dados usando a codificação em fase, o seguinte procedimento deve ser seguido:

- o som original é quebrado em segmentos menores, os quais devem possuir tamanho igual à mensagem a ser escondida;
- uma Transformada de Fourier Discreta deve ser aplicada em cada segmento para criar uma matriz de fases e uma outra de magnitude do sinal;

- calcular as diferenças de fases entre os segmentos adjacentes;
- as mudanças de fase entre segmentos consecutivos são facilmente detectáveis. A fase absoluta dos segmentos pode mudar, mas as diferenças relativas de fase entre segmentos adjacentes deve ser preservada. Por isso a mensagem a ser escondida deve ser inserida somente no vetor de fase do primeiro segmento da seguinte forma:
  - $\pi/2$ se o bit a ser representado for 0;
  - $-\pi/2$ se o bit a ser representado for 1.
- uma nova matriz de fases é criada utilizando a fase do primeiro segmento e a matriz de fase original;
- utilizando a nova matriz de fase e a matriz original de magnitude, o som é reconstruído através da DFT inversa para depois concatenar os segmentos que formarão o som a ser transmitido.

Para a decodificação e a extração da mensagem secreta, o receptor deve conhecer o tamanho dos segmentos. O receptor pode então usar a DFT para obter as fases e extrair a informação. Uma desvantagem associada com a codificação em fase é uma baixa taxa de transmissão de dados pelo fato de que a mensagem secreta é codificada somente no sinal do primeiro segmento. Este problema pode ser resolvido aumentando o tamanho do segmento. Entretanto isso pode mudar muito a relação entre cada componente de freqüência, tornando a técnica fácil de ser detectada. Como resultado, o método de codificação em fase deve ser usado somente quando uma pequena quantidade de dados precisa ser escondida.

Spread Spectrum

No contexto de esteganografia em áudio, o *spread spectrum* (SS) tenta espalhar informações secretas sobre o espectro de freqüências de áudio tanto quanto possível. A codificação usando SS é análoga à LSB que randomicamente espalha os bits da mensagem sobre todo o arquivo de som. Entretanto, diferentemente da codificação LSB, o SS espalha a mensagem secreta sobre o espectro de freqüências do arquivo de som utilizando um código que é independente do sinal. Assim, o sinal resultante ocupa uma banda superior à utilizada para a transmissão do sinal original.

Duas versões de SS podem ser utilizadas na esteganografia: *Direct-Sequence Spread Spectrum* (DSSS) e *Frequency Hopping Spread Spectrum* (FHSS). No DSSS, a mensagem secreta é espalhada utilizando uma chave chamada *chip rate* e depois modulada com um sinal pseudo-randômico. Então a mensagem modulada é misturada com o sinal de cobertura. Já no FHSS, o espectro de freqüência do arquivo de áudio é alterado de tal forma que a mensagem seja codificada segundo um padrão de saltos entre as freqüências do espectro (PETITCOLAS; KATZENBEISSER, 1999).

O método utilizando SS tem bom potencial e é melhor em algumas circunstâncias que o LSB e a técnica de codificação em fase, pois oferece taxa de transmissão moderada

enquanto mantém alto nível de robustez contra técnicas de remoção. Entretanto, o método SS tem a desvantagem de introduzir ruído no som de cobertura, assim como a abordagem LSB.

Escondendo Informações com Eco

Nas técnicas de esteganografia utilizando eco, a informação é escondida em um arquivo de som através da introdução de um eco. Assim como o método de *spread spectrum*, este método também apresenta a vantagem de permitir uma maior taxa de transmissão e robustez superior quando comparado com outros métodos indutores de ruído.

Para esconder os dados de maneira eficaz, variam-se três parâmetros do sinal de eco: amplitude, taxa de deterioração e variação do sinal original (*offset*). Os três parâmetros são configurados abaixo dos limites que o ouvido humano pode perceber facilmente. O *offset* é utilizado para representar a mensagem binária codificada. O codificador utiliza dois valores de tempo de atraso: um para representar o bit 1 (*offset*) e outro para representar o bit 0 (*offset* mais delta), conforme visto na Figura 2.6

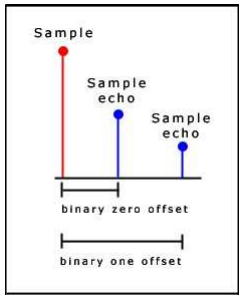


Figura 2.6: Codificando o eco.

Se um eco do sinal original for produzido, somente um bit de informação será codificado. Por isso, o sinal original é quebrado em blocos antes do processo de codificação começar. Uma vez que o processo de codificação é completado, os blocos são concatenados novamente. A cada bloco é assinalado o valor “1” ou “0” baseado na mensagem que será transmitida.

Ao utilizar esta implementação de esteganografia em eco, o processo pode resultar em um sinal que possui uma mistura de ecos, acarretando no aumento do risco de detecção. Uma segunda implementação pode resolver este problema. Primeiro um sinal de eco é criado a partir do som original inteiro usando o valor binário 0 do *offset*. Então um segundo sinal de eco é criado utilizando o sinal original inteiro agora utilizando o valor de *offset* binário 1. Desta forma, o primeiro eco somente contém zeros e o segundo somente contém valores um. Para efetuar a junção, dois *mixers* de sinais são utilizados. O *mixer* tem valor 0 ou 1, dependendo do bit que deve ser codificado. Como exemplo, será codificada a palavra “HEY” que apresenta os dois sinais, conforme Figura 2.7.

O sinal de eco “1” é então multiplicado pelo “1” do mixer e o sinal de eco “0” é multiplicado pelo “0” do mixer. Então os dois resultados são adicionados para obter o sinal final, que é menos abrupto do que o obtido usando o primeiro método.

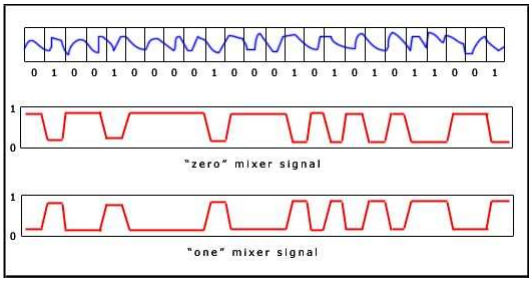


Figura 2.7: Codificando com mixer.

Para extrair a mensagem secreta do stego-sinal, o receptor deve quebrar o sinal na mesma seqüência do bloco usada durante o processo de codificação. Então, a função de autocorrelação do sinal (esta função é uma transformada de Fourier no espectro de freqüência do sinal) pode ser usada para decodificar a mensagem, pois revela um ponto em cada *offset* do tempo do eco, permitindo que a mensagem seja reconstruída.

Através da utilização dos métodos descritos é possível codificar e decodificar informações na forma de bits dentro de um fluxo de áudio com alterações mínimas do som original em uma taxa aproximada de 16 bps. Estas alterações mínimas são as que, na média, o ouvido humano não pode diferenciar entre o sinal original e o sinal alterado.

Outros trabalhos relacionados à esteganografia em áudio podem ser encontrados em (BONEY; TEWFIK; HAMDY, 1996) (BASSIA; PITAS, 1998) (PRANDONI; VETTERLI, 1998) (SWANSON; ZHU; TEWFIK, 1999) (SU et al., 1999) (SWANSON et al., 1998) (LU; LIAO; CHEN, 2000) (LI; YU, 2000) (KIM, 2000).

2.2.8. Técnicas de Esteganálise

Grande parte das técnicas de esteganografia possuem falhas ou inserem padrões que podem ser detectados. Algumas vezes, basta um agressor fazer um exame mais detalhado destes padrões gerados para descobrir que há mensagens escondidas. Outras vezes, o processo de mascaramento de informações é mais robusto e as tentativas de detectar ou mesmo recuperar ilicitamente as mensagens podem ser frustradas. A pesquisa de métodos para descobrir se há alguma mensagem escondida por esteganografia é chamada de **esteganálise**.

Recuperar os dados escondidos está além da capacidade da maioria dos testes atuais, uma vez que muitos algoritmos de mascaramento utilizam geradores aleatórios muito seguros para esconder a informação durante o processo de mascaramento. Muitas vezes, os bits são espalhados pelo objeto de cobertura. Desta forma, os melhores algoritmos de esteganálise podem não ser capazes de dizer onde está a informação, mas devem dizer se há dados escondidos.

Tipos de Ataques

Existem diversas abordagens para se detectar a presença de conteúdo escondido em imagens digitais. Estas abordagens podem ser divididas em três tipos (ROCHA, 2006): ataques

aurais, estruturais e estatísticos.

- ataques aurais - estes ataques consistem em retirar as partes significativas da imagem como um meio de facilitar aos olhos humanos a busca por anomalias na imagem. Um teste comum é mostrar os bits menos significativos da imagem. Câmeras, scanners e outros dispositivos sempre deixam alguns padrões nos bits menos significativos.

- ataques Estruturais - a estrutura do arquivo de dados algumas vezes muda assim que outra mensagem é inserida. Nesses casos, um sistema capaz de analisar padrões estruturais seria capaz de descobrir a mensagem escondida. Por exemplo, se mensagens são escondidas em imagens indexadas (baseadas em paletas de cores), pode ser necessário usar diferentes versões de paletas. Este tipo de atitude muda as características estruturais da imagem de cobertura, logo as chances de detecção da presença de uma mensagem escondida aumentam (WAYNER, 2002).

- ataques estatísticos - os padrões dos pixels e seus bits menos significativos freqüentemente revelam a existência de uma mensagem secreta nos perfis estatísticos (WAYNER, 2002; WESTFELD; PFITZMANN, 2000; PROVOS; HONEYMAN, 2003). Os novos dados não têm os mesmos perfis esperados. Muitos dos estudos de Matemática e Estatística têm por objetivo classificar se um dado fenômeno ocorre ao acaso. Cientistas usam estas ferramentas para determinar se suas teorias explicam bem tal fenômeno. Estas técnicas estatísticas também podem ser usadas para determinar se uma dada imagem e/ou som possui alguma mensagem escondida. Na maioria das vezes, os dados escondidos são mais aleatórios que os dados que foram substituídos no processo de mascaramento ou inserem padrões que alteram as propriedades estatísticas inerentes do objeto de cobertura (WESTFELD; PFITZMANN, 2000; PROVOS; HONEYMAN, 2003; WAYNER, 2002).

