Vol.2011-DPS-146 No.29
Vol.2011-CSEC-52 No.29
2011/3/11

# 疑似 Chirp 変換による埋込みと難読化を用いた電子透かし

大関和夫[†]　　魏遠玉[††]

本研究は，まず電子透かしの検出器を難読化してその動作を隠蔽する．次に，その難読化条件によって制約された埋込み方式に対し，埋込み手法の高能率化を図るために，DCT（離散コサイン変換）に変えて新たな Chirp 変換を開発する．この疑似 Chirp 変換は、1 個の変換後の係数に複数の周波数成分が含まれ、画像成分が分散される。圧縮とは異なり、埋込みには、分散することが効果的であり、例えば、平坦部でさえも分散されるため、埋込みが可能になる。疑似 Chirp 変換を作り，小ブロックのデータに対しても、効率が向上した。分布の比較，埋込み・検出の評価を行い良好な結果を得た．

# A New Watermarking System with Obfuscated Embedder and Quasi-Chirp Transform

Kazuo Ohzeki[†]　and YuanYu Wei[††]

Watermark detection software is obfuscated to hide embedding and detection algorithms. Then, to adjust the resultant situation of the block size being limited, due to the obfuscation, a new quasi-chirp transform is developed to improve embedding efficiency. The quasi-chirp transform is different from the conventional DCT or Fourier transform. It contains multiple frequency components for a single transforming basis. It disperses image data rather than compressing, as the DCT does. The dispersed data increases the range for embedding watermarks. The chirp transform is able to embed, even on the flat area of an image. Using this chirp transform, embedding and detection experiments for image data with small block sizes are carried out. An efficiency and a robustness of the watermark are evaluated.

## 1. はじめに

本研究は、まず電子透かしの検出器を難読化してその動作を隠蔽する。電子透かしとして秘密情報と鍵を用いる Semi-Private(Semi-Blind)方式[1-3]を対象とする。まず、それに対する難読化条件によって制約された埋込み方式に対し、埋込み手法の高能率化を図るために、DCT（離散コサイン変換）に変えて新たな疑似 Chirp 変換を開発する。次に、電子透かし方式としての総合的な特性の評価を行っていく。

電子透かしの検出器を難読化する意義は、難読化により、著作権情報等を検出するときに事実上公開される検出器の動作が隠蔽され、埋込み検出の一連の動作も隠蔽され続けることにある。隠蔽されることによって、検出器の動作を解析することによる解明は直接的にはできなくなる。これにより、パラメータの変更でユーザーごとに異なる電子透かし（Fingerprinting)を埋込んだまま、埋込みシステムの基本構成を長期に変更すること無く運用できることになる。この効果は、プラーベートな電子透かしにおいても、耐性向上の一つとして有効である。

ソフトウェアの難読化は，Collberg らの基本的な多方式[4]、Drape の解説［5］，多数の実施例や，Barak[6-7]らの不可能性を証明する考えなど多岐にわたっている．Collberg の多数の実施例は，解読を難しくすることはできるが，解読を課題として取り組めば，比較的少ない手順で面倒だが理論的にはいつかは解読可能なものである．一方，Barak らの難読化不可能論があるが，その理論は，連続関数に限定された場合のものと見做され，離散的な現実の実関数は難読化可能なものがあると考えられる．ここでは，このような分析を進め，難読化技術の詳細化を行い，表関数による究極的な難読化を構成することを試みる．

筆者らは、Barak らの不可能論に対し Barak の論旨と何点かで異なる方式を提案した．提案する方式は，予め全ての計算結果を求め，ROM（Read Only Memory：読み出し専用メモリー）乃至は，ソフトウェアプログラムの固定配列に格納しておくもので，入力と，出力の関係以外は，何も示されていない．難読化では配列化[5]という変数の複雑化を図る手法があるので、それと区別するため、ROM 方式[8]と呼ぶことにする。Barak の方式が Compiler であり、また Virtual な Black-Box であるのに対し、提案する方式は神託機械（Oracle 機械）と同じ形状となるが，実在する White-Box になっている点が異なる。

このような考えに類似の研究は，Chow らの White-Box 難読化[9]にも見られる．Chow 等の方式は，DES 暗号化処理を難読化するもので，処理中のアフィン変換を表形式に

[†]芝浦工業大学工学部情報工学科
ISE, College of Eng., Shibaura-Institute of Technology
[††] 芝浦工業大学 大学院 機能制御システム専攻
Functional control systems, Graduate School of Eng., Shibaura-Institute of Technology

