Steganographie in Rechnernetzen

Hannes Federrath

TU Dresden, Fakultät Informatik, 01062 Dresden

E-Mail: federrath@inf.tu-dresden.de

http://www.inf.tu-dresden.de/~hf2

1 Einführung

Mit Steganographie können geheime Nachrichten über offene, unsichere Datennetze übermittelt werden, ohne daß deren Existenz für Außenstehende überhaupt nachweisbar ist. Hierzu werden die geheimen Nachrichten in einer offenen, unverdächtigen Kommunikation versteckt.

Die Diskussionen um ein Kryptoverbot führten zu einer verstärkten Beachtung der „Wissenschaft vom Verstecken von Nachrichten". Neben der politischen Bedeutung gewinnt sie zunehmend auch an Bedeutung für den Urheberrechtsschutz digitaler Daten. Der mit der Verbreitung von Multimedia-Diensten verbundene Wunsch, auch die Urheberrechte bei der Verbreitung digitaler Objekte (Daten, Programme, Computerkunst etc.) über CD-ROM und Internet zu sichern, kann durch sog. Watermarking und Fingerprinting erreicht werden. Anstelle einer geheimen Botschaft wird in das digitale Objekt Information über den Urheber bzw. Käufer eingebettet.

Dieses Papier diskutiert die Grundprinzipien, Schutzziele und Angreifermodelle steganographischer Systeme, sowohl für vertrauliche Kommunikation als auch zum Watermarking. Es wird an einem Beispiel gezeigt, wie ein Algorithmus, der unsicher gegen einen Angriff ist, verbessert werden kann.

1.1 Steganographie

Mit **Steganographie** wird eine geheimzuhaltende Nachricht (in Abbildung 1b mit *embedded* bezeichnet) in eine Hülle (*cover*) derart eingebettet, daß

1. dem Ergebnis (*stegotext*) die minimalen Veränderungen nicht anzusehen sind und
2. die Veränderungen nicht mit Meßmethoden nachweisbar sind.

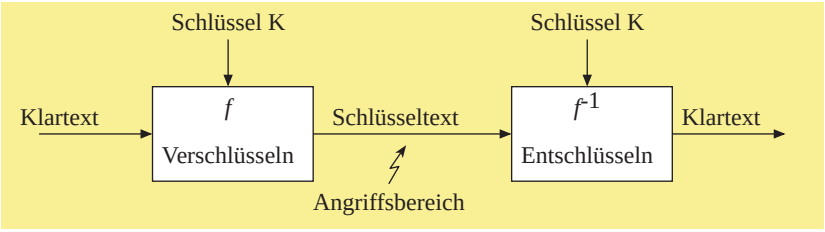
Als Hülle kann jedes Medium dienen, das einen indeterministischen Prozeß, z.B. eine Quantisierung, durchlaufen hat. Digitalisierte Sprache oder Musik, digitalisierte Bilder, Videos etc. sind hervorragend als Medien geeignet. Im Computer künstlich erzeugte Grafiken sind beispielsweise nicht gut geeignet.

Steganographie ist technisch gesehen *keine* Verschlüsselung von Daten.

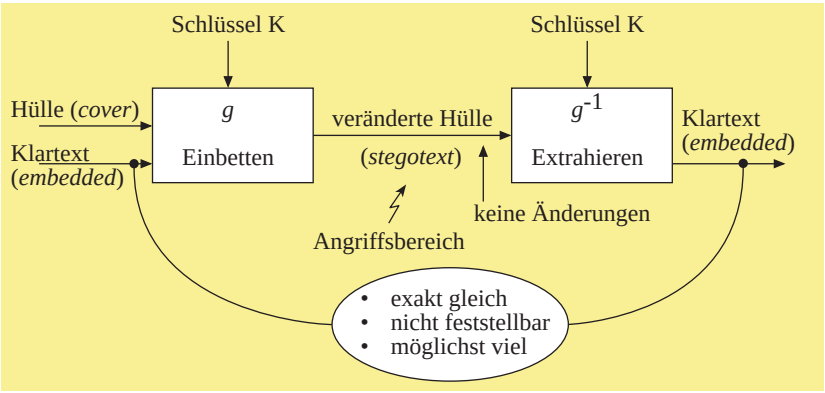
Während bei Kryptographie der Klartext M mittels einer kryptographischen Funktion f in einen für jeden Außenstehenden, d.h. jeden, der nicht den Schlüssel K besitzt, unleserlichen Schlüsseltext überführt wird, entsteht bei Steganographie als Ergebnis der Einbettungsfunktion g eine veränderte Hülle (*stegotext*), die für jeden Außenstehenden, d.h. jeden, der nicht den passenden Schlüssel K besitzt, ebenso eine unveränderte Hülle (*cover*) hätte sein können.

Auf den Punkt gebracht bedeutet dies, daß die Verwendung von Verschlüsselung für den Außenstehenden zumindest erkennbar ist. Obwohl er nicht in Kenntnis des Nachrichteninhalts kommt, entsteht zumindest der Verdacht, daß die Kommunikationspartner etwas zu verbergen haben.

a) Kryptographisches System



b) Steganographisches System zur vertraulichen Kommunikation



b) Steganographisches System zum Watermarking

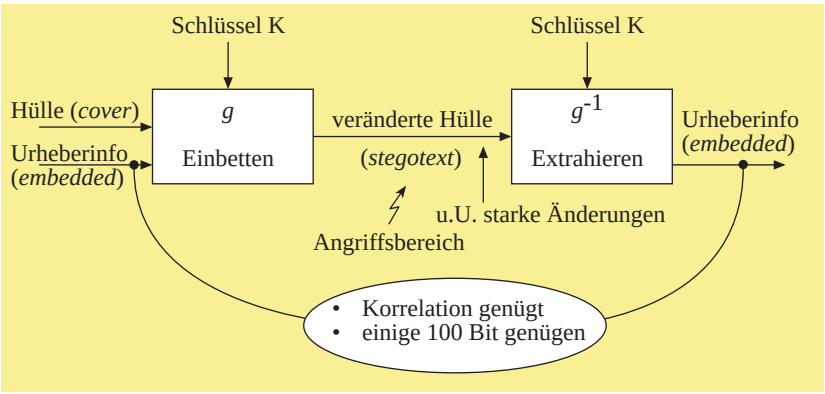


Abbildung 1. Grundaufbau von kryptographischem und steganographischen Systemen

Die Steganographie dagegen kann die Existenz einer geheimen Botschaft verbergen. In einer offenen, unverfänglichen und unverschlüsselten Kommunikation wird unbemerkt eine geheime Botschaft transportiert. Das Grundprinzip der Steganographie setzt keine vorherige Verschlüsselung der geheimen Botschaft voraus, obgleich sie der Geheimhaltung nicht schadet und zumindest bei Steganographie von zweifelhafter Stärke sehr zu empfehlen ist.