Principais Técnicas de Esteganálise

A seguir, são apresentadas algumas das principais técnicas de esteganálise baseadas em ataques estatísticos existentes.

1. Esteganálise por teste do χ^2 (*Chi-Square Test Approach*).

   O teste Chi-quadrado permite verificar igualdade (semelhança) entre categorias discretas e mutamente exclusivas (por exemplo, diferenças de comportamento entre homens e mulheres). Cada indivíduo ou item deve pertencer a uma e somente uma categoria.

   As seguintes suposições precisam ser satisfeitas:

   (a) os dois grupos são independentes;

   (b) os itens de cada grupo são selecionados aleatoriamente;

   (c) as observações devem ser freqüências ou contagens;

(d) cada observação pertence a uma e somente uma categoria;

(e) a amostra deve ser relativamente grande (pelo menos 5 observações em cada célula e no caso de poucos grupos (2 x 2) pelo menos 10);

A hipótese *H0* é que não existe diferença entre as freqüências (contagens) dos grupos. A hipótese alternativa é que existe diferença. Para se testar as hipótese é preciso testar se existe diferença significativa entre as freqüências observadas e as esperadas em cada extrato.

Exemplo: Deseja-se saber se existe diferença na percepção de homens e mulheres em relação a uma afirmativa feita. As categorias são homens e mulheres, e número total de mulheres é diferente do número total de homens. Cada item pertence a uma e somente uma destas categorias. Da mesma maneira, cada indivíduo poderá responder somente de uma forma. O resultado deve ser comparado com o que seria obtido se não houvesse diferença entre os grupos. Para ilustrar, supõe-se 99 homens e 99 mulheres na amostra. Neste caso, se os grupos se comportassem da mesma forma e respondessem igualmente para cada situação o resultado seria 33 pessoas em cada célula.

Em geral os grupos não são igualmente distribuídos. O valor esperado de cada célula é uma proporção do valor total. Um caso real é apresentado na Tabela 2.1:

Tabela 2.1: Tabela exemplo para o teste χ^2.

	Homens	Mulheres	Total
Concorda	58	35	93
Neutro	11	25	36
Não concorda	10	23	33
Total	79	83	162

Os valores esperados para cada célula são obtidos multiplicando o percentual da coluna pelo total da linha, isto é, total da linha x (total coluna / total). Por exemplo: 45,35 = 93 x 79/162, conforme Tabela 2.2.

O valor de chi-quadrado para cada célula é a diferença ao quadrado entre o valor esperado e o valor medido dividido pelo valor esperado, conforme formula a seguir.

$$\chi^2 = \frac{(ValorEsperado - ValorMedido)^2}{ValorEsperado} \tag{2.9}$$

O chi total é a soma dos valores de cada célula. O valor de χ^2 calculado deve ser comparado com o valor de chi tabelado, quanto maior o valor de chi calculado, maior a diferença. Para obter o valor de chi tabelado (tabela de distribuição χ^2) deve-se escolher o valor do nível de significância(alfa) adequado para a situação.

Em esteganografia, as funções de cobertura de alguns softwares, por exemplo o Ezstego (EZSTEGO, 2007) reescrevem os bits menos significativos dos bytes sorteados para tal fim guardando seus índices. Isso gera valores modificados de bytes

Tabela 2.2: Cálculo do χ^2.

Esperado				
		Homens	Mulheres	Total
	Concorda	45,35185	47,64815	93
	Neutro	17,55556	18,44444	36
	Não concorda	16,09259	16,90741	33
	Total	79	83	162
Chi				
		3,527434	3,357437	
		2,447961	2,329987	
		2,306632	2,195469	
Chi Tabelado =	2			

que só diferem, quando diferem, no último bit. Este par de valores (iniciais e transformados) será chamado de PoVs (*Pair of Values*). Se os bits usados para escrever no bit menos significativo são igualmente distribuídos, a freqüência dos valores de cada PoV se torna igual. A idéia dos ataques estatísticos é comparar uma distribuição de freqüência teórica esperada m um histograma com algumas distribuições observadas em possíveis imagens que podem ter sido modificadas. A distribuição de freqüência teórica é obtida com o chi tabelado usando o nível de significância adequado (alfa).

Um ponto crítico é como obter a distribuição de freqüência teórica. Esta freqüência não deve ser derivada da amostra que está sendo analisada pois a amostra pode ter sido modificada por esteganografia. O problema é que na maioria dos casos não se tem a amostra original para comparar. Na amostra original, a freqüência teórica esperada é a média aritmética das duas freqüências de um PoV. Isso porque a função mascaramento do método esteganográfico sobrescreve os bits menos significativos e isso não muda a soma destas duas freqüências (freqüência de um PoV). A contagem dos valores de freqüência pares é transferida para o valor ímpar correspondente de freqüência em cada PoV e vice-versa. Este fato permite obter a distribuição de freqüência esperada da amostra analisada, não necessitando da original para o teste.

2. Análise RS.

   Apresentada por Jessica Fridrich (FRIDRICH; GOLJAN, 2002), esta técnica consiste na análise das inter-relações entre os planos de cores presente nas imagens analisadas. A classificação é feita pontualmente, sem utilização de treinamento e é dependente do contexto da imagem analisada.

   Este é um dos métodos de detecção mais robustos disponíveis. Para análise podem ser utilizadas imagens coloridas ou em tons de cinza. Não existe distinção na profundidade de cores na imagem analisada, isto pode ser válido tanto para imagens de 8 bpp (bits por pixel) quanto para imagens de 32 bpp.

   As definições feitas por Rocha (ROCHA, 2006) ressaltam os seguintes aspectos:

   “Seja IMG a imagem testada. IMG possui MxN *pixels* e cada *pixel* tem os seus

valores dados por um conjunto P. Como exemplo, para uma imagem de 8 bpp, tem-se P = 0,....255. Então, divide-se IMG em grupos de *pixels* em disjuntos G de n *pixels* adjacentes, onde G = $(x_1,...,x_n)$.

Como exemplo, pode-se escolher grupos de n = 4 *pixels* adjacentes. Feito isso, define-se uma função de discriminação f responsável por atribuir um número real $f(x_1,...,x_n)$ para cada grupo de *píxels* G = $(x_1,...,x_n)$. Quanto mais aleatório for o grupo, maior o valor da função de discriminação, dada por $f(x_1,x_2,...,x_n) = \sum_{i=1}^{n-1} |x_{i+1} - x_i|$.

Finalmente, define-se uma operação inversível F sobre P chamada *flipping*. Por *flipping* entende-se a permutação dos níveis de cores e consiste em 2 ciclos. Assim, F tem a propriedade que F^2 = Identidade ou F(F(x)) = x para todo $x \in P$. A permutação $F_1 : 0 \Leftrightarrow 1, 2 \Leftrightarrow 3, ..., 254 \Leftrightarrow 255$ corresponde a negar o LSB de cada nível de cor. Adicionalmente pode-se definir uma função de *shifting* (deslocamento) $F_{-1} : -1 \leftrightarrow 0, 1 \leftrightarrow 2, ..., 255 \leftrightarrow 256$, ou $F_{-1}(x) = F_1(x+1)$-1 $\forall x$.

Para completar, define-se F_0 como sendo a permutação de identidade F(x)= x $\forall x \in P$. Estas operações são utilizadas para classificar os grupos de *pixels* em três categorias diferentes R, S e U:

- grupos regulares: $G \in R \Leftrightarrow f(F(G)) > f(G)$;
- grupos singulares: $G \in S \Leftrightarrow f(F(G)) < f(G)$;
- grupos não-usáveis: $G \in U \Leftrightarrow f(F(G)) = f(G)$.

Nestas expressões, F(G) significa que a função de *flipping* F foi aplicada para os componentes do vetor G = $(x_1,...,x_n)$. Caso seja desejado aplicar diferentes *flippings* em diferentes *pixels*, deve-se usar uma máscara M que irá denotar quais os *pixels* deverão sofrer alterações. A máscara M é uma n-tupla com valores -1, 0, 1. Define-se o grupo alterado GA como: GA = $(F_{M(1)}(x1), F_{M(2)}(x2), ..., F_{M(n)}(xn))$.

O objetivo da função F é perturbar os *pixels* de uma forma pouco significativa tal como aconteceria no processo de mascaramento de uma mensagem."

O método também proposto por Rocha (ROCHA, 2006) baseado em (FRIDRICH; GOLJAN, 2002) é descrito a seguir:

"Seja R_M a percentagem do número de grupos regulares em relação ao total de grupos para a máscara M. De forma similar, S_M irá notar o número relativo de grupos singulares. Tem-se que $R_M + S_M \leq 1$ *e* $R_{-M} + S_{-M} \leq 1$, para a máscara negativa. A hipótese estatística para o método é que, em imagens típicas, o valor esperado de R_M é aproximadamente igual ao de R_{-M} e o mesmo é verdade para S_M *e* S_{-M}. A equação definida em $R_M \cong R_{-M}$ *e* $S_M \cong S_{-M}$, foi empiricamente comprovada (FRIDRICH; GOLJAN, 2002). A randomização do plano LSB força a diferença entre R_M *e* S_M para zero à medida que o tamanho m da mensagem escondida cresce. Depois de alterar os LSBs de 50 por cento dos *pixels* (é o que acontece quando se esconde uma mensagem aleatória em todos os *pixels*), obtém-se $R_M \cong S_M$, isto é o mesmo que dizer que a capacidade de mascaramento no plano LSB agora é zero. O fato surpreendente é que um efeito contrário acontece com R_{-M} *e* S_{-M}, sua diferença aumenta proporcionalmente ao tamanho da mensagem escondida."

Desta forma, ao analisar a imagem testada, esta provavelmente estará escondendo uma mensagem se:

- Condição 1: $R_M - R_{-M}$ = i e i é muito grande;
- Condição 2: $R_M - S_M$ = k e k é muito grande.

Valores muito grandes para i acontecem quando $i \geq 2,5\%$ do total de grupos. Valores muito grandes para k acontecem quando $k \geq 25\%$ do total de grupos. Um mascaramento detectável ocorre toda vez que a primeira condição é verdadeira. Caso apenas a segunda condição seja verdadeira, há apenas uma suspeita de que houve um mascaramento (FRIDRICH; GOLJAN, 2002).

3. Métricas de qualidade de imagens (*Image Quality Metrics*).

   Métricas de qualidade de imagem são utilizadas, de forma geral, na avaliação de codificação de artefatos, predição de desempenho de algoritmos de visão computacional, perda de qualidade devido a inadequabilidade de algum sensor, entre outras aplicações.