Vol.2011-DPS-146 No.29
Vol.2011-CSEC-52 No.29
2011/3/11

変形している．一方，課題として，表にする処理は部分的で，多くの計算部が残っているため，計算量が多く，遅いことが述べられており，問題となっている．筆者らの提案方式は，メモリサイズは増加するが，処理速度は表の値を参照するステップが基本で，配列構造を多段にした場合は，その段数分だけのステップ数になるが，いずれにしても処理速度は，ほとんど 0 に近く，Chow 等の方式と異なると考えられ，実用性が高いといえる．筆者らは，検出器公開型の電子透かしにおいて，公開する検出器を難読化する方式について提案してきた．難読化で ROM 関数によるもの[8]は，状態数が増加し、全探索する以外に解が無ければ解読を不可能に近く構成できる．

提案する難読化方式である ROM 方式は、結果を提示するまでの時間が難読化前のプログラム実行速度に対し、きわめて早いこと、結果は難読化前のプログラム実行結果と完全に一致することなど難読化の基本条件[5]の 2/3 を満たしている。また、難易度に関しては、入力と出力以外が隠蔽されていることから、これも最大の難易度に達していることが分かる。一方、一つの問題としてプログラムサイズが膨大になる点がある。これが制約条件となり、この難読化手法を電子透かしの検出器に導入する場合、電子透かしとしての埋込み、従って検出のアルゴリズムを工夫する必要が出てくる。以下、電子透かしの新しい埋込み方式に関して述べる。

演算においてプログラムサイズに制限が有るため、画像の値を多用する演算では、サイズがますます増大する。そこで、まず、サイズの小さい演算手法を用いる埋込みから順次検討していくことにする。今回は、4 次と 8 次の直交変換に関し、より高能率化を図ることにする。電子透かしやステガノグラフィー方式[10-12]として、離散コサイン変換（DCT)方式が使用されている。DCT は JPEG ファイルに埋込みが可能であることや JPEG ファイル圧縮攻撃にも適合性がよいことで、多用されているが、実際に 4 次や 8 次の変換では、画像信号に低周波成分が多いという周波数特性により、高次の係数は 0 に近い数になり、埋込みのスペースが少ないことが問題である。また、撮影された一般画像では、空や地面などに平坦なレベルが続くことが多く、これらの DCT 変換後の係数は DC 成分 1 個のみが非ゼロとなり、他の成分はほとんどゼロになって埋込みが十分達成できないという問題があった。

そこで、本研究では、電子透かしやステガノグラフィーにおいて有効な変換を開発していく。そのためには、直交変換の中で変換係数が対称的に均整のとれた形で並んだものではなく、むしろ、非均衡なものが好ましいと考えられる。そのような直交変換として、SLANT 変換[13]がある。また、意図的に傾きを入れた斜交変換[14]という変換もある。ここでは、上記目的に合わせて、埋込みに好適な新たな変換を設計する。これは、非均衡な係数を持つと共に、低域係数の埋込みを柔軟化するため、周波数を多様にする様な設計を組み入れることにする。結果として得られる変換は非均衡で多数の周波数を含む直交変換であり、疑似 Chirp 変換と呼ぶことにする。この疑似 Chirp 変換は、DCT や Fourier 変換と異なり、埋込みを行いたい成分 1 カ所の位置を DC の次から中域までの間の AC 成分に設定することをめざしている。4 次や 8 次程度の変換では、電子透かしの埋込みは耐性の確保と劣化の制限から、1 カ所にある程度大きい透かしを入れることが有効と考えられる。従って、何番目の周波数に埋込むかは、とびとびに離散化された周波数の中から選ぶため、周波数を量子化していることを意味する。それに対し、非均衡の変換係数を使用すれば、連続的にある周波数の中から希望の周波数を選ぶことができるようになる。また、後述するように、非均衡によって、平坦な画像ブロックに対しても、変換後の値の平均をゼロにすることなく、また分散化さえ達成することが可能と考えられる。

本研究では、このような新しい周波数特性を有する正規直交変換作り、電子透かしの埋込み、検出を行い、性能の評価を行った。また、主に JPEG 圧縮による耐性を評価し、DCT 方式との比較を行い、埋込み性能の向上を達成した。

以下、2 章では、関連する従来方式について述べ、その問題点を上げる。3 章では、Chirp 変換の構成を検討して、4 次と 8 次の Chirp 変換を開発する。4 章では、実験と評価を行い、5 章でまとめと今後の課題について述べる。
今後の課題について述べる。