1.2 Watermarking

Mit Steganographie können neben der geheimen Nachrichtenübermittlung auch Urheberrechte in digitalen Informationen geschützt werden. Das **Watermarking** bezeichnet dabei einen Vorgang, bei dem ein digitales Objekt mittels steganographischer Techniken verändert wird, um Rechte an diesem Objekt zu schützen. Werden in ein digitales Objekt Daten über dessen Käufer eingebettet, beispielsweise um die (illegale) Verbreitung von Kopien nachzuvollziehen, bezeichnet man den Prozeß als Fingerprinting.

Das Objekt, welches z.B. ein Text, eine Grafik, eine Audiodatei bzw. ein Video sein kann, wird dabei möglichst *robust* (resistent gegenüber Transformationen) und *nicht beeinträchtigend* gekennzeichnet, ähnlich wie Papier mit einem Wasserzeichen (daher Watermark). Das Kennzeichen muß trotzdem *nachweisbar* sein. Damit sind schon die Kriterien gegeben, die ein Watermark zu erfüllen hat.

1.3 Vergleich

Die Gütekriterien von Watermarking ähneln in gewissem Maße denen, die bei steganographisch geschützter Kommunikation zu erfüllen sind, weisen dazu aber auch charakteristische Unterschiede auf, wie folgender Vergleich (Tabelle 1) zeigt.

Steganographie zur vertraulichen Kommunikation	**Watermarking** zum Schutz von Rechten an digitalen Objekten
Fehlerfreie Übertragung der eingebetteten (zu schützenden) Nachricht	**Robustheit** des eingebetteten Watermarks
Unauffälligkeit im Träger	**Beeinträchtigungslosigkeit** gegenüber dem (zu schützenden) Objekt
Nichtnachweisbarkeit der eingebetteten Nachricht durch Dritte	**Nachweisbarkeit** des Watermarks durch Dritte (bedingt Offenlegung des Schlüssels)

Tabelle 1. Gegenüberstellung von Watermarking und Steganographie

Analog zur Kryptographie unterscheidet man in der Steganographie verschiedene Angriffsarten, die klassifiziert werden nach den a-priori-Informationen eines Angreifers, also den Informationen, die ein Angreifer *vor* dem Angriff besitzt. Die markierten Felder der Tabelle 2 kennzeichnen die relevanten Angriffe.

	Steganographie	**Watermarking**
Cover-Stegotext-Angriff: Angreifer besitzt sowohl Hülldaten als auch Stegodaten	Aufdecken von Steganographie ist trivial	Watermark ist nutzlos, da dem Angreifer das unmarkierte Original vorliegt
Stegotext-only-Angriff: Angreifer besitzt nur Stegodaten	Regelfall	Regelfall (Angreifer kennt Urheber nicht)
Embedded-Stegotext-Angriff: Angreifer kennt eingebetteten Text und Stegodaten	irrelevant, da ja einzubettender Text geschützt werden soll	Regelfall (Urheberinformation ist dem Angreifer bekannt)

Tabelle 2. Klassifikation von Angriffen

2 Steganographie zur vertraulichen Kommunikation

Unabhängig von der Güte existierender steganographischer Systeme, auf die im nächsten Abschnitt eingegangen wird, müssen bei der Verwendung von Steganographie folgende Randbedingungen eingehalten werden:

1. Das (digitale) Original, also die Hülle (*cover*), muß nach der erfolgten Einbettung unwiederbringlich vernichtet werden. Durch einen einfachen Vergleich des Originals mit der veränderten Hülle (*stegotext*) wäre sonst ein Aufdecken der geheimen Kommunikation möglich.
2. Aus 1. folgt auch: Eine Hülle darf nie mehrmals verwendet werden, um unterschiedliche geheime Botschaften zu transportieren.

2.1 Brechen steganographischer Systeme

Das Brechen steganographischer Systeme ist zweistufig (vgl. [ZFPW_97]):

- Stufe 1: Erkennen, daß etwas eingebettet wurde. Dies bedeutet noch nicht, daß die geheime Botschaft offengelegt wurde. Dies erfolgt erst in Schritt 2:
- Stufe 2: Offenlegen der geheimen Botschaft.

Ein steganographisches System, das gemäß der Stufe 1 gebrochen wurde, ist unnütz und erfüllt seinen Zweck nicht. Will man sich lediglich sichern gegen ein Brechen der Stufe 2, ist die Wahl eines kryptographischen Verschlüsselungssystems angebracht, da es seine Aufgabe effizient erfüllt und kryptographische Systeme weitaus besser untersucht sind als steganographische.

Gute steganographische Systeme zeichnen sich mindestens durch folgende Eigenschaften aus:

- Der Algorithmus ist vollständig offengelegt.
- Es erfolgt eine Parametrisierung durch einen steganographischen Schlüssel.
- Die Einbettung beruht auf indeterministischen Effekten natürlichen Ursprungs (z.B. Quantisierungsrauschen, natürliche Störungen etc.)

Leider existiert bisher kein Beweis der Sicherheit eines Systems. Es läßt sich jedoch zeigen, daß unter bestimmten Umständen informationstheoretisch sichere Steganographie möglich ist (vgl. [KlPi_97]).

In [West_97] wurden einige Untersuchungen zur Güte existierender, frei (im Internet) verfügbarer steganographischer Systeme vorgenommen. Es zeigte sich, daß die Systeme meist schlecht sind, d.h. einfach gebrochen werden können. Das Wissen über die Sicherheitslücke führt in vielen Fällen zu deren unmittelbarer Beseitigung. Am Beispiel des Algorithmus Jsteg soll dies deutlich gemacht werden. Weitere Angriffe sind in [West_99] zu finden.