   Nesta abordagem proposta por Ismail Avcibas (AVCIBAS; MEMON; SANKUR, 2001), essas mesmas métricas são utilizadas para construir um discriminador de imagens de cobertura (sem conteúdo escondido) de estego-imagens (com conteúdo escondido) através da utilização de regressão multi-variada. A classificação é feita por um discriminante linear após um certo treinamento (estabilização dos coeficientes da regressão multi-variada).

4. Métricas de tons contínuos e análise de pares de amostragem (*Continuous Tone Metrics and Sample Pair Analysis*).

   Proposta por Sorina Dumitrescu (DUMITRESCU; WU, 2002), esta abordagem consiste em analisar as relações de identidade estatística existentes sobre alguns conjuntos de *pixels* considerados. As identidades observadas são muito sensíveis ao mascaramento LSB e as mudanças nestas identidades podem indicar a presença de conteúdo escondido.

2.3. Aplicações

Em atividades militares, a descoberta de comunicações secretas pode levar a um ataque imediato do inimigo. Mesmo com a criptografia, a simples detecção do sinal é fatal pois descobre-se não somente a existência de inimigos como também a sua posição. Unindo o conceito de ocultamento de informação com técnicas como modulação em espalhamento de espectro torna-se mais difícil de os sinais serem detectados ou embaralhados pelo inimigo.

Várias técnicas relacionadas a ocultamento de informação levam em consideração sistemas com níveis de segurança. Em uma rede de computadores militares existem vários níveis de segurança. Um vírus ou um programa malicioso se propaga dentro do sistema passando de níveis de segurança inferiores para os superiores. Uma vez que alcança seu objetivo, tenta passar informações sigilosas para setores de nível de segurança

menores. Para isso, ele se utiliza de técnicas de ocultamento para esconder informações confidenciais em arquivos comuns de maneira que o sistema lhe permita ultrapassar níveis de segurança diferentes.

Existem situações onde se deseja enviar uma mensagem sem que seja possível descobrir quem a enviou. Geralmente, esse tipo de situação é mais uma característica de atividades ilegais onde os criminosos envolvidos não desejam ser descobertos se sua mensagem for rastreada. Entretanto, essa situação também tem aplicações em atividades legais onde se deseja que a privacidade do remetente seja mantida. Alguns exemplos dessas situações são: registros médicos ou votações online.

Um tema importante a ser considerado pelo criador das técnicas de ocultamento de informação é a ética. Assim como os conhecimentos apresentados podem ser usados para garantir privacidade de dados médicos, votações ou prover serviços online com segurança, algumas pessoas podem encontrar meios de se aproveitar das vantagens dessa 'comunicação invisível e não rastreável' para executar ações ilícitas como difamações, chantagens ou seqüestros. É um dever dos desenvolvedores de sistemas de ocultamento de informação prestar atenção aos abusos que podem ocorrer.

Existe também grandes aplicações na área da indústria médica no que diz respeito a imagens médicas. Normalmente, é usada uma forma de comunicação padrão chamada DICOM (*digital imaging and communications in medicine*) que separa a imagem das informações relativas ao paciente e ao exame como o nome, data e o médico. Em alguns casos, a ligação entre os dados e a imagem é perdida. Então, se as informações fossem ocultadas dentro da própria imagem, não haveria risco de a imagem se separar dos dados. Ainda não existem pesquisas aprofundadas sobre o efeito que tais inserções de dados na imagem poderiam causar alteração na precisão do diagnóstico. Estudos recentes na área de compressão de imagens médicas revelam que tais procedimentos não atrapalham, o que pode indicar uma certa robustez do diagnóstico a pequenas alterações como as causadas pelas técnicas de ocultamento de informação (FILHO et al., 2005).

Em alguns casos, se deseja monitorar um dado arquivo com direitos autorais que está sendo distribuído na Internet, por exemplo. Para isso, utiliza-se um programa robô que procura em *sites* a divulgação desses arquivos. Ele baixa os arquivos, tenta retirar qualquer informação que possa estar escondida e compara com a informação do arquivo original. Caso as informações sejam compatíveis, sabe-se que o arquivo está sendo distribuído de maneira ilegal. O mesmo pode ser feito com músicas tocadas em transmissões via rádio. O programa procura no sinal do rádio marcas inseridas propositalmente nas músicas a serem protegidas. Se em um dado momento a marca é inteiramente decodificada do sinal, sabe-se que a estação de rádio investigada tocou a música sem autorização.

Pode-se também inserir pedaços de informações dentro dos dados que estão sendo transmitidos para que o público que a receba possa usar. Como exemplo, pode-se ter informações de um dado produto anunciado por uma rádio onde o cliente, com um simples apertar de botão, pode descobrir o preço, local de venda mais próximo ou fabricante. Essas informações são enviadas sem a necessidade de se usar outra banda para transmissão pois ela é inserida no próprio sinal de rádio sem prejudicar a qualidade do mesmo.

Outra aplicação seria inserir uma forma de indexação de músicas a serem arma-

zenadas no banco de dados de uma estação de rádio para que elas sejam acessadas de maneira mais fácil. Pode-se inserir também dados da transmissão como país de origem, autor e produtora.

Atualmente a esteganografia tem sido também explorada em ramos de sistemas de detecção de intrusão (SIEFFERT et al., 2004) e em sistemas de arquivos (HIROHISA, 2007).

Outras aplicações de esteganografia incluem as técnicas de autenticação, criptografia e rastreamento de documentos, que por serem utilizadas normalmente em conjunto com a técnica de marca d'água, são mencionadas a seguir.

2.3.1. Marcas D'Água

O grande crescimento dos sistemas de multimídia interligados pela rede de computadores nos últimos anos apresenta um enorme desafio nos aspectos tais como propriedade, integridade e autenticação dos dados digitais (áudio, vídeo e imagens estáticas). Para enfrentar tal desafio, o conceito de marca d'água digital foi definido.

Uma marca d'água é um sinal portador de informação, visualmente imperceptível, embutido em uma imagem digital. A imagem que contém uma marca é dita imagem marcada ou hospedeira. Apesar de muitas técnicas de marca d'água poderem ser aplicadas diretamente para diferentes tipos de dados digitais, as mídias mais utilizadas são as imagens estáticas.

Existe uma certa confusão entre as marcas d'água imperceptíveis e as visíveis utilizadas em cédulas de dinheiro, por exemplo. As visíveis são usadas em imagens e aparecem sobrepostas sem prejudicar muito a percepção da mesma. São usadas geralmente para que se possa expor imagens em locais públicos como páginas na Internet sem o risco de alguém copiá-la e usá-la comercialmente, pois é difícil remover a modificação sem destruir a obra original. É possível também inserir digitalmente marcas visíveis em vídeo e até audíveis em música.

Marcas Robustas e Frágeis

As marcas d'água digitais são classificadas, de acordo com a dificuldade em removê-las, em robustas, frágeis e semifrágeis. Esta classificação também normalmente determina a finalidade para a qual a marca será utilizada.

As marcas robustas são projetadas para resistirem a maioria dos procedimentos de manipulação de imagens. A informação embutida em uma imagem através de uma marca robusta poderia ser extraída mesmo que a imagem hospedeira sofresse rotação, mudança de escala, mudança de brilho/contraste, compactação com perdas com diferentes níveis de compressão, corte das bordas (*cropping*), etc. Uma boa marca d'água robusta deveria ser impossível de ser removida, a não ser que a qualidade da imagem resultante deteriore a ponto de destruir o seu conteúdo visual. Isto é, a correlação entre uma imagem marcada e a marca robusta nela inserida deveria permanecer detectável mesmo após um processamento digital, enquanto a imagem resultante do processamento continuar visualmente reconhecível e identificável como a imagem original. Por esse motivo, as marcas d'água robustas são normalmente utilizadas para a verificação da propriedade (*copyright*) das

imagens. Apesar de muitas pesquisas, parece que ainda não foi possível obter uma marca d'água robusta realmente segura.

As marcas frágeis são facilmente removíveis e corrompidas por qualquer processamento na imagem. Este tipo de marca d'água é útil para checar a integridade e a autenticidade da imagem, pois possibilita detectar alterações na imagem. Em outras palavras, uma marca d'água frágil fornece uma garantia de que a imagem marcada não seja despercebidamente editada ou adulterada. As marcas frágeis de autenticação detectam qualquer alteração na imagem. Às vezes, esta propriedade é indesejável. Por exemplo, ajustar brilho/contraste para melhorar a qualidade da imagem pode ser um processamento válido, que não deveria ser detectado como uma tentativa de adulteração maliciosa. Ou então, compactar uma imagem com perdas (como JPEG ou JPEG2000) em diferentes níveis de compressão deveria ser uma operação permitida. Ainda, imprimir e escanear uma imagem não deveria levar à perda da autenticação. Assim, foram criadas as marcas d'água semifrágeis.

Uma marca semifrágil também serve para autenticar imagens. Diferentemente, estas procuram distinguir as alterações que modificam uma imagem substancialmente daquelas que não modificam o conteúdo visual da imagem. Uma marca semifrágil normalmente extrai algumas características da imagem que permanecem invariantes através das operações permitidas e as insere de volta na imagem de forma que a alteração de uma dessas características possa ser detectada.

Tipos de Marcas de Autenticação

Pode-se subdividir as marcas de autenticação (tanto frágeis como semifrágeis) em três subcategorias: sem chave, com chave secreta (cifra simétrica) e com chave pública/privada (cifra assimétrica).

Uma marca de autenticação sem chave é útil para detectar as alterações não intencionais na imagem tais como um erro de transmissão ou de armazenamento. Funciona como uma espécie de *checksum*. Se o algoritmo de autenticação sem chave estiver disponível publicamente, qualquer pessoa pode inserir este tipo de marca em qualquer imagem e qualquer pessoa pode verificar se uma imagem contém uma marca válida.

A marca de autenticação com chave secreta (cifra simétrica) é usada para detectar uma alteração que pode ser inclusive intencional ou maliciosa. Este tipo de marca é similar aos códigos de autenticação de mensagem, sendo que a única diferença é que o código de autenticação é inserido na imagem ao invés de ser armazenado separadamente. Os algoritmos para inserção e detecção deste tipo de marca podem ser disponibilizados publicamente, e uma chave secreta é usada em ambas as fases.