2.2 Beispiel Jsteg

Jsteg ist im Internet beispielsweise unter ftp://ftp.funet.fi/pub/crypt/steganography/ zu finden. Der Algorithmus von Jsteg basiert auf der weit verbreiteten Jpeg-Kompression. Die verlustbehaftete Kompression nach dem Jpeg-Verfahren ist besonders geeignet für Bildinformation mit fließenden Helligkeits-

und Farbübergängen, z.B. digitale (oder digitalisierte) Fotografien. Sie beruht auf der diskreten Kosinustransformation (DCT) und zerlegt ein Pixelbild in einzelne Bildblöcke, meist aus 8x8 Pixeln bestehend, die dann in sog. DCT-Koeffizienten transformiert werden. Ein Koeffizient in einem Jpeg-Datenstrom repräsentiert einen Anteil am Farb- bzw. Helligkeitsverlauf im betrachteten Bildblock.

Das Einbetten bei Jsteg beruht auf dem einfachen Überschreiben der niederwertigsten Informationsbits der DCT-Koeffizienten, d.h. von Bits mit einer sehr geringen Bedeutung für den visuellen Eindruck. Die niederwertigsten Bits der DCT-Koeffizienten werden fortlaufend durch die geheime Botschaft ersetzt.

2.3 Typische Angriffe

Es wurden folgende Angriffe durchgeführt:

1. Visuelle Analyse, d.h. sehr genaues Betrachten verschiedener Bilder, Berechnen von Histogrammen über Farbverteilung, Helligkeit etc. Die durchgeführten Analysen lieferten keine Anhaltspunkte.

2. Häufigkeitsverteilung der niederwertigsten Bits. Es zeigte sich, daß in einem reinen Jpeg-Datenstrom, also ohne Steganographie, ein leichtes Übergewicht von Einsen in den niederwertigsten Bits zu finden ist. Folglich müßte beim Einbetten, d.h. beim Überschreiben der niederwertigsten Bits, darauf geachtet werden, daß die Häufigkeiten von Einsen und Nullen erhalten bleiben. Die Steganographie nach Jsteg beachtet diesen Sachverhalt jedoch nicht, was zur Folge hat, daß beispielsweise bei der Einbettung einer zur Erhöhung der Sicherheit bereits (vor)-verschlüsselten Nachricht sich die Häufigkeiten von Null und Eins ausgleichen (siehe Abbildung 2).

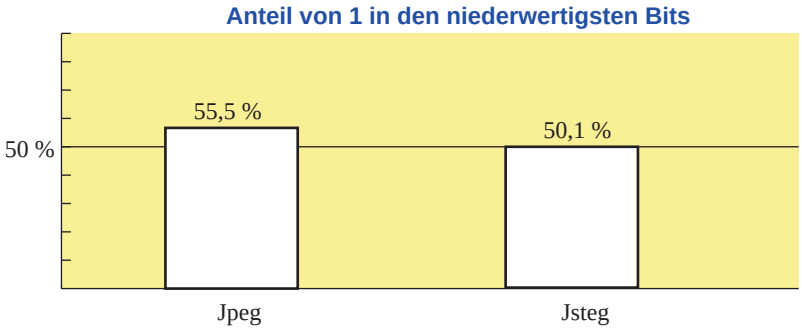


Abbildung 2. Häufigkeitsverteilung der niederwertigsten Bits

3. „Treppenangriff“. Bei diesem Angriff werden nun nicht nur die niederwertigsten Bits, sondern die DCT-Koeffizienten als Ganzes betrachtet. Bei einem normalen Jpeg-Bild zeigt sich eine typische Häufigkeitsverteilung der DCT-Koeffizienten, die in der Abbildung 3a dargestellt ist. Berechnet man nun die Häufigkeitsverteilung für ein mit dem Jsteg-Algorithmus verändertes Bild, zeigt sich ebenfalls ein typischer Verlauf, der jedoch erheblich von dem bei Jpeg abweicht. Es entstehen Treppenstufen. Dies deckt die Verwendung des Jsteg-Algorithmus auf, führt jedoch nicht unmittelbar zum Aufdecken der geheimen Botschaft. Der Grund für die auffällige Veränderung der Häufigkeitsverteilung liegt im Ausgleich der Häufigkeiten benachbarter Koeffizienten, bedingt durch das einfache Überschreiben ohne Beachtung der höherwertigen Bits.

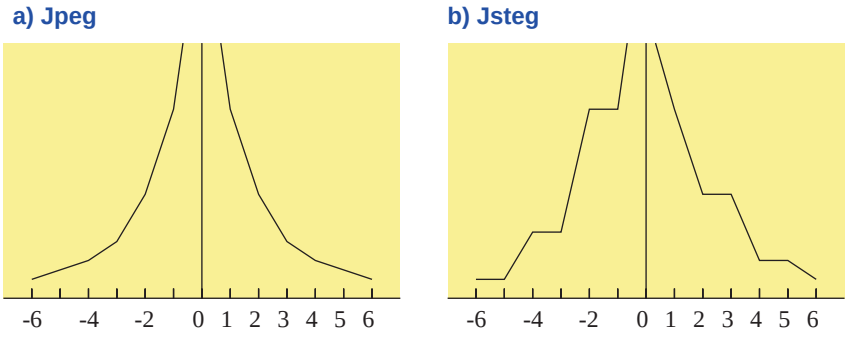


Abbildung 3. Häufigkeitsverteilungen der DCT-Koeffizienten

Der beschriebene Treppenangriff ist nicht nur auf Jsteg anwendbar, sondern auf alle steganographischen Algorithmen, die lediglich das niederwertigste Bit überschreiben.

2.4 Verbesserung und Abwehr des Angriffs

Wenn man sich überlegt, wie die Einbettungsfunktion arbeitet, wird klar, warum sich die Häufigkeiten benachbarter Koeffizienten ausgleichen, sofern man davon ausgeht, daß der einzubettende Text eine Gleichverteilung an Nullen und Einsen besitzt. Diese Annahme ist sicher nicht allgemeingültig, in der Praxis jedoch eine sinnvolle Annahme, da z.B. verschlüsselter Text diese Annahme erfüllt. In vielen Algorithmen wird daher zunächst der Klartext verschlüsselt und dann ohne die Verwendung eines weiteren Geheimnisses in die Hülle eingebettet.