As marcas de autenticação com chave pública (cifra assimétrica) utilizam a criptografia de chave pública para inserir uma assinatura digital na imagem. Usando uma cifra de chave pública, a autenticidade de uma imagem pode ser julgada sem a necessidade de se tornar pública qualquer informação privada.

Marca de Autenticação em Imagens de Tonalidade Contínua e Imagens Binárias

Existe uma forma natural de embutir as marcas de autenticação em imagens de tonalidade contínua (*contone*) não compactadas, que é inserir os dados nos bits menos significativos (LSBs). Alterar os LSBs afeta muito pouco a qualidade da imagem, ao mesmo tempo em que se conhece exatamente os bits que serão afetados pela inserção da marca.

Não ocorre o mesmo com as imagens binárias, onde cada *pixel* consiste de um único bit, de forma que não existe LSB. Isto traz dificuldades especiais para projetar marcas de autenticação para este tipo de imagem. Inserir uma marca de autenticação em imagens *contone* compactadas com perdas também apresenta dificuldades especiais.

Entre os três tipos de marca de autenticação, a de chave pública é a que oferece mais recursos. Alguns possíveis usos de uma marca de autenticação de chave pública são mostrados a seguir:

- câmera digital segura - costuma-se citar o artigo de (FRIEDMANN, 1993) como o trabalho que inspirou os primeiros trabalhos de marca d'água de autenticação. A câmera digital proposta produz dois arquivos de saída para cada imagem capturada: a primeira é a própria imagem digital capturada pela câmera em algum formato; e a segunda é uma assinatura digital produzida aplicando a chave privada da câmera (que deve estar armazenada de forma segura em um circuito integrado dentro da câmera). O usuário deve tomar cuidado para guardar os dois arquivos, para que se possa autenticar a imagem mais tarde. Uma vez que a imagem digital e a assinatura digital são geradas pela câmera e armazenadas no computador, a integridade e a autenticidade da imagem pode ser verificada usando um programa para decodificar a assinatura digital, que pode ser distribuído livremente aos usuários. O programa de verificação recebe como entrada a imagem digital, a assinatura digital e a chave pública da câmera. Ele calcula a função *hash* da imagem digital, decriptografa a assinatura digital e verifica se as duas impressões digitais obtidas são iguais. O esquema proposto por Friedman poderia ser melhorado de duas formas. A primeira seria embutir a assinatura digital no arquivo da imagem, o que eliminaria a necessidade de armazenar dois arquivos para cada imagem. Alguns formatos de imagem permitem armazenar alguns dados adicionais no cabeçalho ou rodapé do arquivo. Mas o mais interessante seria embutir a assinatura digital na própria imagem. A segunda seria permitir a localização da região alterada. Isto poderia ser interessante, por exemplo, para descobrir a intenção do falsificador ao adulterar a imagem. A marca d'água de autenticação de chave pública pode ser usada para incorporar essas melhorias à câmera de Friedman;

- autenticação de imagens distribuídas pela rede - uma agência de notícias necessita distribuir pela Internet uma fotografia jornalística, com alguma prova de autenticidade de que a foto foi distribuída pela agência e que ninguém introduziu alterações maliciosas na foto. A agência pode inserir uma marca d'água de autenticação na imagem e distribui a foto marcada;

- fax confiável - uma "máquina de FAX confiável" poderia conter internamente uma chave privada e inserir uma marca d'água em todos os documentos transmitidos por

ela. O receptor de FAX, usando a chave pública da máquina transmissora, poderia verificar que o documento foi originado de uma máquina específica de FAX e que o documento não foi manipulado.

Extração de Marca D'água

Com relação a extração da marca dágua, tem-se três tipos de sistemas diferentes. Cada um deles se diferencia pela sua natureza ou combinação de entradas e saídas:

- marcas d'água privadas (também chamadas de não-cegas) - esse sistema requer a marca d'água original. Dentro desse esquema, existem 2 tipos. No primeiro, é necessário o arquivo original para achar pistas de onde se localiza a marca dentro do arquivo marcado. O sistema do segundo tipo necessita das mesmas informações do anterior, mas ele somente tenta responder a seguinte pergunta: o arquivo contém a marca d'água? Sim ou não?. Espera-se que este sistema seja mais robusto já que transporta pouca informação e requer acesso a dados secretos;

- marcas d'água semiprivadas ou semi-cegas - diferente do anterior, não utiliza o arquivo original na extração. Entretanto, tenta responder a mesma questão. Algumas aplicações onde se poderia utilizar esse esquema seria para provar a propriedade em côrte ou em mecanismos de controle de cópia como em aparelhos de DVDs. No último caso, a chave poderia ser guardada dentro da memória do DVD e qualquer disco que fosse colocado no aparelho só poderia ser decodificado se a marca d'água pudesse ser extraída dos dados de vídeo do anterior. Como não se pode colocar os dados originais de todos os possíveis vídeos a serem usados no aparelho, não se pode usar o esquema de marcas d'água privadas descrito no item anterior;

- marcas d'água públicas ou cegas - não requer nem o arquivo original nem a marca. A intenção do esquema é tentar retirar a marca do dado sem pistas de onde ele se localiza ou como ele seria.

Um estudo sobre diversos algoritmos de marca d'água está descrito em (MEERWALD, 2001).

Para complementar, a tabela 2.3 apresenta um comparativo indicando o objetivo, as especificações e os detalhes de detecção e extração dos algoritmos de marca d'água e esteganografia .

2.3.2. Aplicativos Existentes

Atualmente as redes de computadores provêem um canal de fácil utilização para a esteganografia. Vários tipos de arquivo podem ser utilizados como imagem de cobertura incluindo imagens, sons, texto e até executáveis. Por isso é grande o número de aplicativos já criados para tentar usar esta facilidade. Por outro lado, existem também alguns softwares de esteganálise que tentam localizar os dados embutidos nas diversas mensagens de cobertura. Tais aplicações podem ser encontradas facilmente na Internet e funcionam em

Tabela 2.3: Tabela comparativa entre Esteganografia e Marca D'água (WANG; WANG, 2004).

	Exigências	Marca d'agua		Esteganografia
		Privado	Público	
Objetivo	Proteção de direitos de propriedade intelectual	++++		–
	Transmissão da mensagem secreta sem levantar suspeita	–		++++
Especificações	Invisibilidade Perceptível	++++		+++++
	Estatístico ou Algoritmo de Invisibilidade	+		+++++
	Robustes em relação à remoção, destruição, ou falsificação maldosa	+++++		–
	Resistência em relação ao processo de um sinal normal	++++		+
	Capacidade de resistência a compressão comum	++++		++
	Alto Custo	++		++++
Detecção/ Extração	Extração/Detecção sem objeto inserido	–	++++	++++
	Extração somente com presença do objeto inserido	++++	–	–
	Exigência de baixa complexidade na extração/deteção	++		+++
	capacidade opcional de download automático do objeto	+		++
Nota: Crucial: +++++ necessário: ++++ importante: +++ desejável: ++ útil: + desnecessário ou irrelevante: – Os esquemas públicos de marca d'água não necessitam do objeto original na deteção/extração; os esquemas confidenciais requerem a presença do original.				

várias plataformas, desde o DOS, Windows passando por MAC/OS até o Unix/Linux. Esta subseção apresenta alguns exemplos destes softwares e seu funcionamento.

As ferramentas *Ezstego* e *Stego Online* (EZSTEGO, 2007) foram desenvolvidas em Java por Romana Machado e limitadas a imagens indexadas de 8bits no formato GIF. Outra aplicação em Java fácil de usar é o *Revelation* (SOFTWARE, 2007), que esconde arquivos em imagens de cobertura no formato *bitmap* de 24 bits. Por serem escritas em Java as ferramentas são altamente portáveis, podendo ser executadas em Linux, Unix, Windows e MAC/OS.

A tela inicial do Revelation é bem auto explicativa (Figura 2.8). A opção *Conceal* esconde o dado embutido na imagem de cobertura a ser escolhida. A opção *Reveal* decodifica o dado embutido a partir da imagem de cobertura. A seguir encontram-se as telas do processo de leitura do dado embutido. Um arquivo chamado *texto.txt* foi escondido em uma imagem chamada *bitmap.bmp* utilizando o software Revelation (Figura 2.9). Após a decodificação, o arquivo de saída gerado se chama *texto2.txt*, conforme pode ser visto na Figura 2.10.

Figura 2.8: Tela inicial do software Revelation utilizado para esteganografia em figuras com formato bitmap de 24 bits.

Figura 2.9: Passo 1 da decodificação do arquivo escondido na imagem bitmap.bmp. A imagem modificada é escolhida.

Figura 2.10: Segundo passo na decodificação usando o Revelation. Ao pressionar o botão de próximo passo (seta para a direita) a imagem de cobertura é decodificada.

A última tela, com o prompt do sistema operacional, compara os dois arquivos e mostra que são idênticos. Sendo assim, foi possível esconder o texto na imagem e depois recuperá-lo sem problemas (Figura 2.11).

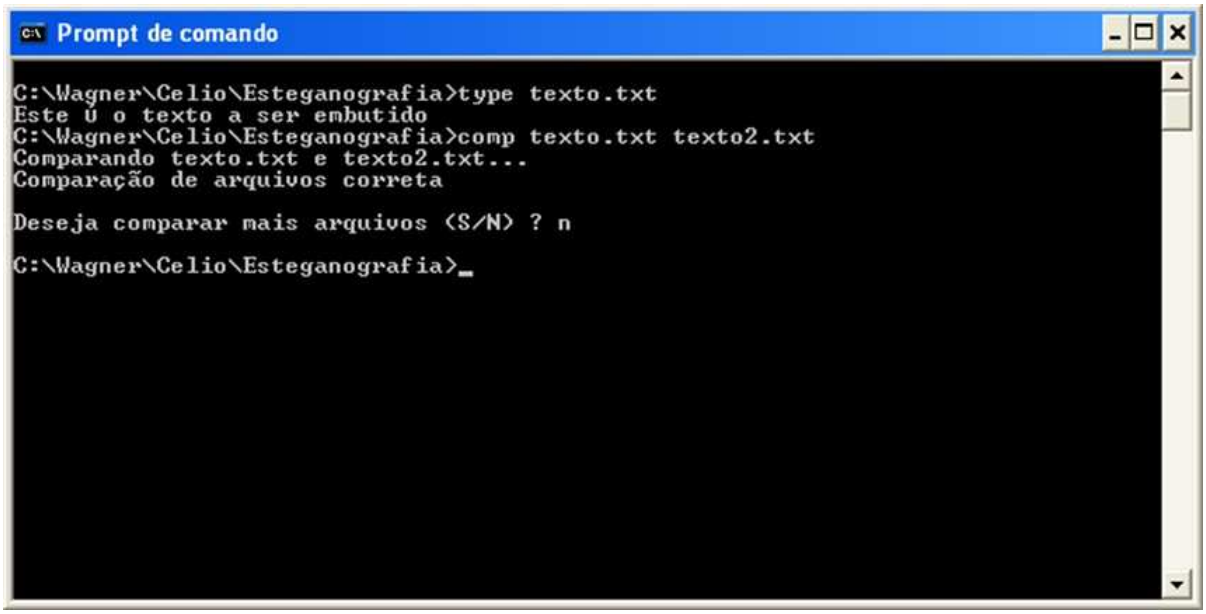


Figura 2.11: Comparação dos arquivos antes e depois do processo. O arquivo de saída do Revelation é idêntico ao arquivo que foi escondido.