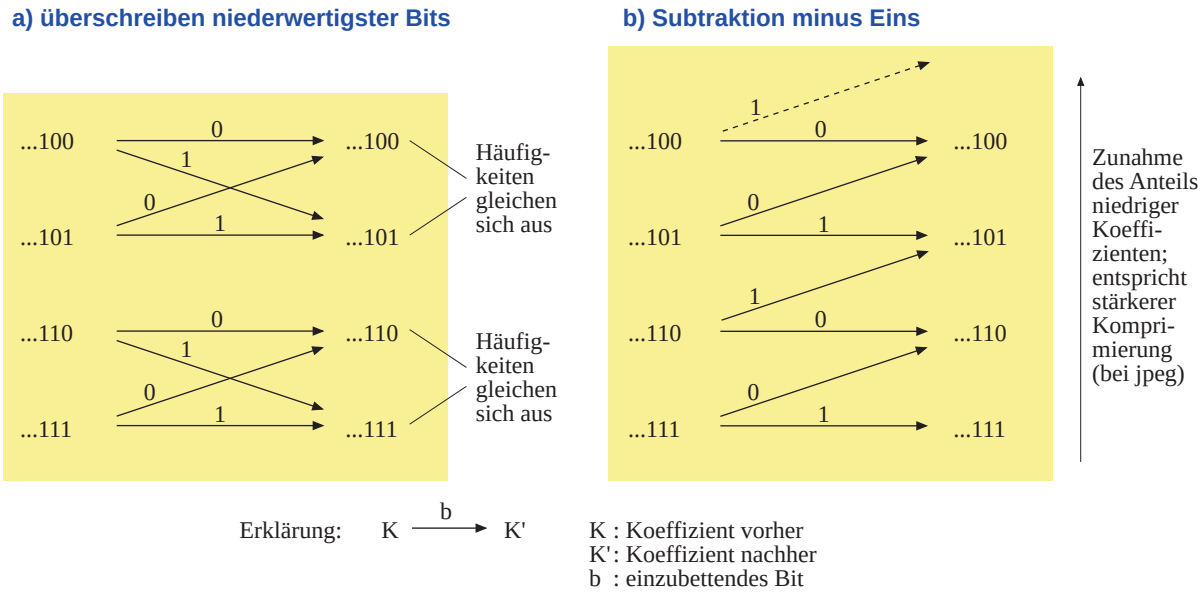


Abbildung 4. Zwei Einbettungsfunktionen

Abbildung 4a zeigt eine Einbettungsfunktion zum Überschreiben der niederwertigsten Bits. Der einzubettende Text wird bitweise in die jeweils im Bild auftretenden Koeffizienten eingebettet.

Die Kenntnis dieses Angriffs führt unmittelbar zu einer leichten Modifikation des Jsteg-Algorithmus: Wählt man anstelle des Überschreibens als Einbettungsfunktion eine Subtraktion minus Eins (Abbildung 4b), falls das niederwertigste Bit verändert werden muß, gleichen sich die Häufigkeiten benachbarter Koeffizienten nicht aus, sondern es ist lediglich eine Zunahme der zahlenmäßig niedrigen Koeffizienten zu verzeichnen. Diese kann jedoch ebenfalls ihre Ursache haben in einer stärkeren Komprimierung der Hülle.

Damit ist für einen Angeifer wieder keine Entscheidbarkeit vorhanden, ob Steganographie eingesetzt wurde. Dieses Beispiel soll zeigen, daß mit dem Wissen über die Schwächen häufig eine Verbesserung der Algorithmen möglich ist.

3 Watermarking zum Urheberschutz

Die Zielstellungen des Watermarking unterscheiden sich grundlegend von denen der vertraulichen Kommunikation mittels Steganographie. Letztere soll die Existenz einer Kommunikation (und damit insbesondere auch deren Vertraulichkeit) schützen, während Watermarking Rechte, insbesondere Urheber- oder Eigentumsrechte, an digitalen Objekten schützt. Die Bedeutung eines solchen Schutzes digitaler Objekte, die ja identisch kopierbar und damit in Originalform verbreitbar sind, gewinnt besonders bei kommerzieller Nutzung digitaler Medien an Bedeutung.

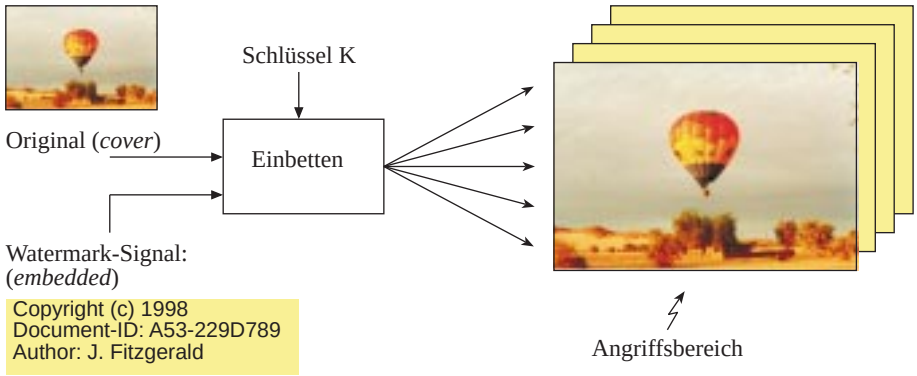


Abbildung 5. Distribution eines markierten digitalen Objekts

Um zu gewährleisten, daß ein mit einem Watermark versehenes Objekt nicht unberechtigterweise so transformiert werden kann, daß das Watermark dabei entfernt wird und das Original (nahezu) unversehrt zurückbleibt, muß das Watermark eine große Widerstandsfähigkeit gegen Bearbeitungsschritte wie Formatkonvertierung, verlustbehaftete Kompression, Filterung, Resampling, Ausschneiden von Bildteilen, Rotation, Skalierung, Spiegelung etc. aufweisen.

Ein Angreifer soll selbst dann, wenn er im Besitz mehrerer Kopien eines markierten Objektes ist, nicht in der Lage sein, das Watermark zu entfernen.

Es ist klar, daß das Watermark möglichst so in das digitale Objekt eingebracht werden muß, daß es nicht zu Beeinträchtigungen des Dokumentes kommt. So sollten Watermarks in Grafiken bzw. Videos nicht sichtbar, in Sounddateien nicht hörbar sein.

Gegenwärtige Watermarkingtechniken arbeiten bevorzugt im Frequenzbereich (neben Raum- und Zeitbereich), den die digitalen Objekte abdecken [CKLS_96, SmCo_96, Zhao_97]. Dazu werden sog. Spread Spectrum Techniken eingesetzt. Die Information, die das Watermark trägt, ist vom Einsatzzweck (Urheberrechts- oder Eigentumsrechtsschutz) und der zur Verfügung stehenden Infrastruktur abhängig. Das kann z.B. eine Dokument-ID sein oder der Name des Autors.

3.1 Was ist Spread Spectrum?

Folgendes Problem ist aus dem militärischen Bereich der Funktechnik bekannt: Eine Kommunikation per (mobilem) Funk soll zwischen verschiedenen Abteilungen einer militärischen Einheit stattfinden. Normalerweise erfolgt die Sendung auf einer bestimmten Frequenz (bzw. in einem Frequenzband) mit einer bestimmten Leistung und Bandbreite. Abbildung 6 zeigt das Leistungssprektrum einer solchen schmalbandigen Sendung.

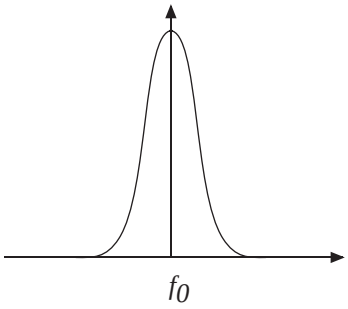


Abbildung 6. Schmalbandiges Senden

Die Folge ist, daß eine deutliche Energiezunahme um die mittlere Frequenz f_0 zu verzeichnen ist. Dies führt zur

- Beobachtbarkeit des Sendens, da ein Spektrumanalysator die Energiezunahme registriert, und
- Peilbarkeit des Senders, da elektromagnetische Wellen Richtungsinformation in sich tragen.

Außerdem könnte ein Gegner die Kommunikation verhindern, indem er mit einem leistungsfähigen Störsender das Frequenzspektrum um f_0 herum stört.

Ein Ausweg zur Beseitigung der genannten Probleme sind Bandspreizverfahren (Spread Spectrum Systems). Das Nutzsignal wird bei der Bandspreizung in einem Spreizmodulator spektral gespreizt, bevor es hochfrequent moduliert und übertragen wird. Auf der Empfängerseite wird analog verfahren. Durch die Demodulation mit der breitbandigen Spreizsequenz wird das Signal spektral entspreizt.

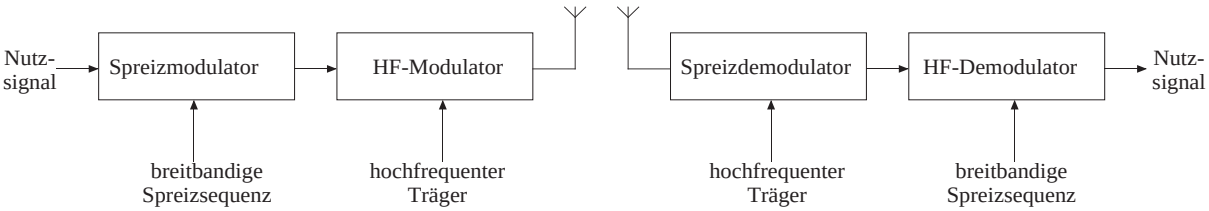


Abbildung 7. Übertragungsmodell bei Spread Spectrum von Funksignalen

Als breitbandige Spreizsequenzen kommen bestimmte Codes zur Anwendung (z.B. Walsh-Funktionen), aber auch rauschähnliche Sequenzen, die PN-Codes (Pseudo-Noise-Code). An die Synchronisation der Spreizsequenz zwischen Sender und Empfänger werden hohe Anforderungen gestellt, da sonst das Signal nicht wieder zurückgewonnen werden kann.

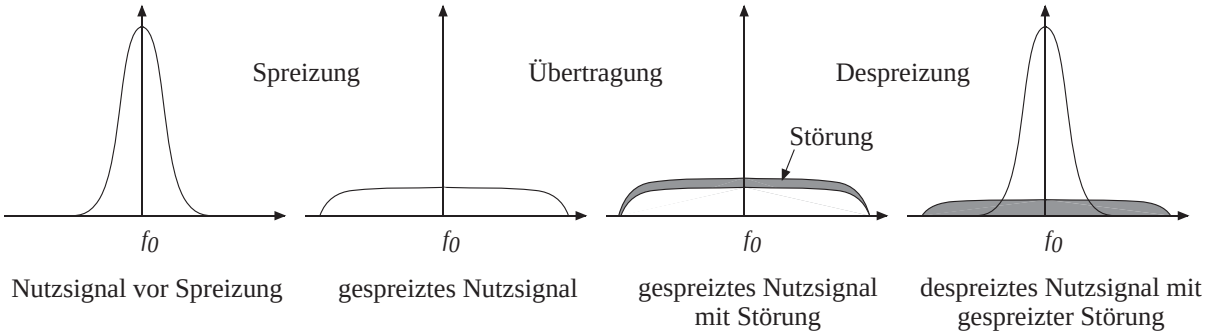


Abbildung 8. Spreizung und Despreizung

3.2 Spread Spectrum Systeme und Watermarking

Durch die spektrale Spreizung erreicht man folgende Eigenschaften, die im Zusammenhang mit Watermarkingsystemen ebenfalls von Bedeutung sind:

- Das breitbandige gespreizte Nutzsignal ist wenig anfällig gegen starke schmalbandige, aber auch gegen schwache breitbandige Störungen. Bezogen auf Watermarkingsysteme bedeutet das, daß
  - starke Änderungen eines digitalen Objektes an wenigen Stellen (z.B. Ausschneiden von Bildteilen) und
  - geringe Änderungen des gesamten digitalen Objektes (z.B. Verrauschen, Kippen, Drehen etc.)

  nicht zu einem Verlust des Watermarks führen.
- Das breitbandige gespreizte Nutzsignal verschwindet durch die Spreizung im Rauschen. Bezogen auf Watermarkingsysteme bedeutet das, daß das Watermark im digitalen Objekt nicht zu erkennen ist, solange man die Spreizsequenz nicht kennt.

Als Watermark wird also ein schmalbandiges Signal in einen breitbandigen Kanal eingebettet. Das schmalbandige Signal ist die z.B. Urheberinformation, der breitbandige Kanal ist das digitale Objekt.

Ein Watermark soll robust sein gegenüber geometrischen Verzerrungen, Veränderungen und Störungen des Objektes und gegenüber Signalmanipulationen bzw. -störungen, genauer gegen

- Digital-Analog-Wandlung,
- Analog-Digital-Wandlung,
- wiederholtem Sampling,
- wiederholter Quantisierung,
- Dithering (Farbanpassung),
- Kompression,
- Rotation,
- Translation,
- Cropping (Ausschneiden von Bildteilen) und

- Skalierung (Größen- bzw. Auflösungsänderung).

Es muß für einen Angreifer außerdem schwer sein, ein Watermark aus dem Objekt zu entfernen, selbst wenn er mehrere unabhängig markierte Kopien des Objektes besitzt.