O *Hide and Seek* (HIDE; SEEK, 2007) foi desenvolvido por Colin Maroney e é capaz de inserir uma lista de arquivos em uma imagem no formato JPEG. A ferramenta porém não faz uso de criptografia. Uma outra ferramenta parecida chamada *Jphide and Seek* (JPHIDE; SEEK, 2007) desenvolvida por Allan Latham, contém na verdade dois arquivos: *jphide.exe* e *jpseek.exe*. O primeiro esconde a mensagem em um arquivo JPEG e o segundo extrai a mensagem. O programa utiliza criptografia de chave simétrica e o usuário é obrigado a fornecer uma *pass phrase*.

O Jphide and Seek também é de simples operação (Figura 2.12). A opção *Hide* esconde o dado embutido (arquivo de entrada) na imagem de cobertura com formato JPEG (Figura 2.13). É interessante notar que o aplicativo analisa a imagem de cobertura e diz qual o tamanho máximo que o arquivo de entrada deve ter para que o processo seja seguro (Figura 2.14). A opção *Seek* decodifica o dado embutido usando a imagem de cobertura e o salva em um arquivo de saída (Figura 2.15).

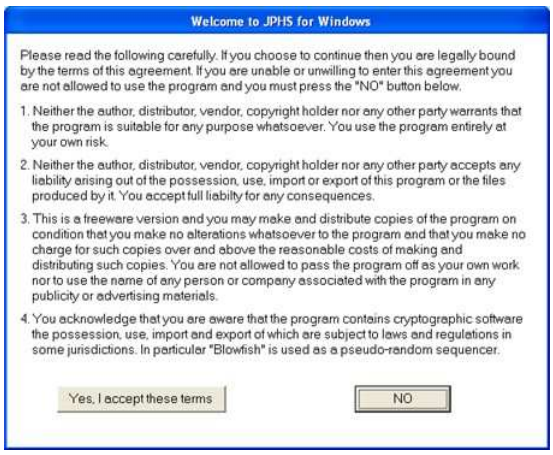


Figura 2.12: Tela inicial do JpHide and Seek.

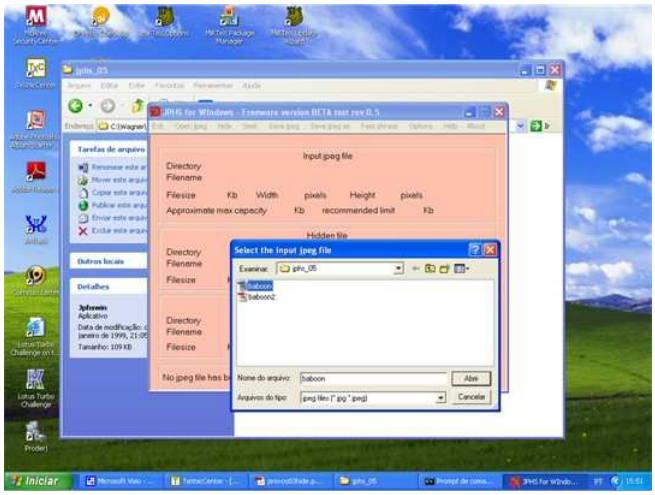


Figura 2.13: Escondendo um arquivo no Jphide and Seek.

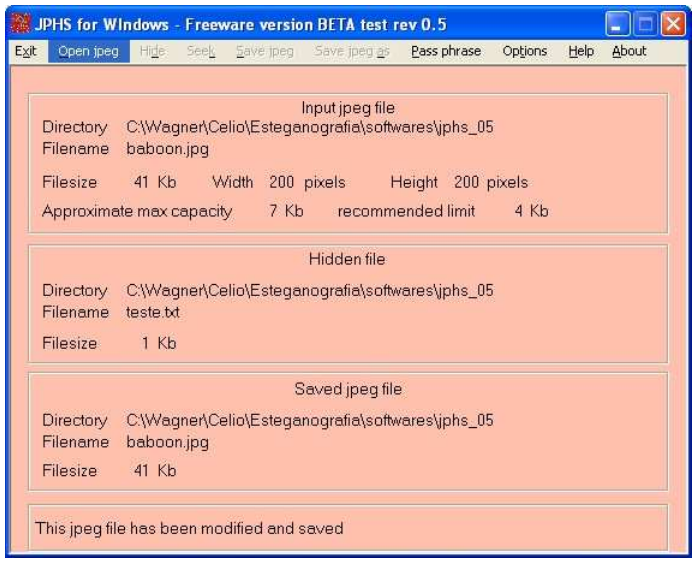


Figura 2.14: Tela após o processo de esteganografia. O arquivo *teste.txt* está escondido dentro da imagem *baboon.jpg*.

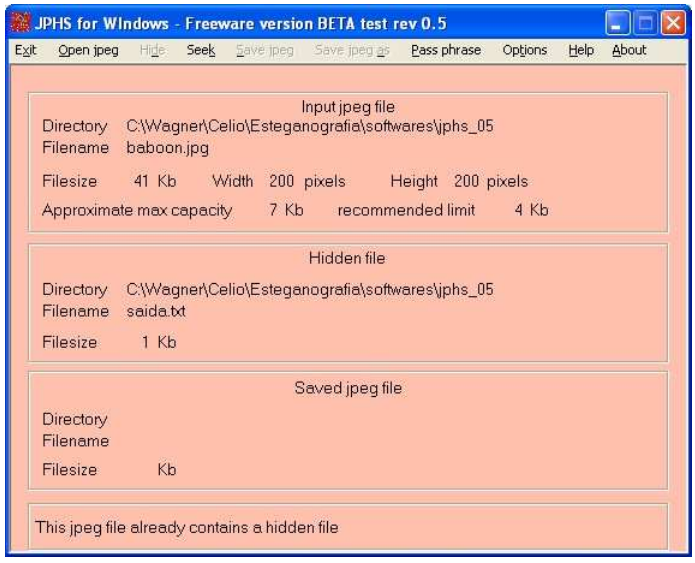


Figura 2.15: Recuperando o dado embutido com Jphide and Seek.

Niels Provos desenvolveu o *Outgess* (OUTGESS, 2007). Este software se propõe a melhorar o passo da codificação da imagem JPEG através de um gerador pseudo-randômico de números. Os coeficientes da transformada de cosseno são escolhidos também de maneira randômica para serem substituídos pelos número gerados aleatoriamente. O bit menos significativo dos coeficientes selecionados é substituído pela mensagem cifrada. Testes estatísiticos de primeira ordem não são capazes de detectar mensagens mascaradas com o *Outgess*. O pseudo-código gerado a partir do Outguess é apresentado na Figura 2.16.

```
Input: message, shared secret, cover image
Output: stego image
initialize PRNG with shared secret
while data left to embed do
   get pseudo-random DCT coefficient from cover image
   if DCT != 0 and DCT != 1 then
      get next LSB from message
      replace DCT LSB with message LSB
   end if
   insert DCT into stego image
end while
```

Figura 2.16: Pseudo-código do OUTGESS (PROVOS; HONEYMAN, 2003).

Os softwares de esteganálise se dispõem a descobrir se os arquivos usados como mensagem de cobertura contém algum dado embutido e se possível identificar o software utilizado no processo de esteganografia. Um destes softwares é o *StegSpy* (STEGSPY, 2007), que permite a identificação de um arquivo que serve como mensagem de cobertura (Figura 2.17). O programa detectará a esteganografia e o software utilizado para esconder o dado embutido (Figura 2.18). A versão atual do software também identifica a localização da mensagem embutida dentro do arquivo de cobertura (Figura 2.19). O *StegSpy* atualmente identifica os programas Hiderman, JPHide and Seek, Masker, JPegX e Invisible Secrets.

Figura 2.17: Tela inicial do StegSpy.

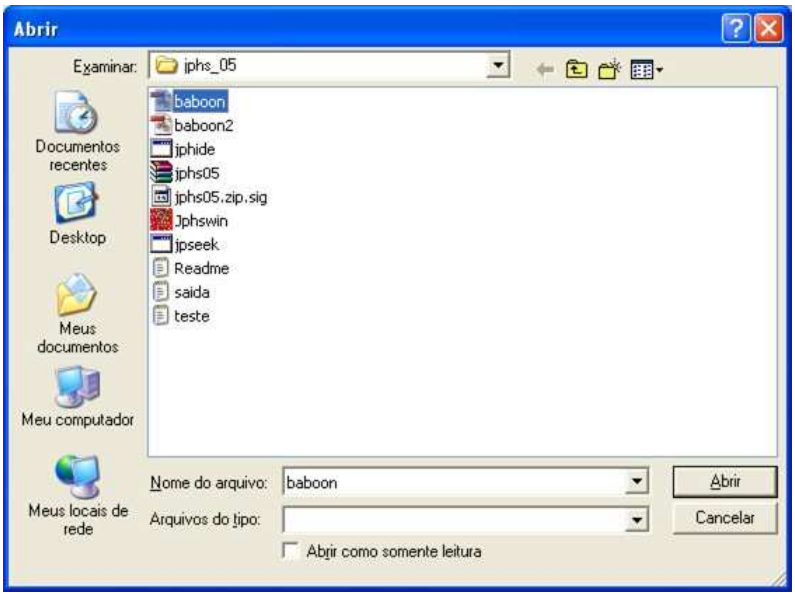


Figura 2.18: Escolhendo uma imagem que sofreu processo de esteganografia. Neste caso o arquivo “baboon.jpg” que é imagem de cobertura utilizada pelo Jphide and Seek.