Herkömmliche steganographische Systeme (insbesondere solche zum Schutz der Vertraulichkeit) sind nicht in der Lage, die genannten Robustheitsanforderungen zu erfüllen. Sie beruhen meistens auf dem einfachen Überschreiben der niederwertigsten Bits der digitalen Hülle. Ein Angreifer könnte also durch einfaches erneutes Überschreiben der niederwertigsten Bits die eingebettete Nachricht manipulieren. Damit ist ein solches System ungeeignet zum Watermarking. Dieses Problem haben Spread Spectrum Watermarkingsysteme nicht oder zumindest in geringerem Maße.

Leider sind, obwohl die Anforderungen bekannt sind, die wenigsten existierenden Systeme sicher gegen gezielte Angriffe, d.h. das Entfernen des Watermarks. In [PeAK_98] ist beispielsweise ein Programm (StirMark) beschrieben, mit dem man Watermarks aus Objekten entfernen kann.

3.4 Prinzip des Spread Spectrum Watermarking

Die folgende Abbildung zeigt das Übertragungsmodell eines Spread Spectrum Watermarkingsystems.

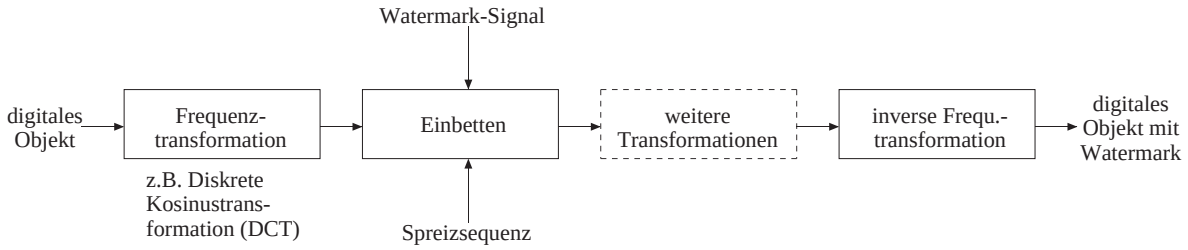


Abbildung 9. Übertragungsmodell des Watermarkingsystems

Um die Veränderungen anschaulich zu machen, die ein digitales Objekt erfährt, soll ein einfacher Algorithmus für das Markieren eines Bildes erklärt werden.

Das Bild wird zunächst einer diskreten Kosinustransformation (DCT) unterworfen. Eine solche Transformation ist besonders effizient für Bildinformation mit fließenden Helligkeits- und Farbübergängen, z.B. digitale (oder digitalisierte) Fotografien. Die DCT arbeitet auf einzelnen Bildblöcken, meist aus einem Pixelquadrat bestehend. Ein solcher Bildblock wird in sog. DCT-Koeffizienten transformiert. Ein Koeffizient in einem frequenztransformierten Datenstrom repräsentiert einen Anteil am Farb- bzw. Helligkeitsverlauf im betrachteten Bildblock.

Im Einbettungsprozeß werden die Informationsbits mittels der Spreizsequenz auf die DCT-Koeffizienten verteilt. Die Spreizsequenz bildet analog zur Verschlüsselung den Schlüssel, während das Watermark die einzubettende Information ist. Um das Watermark robust gegen die beschriebenen Angriffe zu machen, wird ein Informationsbit nicht nur an einer Stelle (bzw. in einen DCT-Koeffizienten) eingebracht, sondern an vielen. Die Anzahl von Einbettungen pro Informationsbit wird durch den Spreizfaktor s bestimmt. Die Spreizsequenz bestimmt, in welche DCT-Koeffizienten (an welcher Stelle) eingebettet wird.

Die Robustheit des Watermarks kommt dadurch zustande, daß im Nachweisprozeß über die Spreizsequenz die s eingebetteten Bits pro Informationsbit ausgelesen werden und über einen Schwellwertverstärker das Informationsbit rekonstruiert wird.

Vereinfachtes Beispiel: Ein Informationsbit des Watermark-Signals wird auf s=10 Bildpunkte verteilt (gespreizt). Die Spreizsequenz bestimmt die Positionen (Bildpunkte), an denen eingebettet wird. Die Einbettung erfolgt durch Überschreiben des niederwertigsten Bits des jeweiligen Bildpunktes.

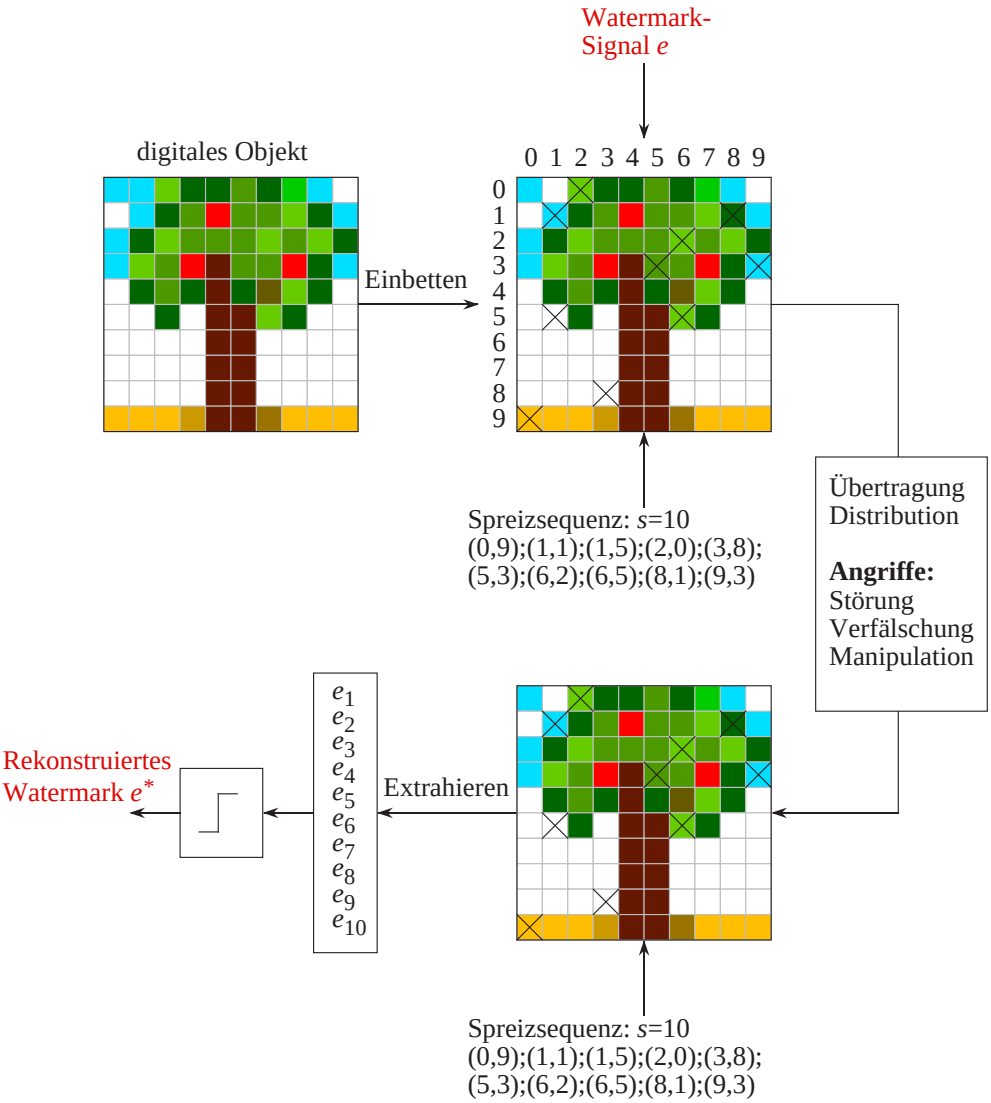


Abbildung 10. Beispiel für eine Einbettung und Extraktion

Bei der Extraktion wird über die s=10 ausgelesenen Bits der Durchschnitt gebildet und auf die nächste ganze Zahl gerundet. Ist mehr als die Hälfte der ausgelesenen Bis zerstört, kann das Informationsbit nicht rekonstruiert werden.

Anmerkung zum Beispiel: Das Beispiel ist deshalb vereinfacht, um die Spreizung anschaulich zu machen. Ein derartiges Watermark ist nicht robust, weil auf eine Wandlung des Bildes in den Frequenzraum verzichtet wurde. Man kann sich das leicht an folgenden zwei Angriffen überlegen:

1. Verschiebt man das Bild lediglich um 1 Pixel nach links oder rechts, läßt sich die Spreizsequenz beim Extrahieren des Watermarks nicht mehr synchronisieren.
2. Wird das Bild breitbandig gestört, z.B. durch Überlagerung von weißem Rauschen, werden alle niederwertigsten Bits überschrieben und das Watermark ist verloren.

3.5 Spread-Spectrum-Modulationsschema allgemein

Um die Einbettungsfunktion etwas formaler fassen zu können, soll noch ein allgemeines Modulationsschema angegeben werden.

Es soll ein markiertes Objekt D(x,y) durch pixelweise Addition des originalen Objektes N(x,y) mit der breitbandigen Sequenz S(x,y) entstehen. Dabei enthält S(x,y) die Watermarkinginformation.

$$D(x,y) = N(x,y) + S(x,y)$$

Jedes Informationsbit b_i des Watermarks wird in S(x,y) repräsentiert durch eine sog. Basisfunktion ϕ_i. S(x,y) ergibt sich somit nach:

$$S(x,y) = \sum_i b_i \, \phi_i(x,y).$$

Die Basisfunktionen $\phi_i(x,y)$ sollten orthogonal zueinander sein. Sie sind im einfachsten Fall unabhängig voneinander gebildete Pseudozufallszahlen.

Zur Extraktion des Watermarks werden die (möglicherweise verfälschten) Informationsbits des Watermarks nach

$$o_i = \sum_{x,y} D(x,y) \, \phi_i(x,y)$$

rekonstruiert. Die Werte der Bits b_i müssen schließlich noch über einen Schwellenwert (*decision threshold*) aus den o_i gebildet werden. Als geeigneter Schwellenwert wird beispielsweise der Median des maximalen und minimalen Wertes der o_i verwendet.

Rechenbeispiel: Es soll in 3x3-Pixel ein Watermark b aus zwei Bit b=(0, 1) hinein markiert werden.

Es sei das bereits frequenztransformierte Original

$$N(x,y) = \begin{pmatrix} 8 & 6 & 4 \\ 5 & 3 & 1 \\ 6 & 2 & 0 \end{pmatrix}.$$

Die Basisfunktionen ϕ_i seien

$$\phi_1 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} \text{ und } \phi_2 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}.$$

Die b_i werden transformiert in Werte aus {–1, 1}. Ein Null-Bit wird also auf »–1« abgebildet, ein Eins-Bit auf »1«. Somit sind

$$b_1 = -1 \text{ und } b_2 = 1.$$

1. Es wird S(x,y) berechnet:

$$\begin{aligned} S(x,y) &= b_1 \cdot \phi_1 + b_2 \cdot \phi_2 \\ &= -1 \cdot \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & -1 & 1 \\ 0 & -1 & 0 \\ 1 & -1 & 0 \end{pmatrix} \end{aligned}$$

2. Einbetten: Berechnen von D(x,y):

$$\begin{aligned} D(x,y) &= N(x,y) + S(x,y) \\ &= \begin{pmatrix} 8 & 6 & 4 \\ 5 & 3 & 1 \\ 6 & 2 & 0 \end{pmatrix} + \begin{pmatrix} 1 & -1 & 1 \\ 0 & -1 & 0 \\ 1 & -1 & 0 \end{pmatrix} = \begin{pmatrix} 9 & 5 & 5 \\ 5 & 2 & 1 \\ 7 & 1 & 0 \end{pmatrix} \end{aligned}$$

3. D(x,y) wird distribuiert und gestört. Es wird hier eine Störung angenommen, die zur teilweisen Auslöschung der Information führt (untere Zeile gleich 0).

$$\tilde{D}(x,y) = \begin{pmatrix} 9 & 5 & 5 \\ 5 & 2 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

4. Extraktion: Berechnen der o_i

$$o_1 = \sum_{x,y} \tilde{D}(x,y) \cdot \phi_1 = \sum_{x,y} \begin{pmatrix} 9 & 5 & 5 \\ 5 & 2 & 1 \\ 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} = \sum_{x,y} \begin{pmatrix} 5 & 19 & 5 \\ 2 & 8 & 1 \\ 0 & 0 & 0 \end{pmatrix} = 40$$

$$o_2 = \sum_{x,y} \tilde{D}(x,y) \cdot \phi_2 = \sum_{x,y} \begin{pmatrix} 9 & 5 & 5 \\ 5 & 2 & 1 \\ 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix} = \sum_{x,y} \begin{pmatrix} 19 & 8 & 0 \\ 8 & 0 & 6 \\ 0 & 0 & 0 \end{pmatrix} = 47$$

5. Schwellenwert bestimmen: (40+47) : 2 = 43,5

$$\begin{aligned} o_1 < 43{,}5 &\rightarrow b_1 = 0 \\ o_2 > 43{,}5 &\rightarrow b_2 = 1 \end{aligned}$$

Ergebnis: Trotz einer Verfälschung des markierten Objektes konnte das Watermark rekonstruiert werden.