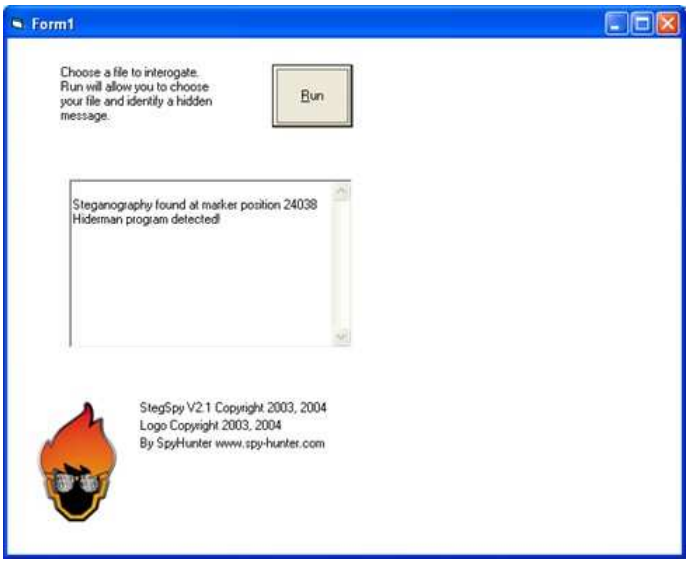


Figura 2.19: O StegSpy acusou uma assinatura de esteganografia no arquivo analisado.

Outra ferramenta de esteganálise é o *StegDetect* (STEGDETECT, 2007) que foi desenvolvida pelo mesmo autor do Outgess (Niels Provos). Este software se propõe a detectar conteúdo esteganográfico gerado pelo softwares Jsteg, JP Hide and Seek, Invisible Secrets, versões mais antigas do Outguess, F5, AppendX, e Camouflage. A versão mais atual do StegDetect suporta análise discriminante linear (LDA) para detectar qualquer estego sistema.

No campo das marcas d'água, existem vários softwares para gerar marcas em diversos tipos de mídias tais como TeleTrax, Alpha Tec, Syscop e DataMark [1]. O ponto fundamental de todos os programas é a robustez da marca produzida. Neste sentido, é preciso testar esta robustez de alguma forma. Em novembro de 1997 a primeira versão do *StirMark* (STIRMARK, 2007) foi publicada. Trata-se de uma ferramenta para testes de robustez de algoritmos de marca d'água. Com o *StirMark* foi possível realizar o primeiro *benchmarking* de algoritmos de marca d'água em 1999 utilizando a versão 3.1 deste software.

O programa SignIt (SIGNIT, 2007) da AlpVision é de fácil utilização para esconder números de série em imagens de vários formatos (Figura 2.20).

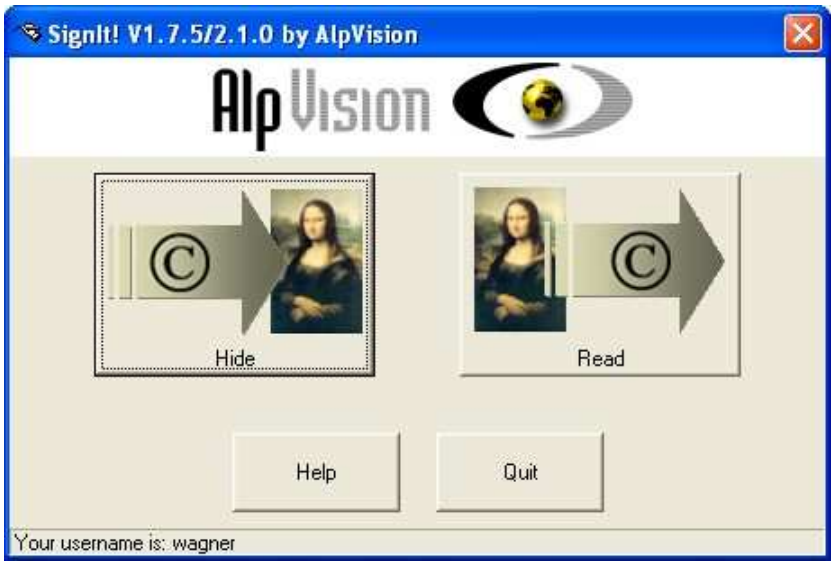


Figura 2.20: Tela inicial do SignIt utilizado para geração e leitura de marcas d´água.

Após instalado deve-se iniciar o programa e escolher entre *esconder* ou *ler* uma marca d´água em uma imagem. Na Figura 2.21, um número IDDN (*Inter Deposit Digital Number*) está sendo escondido em uma imagem. Este número é escondido em todos os lugares na imagem e não pode ser visto pelo olho nu. Além disso, é impossível remover o número de inscrição embutido sem alterar a imagem em um modo visível.

Para controlar a sua utilização, o software se conecta com a empresa desenvolvedora através da Internet (Figura 2.22), que armazena o IDDN de todos os usuários registrados, tornando esse identificador único, podendo ser utilizado para proteger os direitos autorais de imagens e localizar cópias ilegais.

[1]Todos estes programas estão listados em http://home.earthlink.net/ emilbrandt/stego/watermrk.html

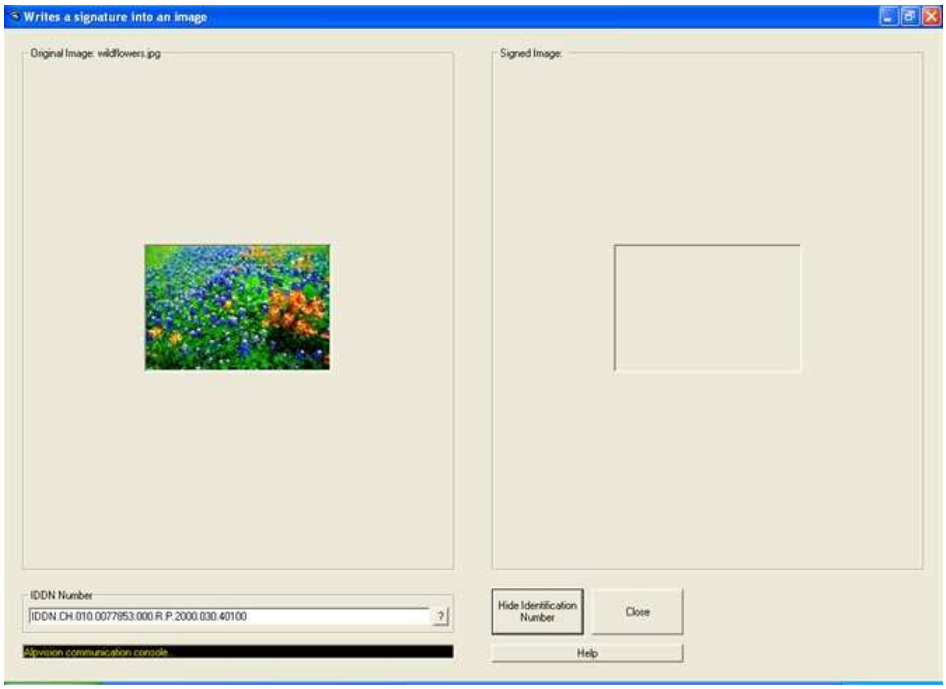


Figura 2.21: Escondendo uma número de série em uma imagem.

Figura 2.22: Software se conecta via Internet com a empresa fabricante para controlar a utilização.

Logo após a inserção da marca d'água (IDDN), o software apresenta uma comparação entre a imagem original e a imagem marcada (2.23).

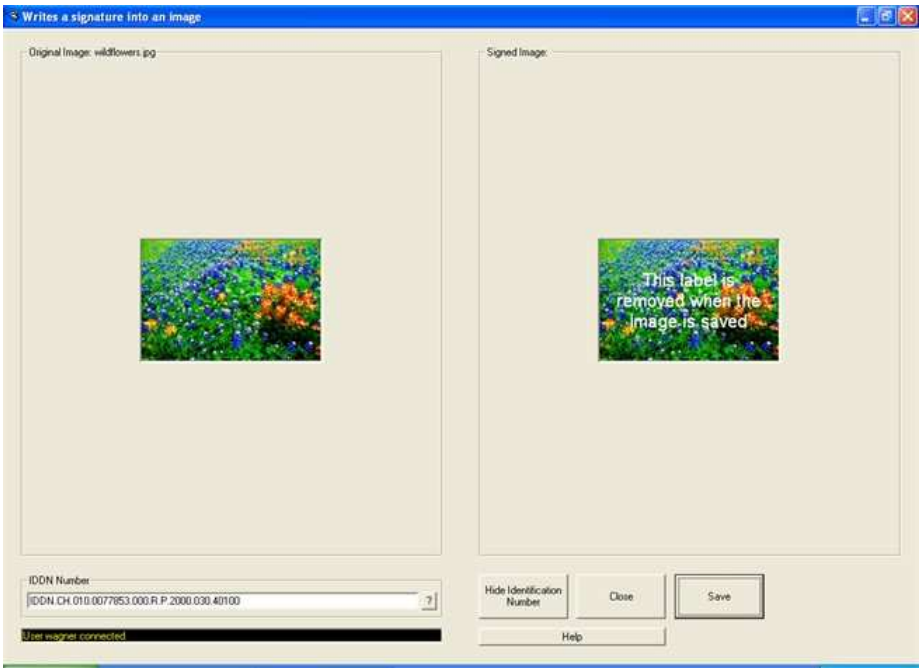


Figura 2.23: Comparação entre a imagem normal e a imagem assinada.

Ainda é possível verificar se uma determinada imagem está protegida por alguma marca, conforme visto na Figura 2.24.

Figura 2.24: Lendo uma imagem com marca d´água. A marca foi reconhecida.

O software GWatermarker foi desenvolvido por Rajan Sheth, Pinto de Adrain e Marina Chandy, todos do Departamento de Tecnologia de Informação pertencente ao Instituto de Tecnologia de San Francis, Mumbai, India (GWATERMARKER, 2007).

GWatermarker insere tanto marca d'água de forma visível a olho nu quanto de maneira invisível de forma robusta. Escrito em Virtual C++.NET, o software utiliza algoritmos próprios para a inserção e remoção das marcas visíveis e invisíveis (algoritmo RC4 para a inserção da chave secreta e o algoritmo hash MD5).

A Figura 2.25 mostra a tela inicial do GWatermarker, com a imagem Lena (LENA, 1972), muito utilizada em trabalhos relacionados a compressão de imagens e marca d'água.

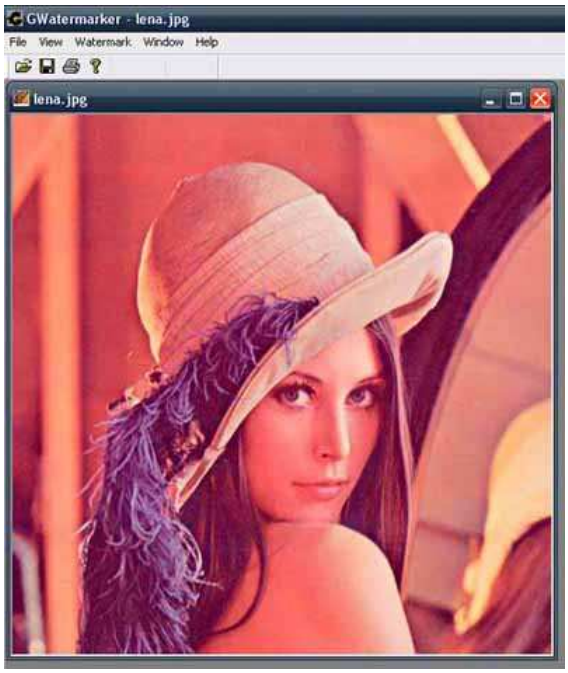
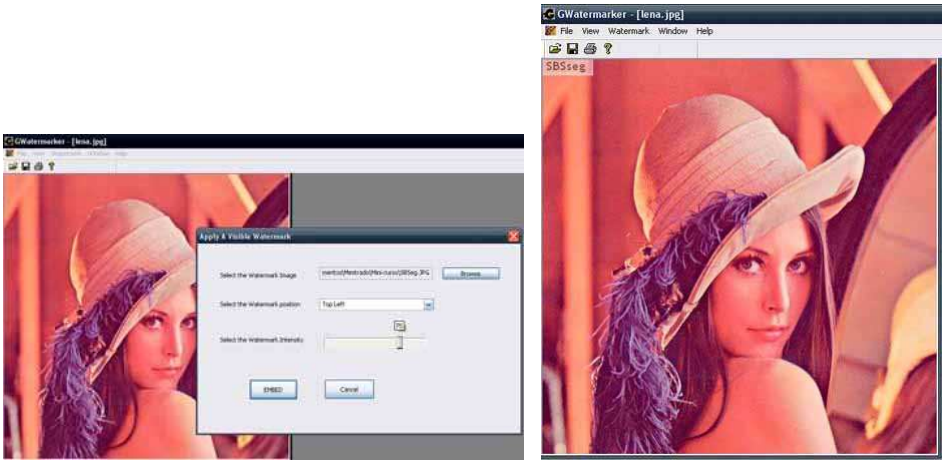


Figura 2.25: Tela inicial.

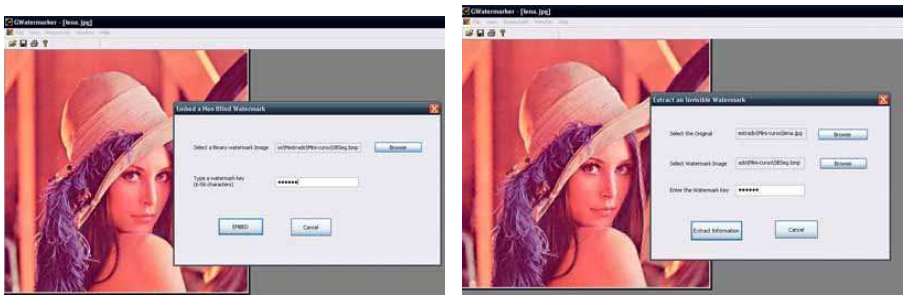
Para inserir uma marca d'água visível, deve-se selecionar o menu *Watermark/Visible Watermarking* e em seguida escolher a imagem, a posição onde ficará a marca e sua intensidade (Figura 2.26(a)). A Figura 2.26(b) apresenta a marca d'água no canto superior esquerdo da imagem, conforme selecionado.

Para a inserção da marca d'água invisível, o procedimento é similar. A partir do menu *Watermark/Invisible (Not Blind)/Embed* (Figura 2.27(a)), define-se a imagem e a chave simétrica (de 6 a 56 caracteres). Para a extração da marca d'água invisível, seleciona-se o menu *Watermark/Invisible (Not Blind)/Extract*, e em seguida a imagem original (sem a marca) a imagem marcada e a chave para a extração (Figura 2.27(b)). A Figura 2.28 confirma a presença da marca d'água na imagem Lena.



(a) Inserindo a marca d'água visível. (b) Marca d'água visível inserida.

Figura 2.26: Marca d'água visível.



(a) Inserindo marca d'água invisível. (b) Extraindo a marca d'agua invisível.

Figura 2.27: Marca d'água invisível.

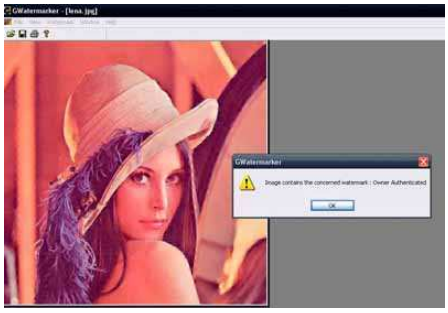


Figura 2.28: Confirmação da presença de marca d'água.

2.4. Considerações Finais e Tendências Futuras

Tanto a esteganografia quanto a marca d'água descrevem técnicas que são usadas na intenção de ocultar uma comunicação dentro de uma informação disfarce. Entretanto, esteganografia se refere tipicamente a uma comunicação ponto-a-ponto. Por isso, o método geralmente não é robusto contra modificações ou tem somente uma robustez limitada que a protege de pequenas alterações que possam ocorrer em termos de transmissão, armazenamento, mudanças de formato, compressão ou conversões digital-analógicas.

Em marcas d'água, por outro lado, o foco está na robustez. Não existe comunicação ponto-a-ponto, mas deseja-se que a marca inserida em um dado seja recuperada de algum modo depois da imagem circular por quaisquer canais típicos da aplicação. Por exemplo, pode-se marcar uma imagem que deseja-se proteger contra cópias sem autorização. Caso alguém a copie e utilize técnicas de processamento de imagem para tentar apagar a marca, ainda assim deve ser possível decodificar a marca da imagem alterada. Isso provaria quem é o verdadeiro autor ou proprietário da imagem. A questão da detecção não é tão importante, apesar de que, se o observador não perceber a marca, ele talvez nem tente removê-la.

Um exemplo de aplicação oposta seria marcar uma imagem para verificar se ela sofrerá alterações. Caso a imagem seja modificada de alguma forma, a marca será destruída, mostrando que o ato realmente aconteceu. A robustez ou a ausência dela define a aplicação da marca utilizada. As marcas d'água robustas devem resistir a ataques e alterações na imagem. As marcas frágeis devem ser destruídas caso a imagem sofra alterações.

Atualmente existem estudos para proteger a esteganografia das técnicas de esteganálise. Em (PROVOS, 2001)são apresentados novos métodos que permitem esconder mensagens de forma segura e resistentes a análise estatística.

Técnicas esteganográficas têm uso legal e ilegal. Como uso legal no presente e no futuro, esteganografia tem sido usada e será cada vez mais utilizada na proteção de direitos intelecutais, principalmente quando se considera as novas formas de comercialização utilizando mídia digital. Neste sentido as técnicas de marca dágua parecem ser um campo profícuo de pesquisa e aplicações no futuro.

Por outro lado, há o uso ilegal de técnicas esteganográficas, que cresce cada vez mais, em virtude da facilidade de acesso a Internet. Usar esteganografia para transitar mensagens ou até pequenas imagens de pornografia ou pedofilia é possível e provável. Um relatório de crimes de tecnologia (HIGH..., 2007) lista alguns tipos de crime comuns utilizando alta tecnologia:

- comunicações criminosas;
- fraudes;
- *hacking*;
- pagamentos eletrônicos;
- pornografia e pedofilia;

- ofensas a propriedade intelectual;
- propagação de vírus e cavalos de tróia.

Um exame preliminar desta lista mostra vários casos de mau uso da esteganografia, principalmente no que se refere à comunicação criminosa. Em termos de segurança da informação há também outras áreas de interesse. Uma área com uso potencial em várias aplicações é o desenvolvimento de protocolos que usam esteganografia para burlar censura. Em (HASELTON, 2000), o coordenador da organização peacefire.org, uma organização que se opõe à censura na Internet a menores de 18 anos, descreve um protocolo que seria "indetectável" por sensores.

Há também a possibilidade de ataques de vírus utilizarem técnicas de esteganografia. As técnicas e ferramentas esteganográficas podem ser utilizadas em conjunto com outras aplicações para automaticamente extrair informações escondidas sem a intervenção do usuário. Um cenário possível para um ataque de vírus poderia ser o envio de uma mensagem escondida em uma imagem enviada por e-mail. Um cavalo de tróia instalado na máquina poderia então extrair o vírus da imagem e infectar várias máquinas.

Finalizando, este trabalho apresentou a evolução da esteganografia ao longo da história e suas aplicações modernas com a chamada esteganografia digital. Foram mostradas as principais técnicas de mascaramento e, em especial, mascaramento em imagens. A esteganografia, quando bem utilizada, fornece meios eficientes e eficazes na busca por proteção digital. Associando criptografia e esteganografia, as pessoas têm em mãos o poder de comunicar-se em segredo pela rede mundial de computadores mantendo suas identidades íntegras e secretas.

Referências Bibliográficas

AVCIBAS, I.; MEMON, N.; SANKUR, B. Steganalysis based on image quality metrics. In: *Proceedings of the Fourth Workshop on Multimedia Signal Processing*. USA: IEEE, 2001. p. 517–522.

BASSIA, P.; PITAS, I. Robust audio watermarking in the time domain. In: *9th European Signal Processing Conference (EUSIPCO'98)*. Island of Rhodes, Greece: [s.n.], 1998. p. 25–28. ISBN 960-7620-05-4. Disponível em: <citeseer.ist.psu.edu/bassia99robust.html>.