3.6 Einbettung von Watermarkingsystemen in ein organisatorisches Umfeld: verbleibende Angriffe

Es wurde das Grundprinzip des Spread Spectrum zum Einbetten von Watermarking-Informationen in digitale Objekte dargestellt. Einige Probleme löst das Einbetten von Urheberinformation jedoch nicht.

So könnte sich jemand eine Kombination aus selbst gewähltem Watermark b und Basisfunktionen ϕ „zusammenbasteln“, so daß er ein Objekt N als seines ausgeben könnte, obwohl er es in Wirklichkeit nie markiert hat. Abhilfe schafft hier nur das *Registrieren* des Watermarks (konkret der Basisfunktionen ϕ bzw. besser der Sequenz S) bei einer Registrierungsstelle, ähnlich wie man das von der Zertifizierung der öffentlichen Testschlüssel (einschließlich Zeitstempel) von der digitalen Signatur kennt.

Leider sind alle bisher existierenden Watermarkingsysteme symmetrische Systeme, d.h. zum Extrahieren und Testen des Watermarks benötigt man ebenfalls die „Schlüssel“, d.h. die Basisfunktionen ϕ. Dies schafft zwei weitere Probleme:

1. Die Registrierungsstelle ist jetzt in der Lage, das Watermark aus dem markierten Objekt D zu entfernen, um so das nicht markierte Dokument N zu erhalten. Hierzu muß sie lediglich $D - S = N$ berechnen. Eine korrupte Registrierungsstelle könnte also das unmarkierte Objekt weiterverbreiten.
2. Die Registrierungsstelle kann mögliche Basisfunktionen ϕ_i aus S und den b_i berechnen, sofern sie nur S besitzt.

Insgesamt besitzt also die Registrierungsstelle eine recht große Mißbrauchsmöglichkeit, solange keine asymmetrischen Stegosysteme (in Analogie zu asymmetrischen Kryptosystemen, wie z.B. das bekannte RSA-System) existieren.

4 Schlußbemerkungen

Die Darstellungen zeigen, daß die „Wissenschaft vom Verstecken von Nachrichten“ (Steganographie) gerade begonnen hat, sich ernsthaft und kritisch mit ursprünglich spielerisch und ad hoc entstandenen Verfahren auseinanderzusetzen. Die „kritische Masse“ war vor etwa 2 bis 3 Jahren erreicht, als in der Politik ernsthaft ein Kryptoverbot bzw. eine Reglementierung von Kryptographie diskutiert wurde.

Ziel des Watermarking ist es, digitale Objekte robust zu kennzeichnen. Dies wurde am Beispiel von Spread Spectrum Watermarking gezeigt. Die verwendete Technologie ist angelehnt an die aus der Funktechnik bekannte spektrale Spreizung von Funksignalen, um sie gegen Funkstörungen unempfindlicher zu machen.

Obwohl das Spread Spectrum Watermarking nicht zur vertraulichen Kommunikation entwickelt wurde, ist es auch hierfür geeignet, allerdings natürlich mit noch geringerem Durchsatz, dafür aber auch störungsgeschützter. Daß dies ein durchaus relevanter Fall sein kann, zeigen Überlegungen, wie Steganographie im Falle eines staatlichen Kryptoverbots ebenfalls verhindert werden soll: Da bei den meisten bekannten Stegoalgorithmen die vertrauliche Botschaft in den niederwertigsten Bits der Hülle untergebracht ist, meist durch einfaches Überschreiben der Originalbits, könnte man die niederwertigsten Bits der übertragenen Hülle einfach stören und würde damit auch die eingebettete Nachricht (zer)-stören. Ein solches Vorgehen führt zu einem geringfügigen Qualitätsverlust der Hülle (z.B. eines über das Internet übertragenen Bildes). Aus der Sicht des Steganographen führt dieses Vorgehen jedoch

dazu, daß er beispielsweise auf die oben beschriebenen Spreiztechniken ausweicht, die ein Informationsbit über die ganze Hülle verteilen. Solange überhaupt noch Kommunikation möglich ist, läßt sich also auch steganographisch kommunizieren.

5 Literatur

CKLS_96 Ingemar Cox, Joe Kilian, Tom Leighton, Talal Shamoon: A Secure, Robust Watermark for Multimedia. Information Hiding, LNCS 1174, Springer-Verlag, Berlin 1996, 185-206.

KlPi_97 Herbert Klimant, Rudi Piotraschke: Informationstheoretische Bewertung steganographischer Konzelationssysteme. Proc. Verläßliche IT-Systeme (VIS'97), DuD Fachbeiträge, Vieweg 1997, 225-232.

PeAK_98 Fabien A.P. Petitcolas, Ross Anderson, Markus G. Kuhn: Attacks on copyright marking systems. Proc. 2nd Workshop on Information Hiding, April 1998, Portland, LNCS 1525, Springer-Verlag, 1998, 218-238.

SmCo_96 Joshua Smith, Barrett Comiskey: Modulation and Information Hiding in Images. Information Hiding, LNCS 1174, Springer-Verlag, Berlin 1996, 207-226.

West_97 Andreas Westfeld: Steganographie in komprimierten Videosignalen. Diplomarbeit, TU Dresden, Institut für Theoretische Informatik, Juli 1997.

West_99 Andreas Westfeld: Angriffe auf steganographische Systeme. angenommen bei: Verläßliche IT-Systeme (VIS'99), Essen, 21. bis 24. September 1999.

ZFPW_97 Jan Zöllner, Hannes Federrath, Andreas Pfitzmann, Andreas Westfeld, Guntram Wicke, Gritta Wolf: Über die Modellierung steganographischer Systeme. Proc. Verläßliche IT-Systeme (VIS'97), DuD Fachbeiträge, Vieweg 1997, 211-223.

Zhao_97 Jian Zhao: Look, it's not there. Byte 22/1 (1997) 7-12.