BONEY, L.; TEWFIK, A. H.; HAMDY, K. N. Digital watermarks for audio signals. In: *International Conference on Multimedia Computing and Systems*. [s.n.], 1996. p. 473–480. Disponível em: <citeseer.ist.psu.edu/article/boney96digital.html>.

BUCCIGROSSI, R. W.; SIMONCELLI, E. P. Image compression via joint statistical characterization in the wavelet domain. *IEEE Trans Image Proc*, v. 8, n. 12, p. 1688–1701, December 1999.

DUDA, R. O.; HART, P. E.; STORK, D. G. *Pattern Classification (2nd Edition)*. [S.l.]: Wiley-Interscience, 2000. ISBN 0471056693.

DUMITRESCU, S.; WU, X. Steganalysis of lsb embedding in multimedia signals. In: *Proceedings of the Intl. Conference on Multimedia and Exp*. USA: IEEE, 2002. v. 3, p. 581–584.

EZSTEGO, S. e. *Stego e Ezstego*. 2007. Disponível em: <http://www.stego.com>.

FILHO de L. et al. Electrocardiographic signal compression using multiscale recurrent patterns. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, v. 52, n. 12, p. 2739–2753, 2005.

FRIDRICH, J.; GOLJAN, M. *Practical Steganalysis of Digital Images — State of the Art*. 2002. 1-13 p.

FRIEDMANN, G. L. The trustworthy digital camera: Restoring credibility to the photographic image. v. 39, n. 4, p. 905–910, nov. 1993. ISSN 0098-3063. Disponível em: <http://www.cl.cam.ac.uk/ fapp2/steganography/bibliography/043125.html>.

GONZALEZ, R. C.; WOODS, R. E. *Digital Image Processing*. 2nd. ed. Boston, MA, USA: Prentice-Hall, 2002.

GWATERMARKER. *GWatermarker*. 2007. Disponível em: <http://www.cse.unt.edu/ smohanty/ISWARwatermarker/>.

HART, S. V.; ASHCROFT, J.; DANIELS, D. J. *Forensic examination of digital evidence: a guide for law enforcement*. Department of Justice - Office of Justice Programs, USA, April 2004. Technical Report NCJ 199408.

HARTUNG, F.; GIROD, B. Digital watermarking of raw and compressed video. In: *Proc. European EOS/SPIE Symposium on Advanced Imaging and Network Technologies*. Berlin, Germany: [s.n.], 1996. Disponível em: <citeseer.ist.psu.edu/hartung96digital.html>.

HASELTON, B. *A Protocol that uses steganography to circumvent network level censorship*. 2000.

HIDE; SEEK. *Hide and Seek*. 2007. Disponível em: <ftp://csua.berkeley.edu/pub/cypherpunks/steganography/hdsk41b.zip>.

HIGH Technology Crime in California. 2007. Disponível em: <www.ocjp.ca.gov/publications/pub$_h$*tk*1.*pdf*>.

HIROHISA, H. Crocus: a steganographic filesystem manager. In: *ASIACCS '07: Proceedings of the 2nd ACM symposium on Information, computer and communications security*. New York, NY, USA: ACM Press, 2007. p. 344–346. ISBN 1-59593-574-6.

JOHNSON, N. *Steganography*. George Mason University, 1998.

JOHNSON, N. F.; JAJODIA, S. Exploring steganography: Seeing the unseen. *IEEE Computer*, v. 31, n. 2, p. 26–34, 1998. Disponível em: <citeseer.ist.psu.edu/johnson98exploring.html>.

JPHIDE; SEEK. *Jphide and Seek*. 2007. Disponível em: <http://linux01.gwdg.de/ alatham/stego.html>.

KAHN, D. The history of steganography. In: *Proceedings of the First International Workshop*. Cambridge, UK: [s.n.], 1996.

KALKER, T. et al. Video watermarking system for broadcast monitoring. In: WONG, P. W.; III, E. J. D. (Ed.). SPIE, 1999. v. 3657, n. 1, p. 103–112. Disponível em: <http://link.aip.org/link/?PSI/3657/103/1>.

KIM, H. Stochastic model based audio watermark and whitening filter for improved detection. In: *ICASSP '00: Proceedings of the Acoustics, Speech, and Signal Processing, 2000. on IEEE International Conference*. Washington, DC, USA: IEEE Computer Society, 2000. p. 1971–1974. ISBN 0-7803-6293-4.

LANGELAAR, G. C.; LAGENDIJK, R. L.; BIEMOND, J. *Real-time Labeling of MPEG-2 Compressed Video*. 1997. Disponível em: <citeseer.ist.psu.edu/519721.html>.

LENA. *Lena*. 1972. Disponível em: <http://www.cs.cmu.edu/ chuck/lennapg/lenna.shtml>.

LI, X.; YU, H. H. Transparent and robust audio data hiding in subband domain. In: *ITCC '00: Proceedings of the The International Conference on Information Technology: Coding and Computing (ITCC'00)*. Washington, DC, USA: IEEE Computer Society, 2000. p. 74. ISBN 0-7695-0540-6.

LINNARTZ, J.-P.; KALKER, T.; HAITSMA, J. Detecting electronic watermarks in digital video. In: *ICASSP '99: Proceedings of the Acoustics, Speech, and Signal Processing, 1999. on 1999 IEEE International Conference*. Washington, DC, USA: IEEE Computer Society, 1999. p. 2071–2074. ISBN 0-7803-5041-3.

LU, C.; LIAO, H.; CHEN, L. *Multipurpose Audio Watermarking*. 2000. Disponível em: <citeseer.comp.nus.edu.sg/lu00multipurpose.html>.

MARVEL, L.; BONCELET, C.; RETTER, J. *Spread spectrum image steganography*. 1999. Disponível em: <citeseer.ist.psu.edu/article/marvel99spread.html>.

MEERWALD, P. *Digital Image Watermarking in the Wavelet Transform Domain*. Dissertação (Mestrado) — Department of Scientific Computing, University of Salzburg, Austria, January 2001. Disponível em: <http://www.cosy.sbg.ac.at/ pmeerw/Watermarking/MasterThesis>.

MORRIS, S. *The future of netcrime now (1) - threats and challenges*. Home Office Crime and Policing Group, USA, 2004. Technical Report 62.

OUTGESS. *Outgess*. 2007. Disponível em: <http://www.outgess.org>.

PETITCOLAS, F. A. P.; ANDERSON, R. J.; KUHN, M. G. Information hiding — A survey. *Proceedings of the IEEE*, v. 87, n. 7, p. 1062–1078, 1999. Disponível em: <citeseer.ist.psu.edu/petitcolas99information.html>.

PETITCOLAS, F. A. P.; KATZENBEISSER, S. *Information hiding techniques for steganography and digital watermarking*. 1st. ed. [S.l.]: Artech House Books, 1999.

POPA, R. *An analysis of steganography techniques*. Dissertação (Mestrado) — The Polytechnic University of Timisoara, Timisoara, Romênia, 1998.

PRANDONI, P.; VETTERLI, M. *Perceptually hidden data transmission over audio signals*. 1998. Disponível em: <citeseer.ist.psu.edu/prandoni98perceptually.html>.

PROVOS, N. Defending against statistical steganalysis. In: *10th USENIX Security Symposium*. [s.n.], 2001. Disponível em: <niels.xtdnet.nl/papers/defending.pdf>.

PROVOS, N.; HONEYMAN, P. Hide and seek: An introduction to steganography. *IEEE Security and Privacy*, IEEE Educational Activities Department, Piscataway, NJ, USA, v. 1, n. 3, p. 32–44, 2003. ISSN 1540-7993.

QIAO, L.; NAHRSTEDT, K. Watermarking methods for MPEG encoded video: Towards resolving rightful ownership. In: *International Conference on Multimedia Computing and Systems*. [s.n.], 1998. p. 276–285. Disponível em: <citeseer.ist.psu.edu/article/qiao98watermarking.html>.

ROCHA, A. de R. *Randomização Progressiva para Esteganálise*. Dissertação (Mestrado) — Universidade Estadual de Campinas, Campinas, Brasil, 2006.

SALOMON, D. *Data Compression: The Complete Reference*. Segunda edição. Nova Iorque: Springer, 2000.

SIEFFERT, M. et al. Stego intrusion detection system. AFRL/ASU Assured Information Security, Rome, NY, USA, 2004.

SIGNIT. *SignIt*. 2007. Disponível em: <http://www.alpvision.com>.

SOFTWARE, R. *Revelation Software*. 2007. Disponível em: <http://revelation.atspace.biz>.

STEGDETECT. *Stegdetect*. 2007. Disponível em: <http://www.outguess.org/detection.php>.

STEGSPY. *StegSpy*. 2007. Disponível em: <http://www.spy-hunter.com/stegspydownload.htm>.

STIRMARK. *Stirmark*. 2007. Disponível em: <htttp://www.petitcolas.net/fabien/watermarking/stirmark/index.html>.

SU, P.-C. et al. Digital image watermarking in regions of interest. In: *PICS*. [S.l.: s.n.], 1999. p. 295–300.

SULLIVAN, K. et al. Steganalysis of quantization index modulation data hiding. In: *IEEE International Conference on Image Processing*. [s.n.], 2004. p. 1165–1168. Disponível em: <http://vision.ece.ucsb.edu/publications/04ICIPKen.pdf>.

SWANSON, M. D.; ZHU, B.; TEWFIK, A. H. Current state of the art - challenges and future directions for audio watermarking. In: *ICMCS, Vol. 1*. [S.l.: s.n.], 1999. p. 19–24.

SWANSON, M. D. et al. Robust audio watermarking using perceptual masking. *Signal Processing*, v. 66, n. 3, p. 337–355, 1998. Disponível em: <citeseer.ist.psu.edu/swanson98robust.html>.

WANG, H.; WANG, S. Cyber warfare: steganography vs. steganalysis. *Commun. ACM*, ACM Press, New York, NY, USA, v. 47, n. 10, p. 76–82, 2004. ISSN 0001-0782.

WAYNER, P. *Disappearing Cryptography: Information Hiding: Steganography and Watermarking (2nd Edition)*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2002. ISBN 1558607692.

WESTFELD, A.; PFITZMANN, A. Attacks on steganographic systems. In: *IH '99: Proceedings of the Third International Workshop on Information Hiding*. London, UK: Springer-Verlag, 2000. p. 61–76. ISBN 3-540-67182-X.