## 2. 関連する従来方式

難読化の対象には，データとプログラムに 2 分される．データでは、文献[15]のように暗号鍵，復号鍵，パスワード，個人情報，秘密定数などがある。プログラムとは，計算機で実行するソフトウェアプログラムのことであるが、一般的に汎用性を有するのは難しく[5][6]、最近では、ウイルスなどの Malware を難読化して隠蔽したものを解明する研究[16]や、堅実な手法で古典的に構成する手法[17]などが行われている．次にプログラムの難読化は全てのプログラムに適用できる汎用型と，特別な機能を有すプログラムに固有の難読化に分けられる．汎用型にはあらゆる関数類が含まれるが，その中には，ごく単純な関数からガンマ関数などの複雑な計算を行う特殊関数まである。もとの関数乃至は演算アルゴリズムがあるレベル以下の簡易なものであるとき，どのような難読化を行っても，入出力の関係の観察により，アルゴリズムを推定することは容易にできる．従って，このように容易な計算に難読化を施すのは効果がない[18]．従って、プログラムの難読化は計算や組み合わせのアルゴリズムが複雑であるかどうかをまず判別し、入出力の関係が分かっても、その内部演算のアルゴリズムを数値計算にて解析しても、十分長期の解析時間がかかるものに対してのみ、難読化する意味がある。

Barak の難読化不能な関数の例には，点関数（Point Function)という定数関数の難読化が登場するが、証明として正しくても，難読化という意味的な例としてふさわしく

Vol.2011-DPS-146 No.29
Vol.2011-CSEC-52 No.29
2011/3/11

ない．つまり，連続領域では，母数が無限で，いくら有限の回数試行しても，x=αというものにたどり着く確率は，限りなく 0 ではあるが，実用領域での離散変数領域では出力がβになるまで探せば良いので，探索問題としては，有限になり簡易なものである．もう一点異なる所は，難読化を Compiler と定義し、難読化後のプログラムに対し，依然として入力から計算して出力を得るという表現をとっているが，筆者らの ROM 関数では，計算ではなく表のデータを検索するだけで，アルゴリズムが観測される元になるような計算動作は行っていない．また，Barak の例では，連続関数についての証明であり，従って入力の変数も連続で，無限個ある．しかし，実際の例では，離散的な関数になり，入力の種類は，有限個の例になる。

筆者らの提案する関数値を事前に計算し、定数表（ROM)として格納しておく方式[8]に類似する方式として、Chow らの White-Box 難読化は，Black-Box 処理と異なり，初めから公開領域で行うことを前提に機密性を持たせることを提案している．これは Black-Box 処理より，難読化変換への課題が増加した分だけ，難易度が高くなったと考えられる．DES を用いた暗号化演算処理を機密化することを目指している．DES 演算のうち，固定の鍵を使った演算部を表にして，鍵のデータを直接表示しないようにしている．演算はアフィン変換という 1 次変換である．小さいサイズのモジュールに分割している．問題点として，演算速度が遅いことと，プログラムサイズが大きくなることをあげている[9]．本論文では、Chow 等のような部分的なアフィン変換方式と異なり、アフィンに限らず全ての関数演算をはじめから終わりまで表にして、難読化としてのあらゆる手がかりをなくすことを行う。

筆者らは，電子透かしの検出器ソフトウェアを難読化することを検討している。この検出器は可変の線形変換と非線形演算である量子化を含んでいるため，未知数に関する多次元の非線形な方程式を解く必要があると考えられる．また，この線形変換自体は，将来非線形に拡張できる可能性を有している。

電子透かしの形式として、検出に原画と鍵を必要とする Private(Non-Blind) Watermark，秘密情報と鍵を必要とする Semi-Private（Semi-Blind), [1] [2][3]ものがある。ここでは、原画から由来する情報を使用し復号する方式を提案しているので、Semi-Blind 型となる。最近の Non-Blind 方式には[19]がある。文献[19]は Naturalness Preserving Transform(NPT)という直交変換を用いている。この変換は、画像の欠落などを補修するもので、電子透かしで発生した劣化を補修することがなされてきた。ロゴや小画像をこの NPT で変換し、カバー画像に埋込む。この NPT は Hartley 変換を 1/2 の部分に 2 倍の周波数を詰め、その後は対称的に折り返した変換を構成している。これは、複素フーリエ変換を実数化したような配置になっている。変換の核となる変換は Hartley 変換に限らず他の直交変換も可能と述べている。この変換は、均整のとれた従来からある DCT 等の直交変換と類似しており、本論文で、提案する疑似 Chirp 変換とは異なっている。

## 3. 提案方式

まず、難読化に関しては、電子透かしとして何らかの変換を行って、量子化によって埋込みを行う方式を、基本とする。変換としては、今回は Chirp 変換による正規直交変換を行う。主たる演算は、この変換と量子化である。直交変換は、行列積であるので、入力の画素値に対し変換係数を乗じて累積していく。そこで、4 個から 8 個の画素ブロックを構成し、これに対応して、4 次と 8 次の線形変換を行う場合、入力と、出力の関係は、図 1 のようになる。

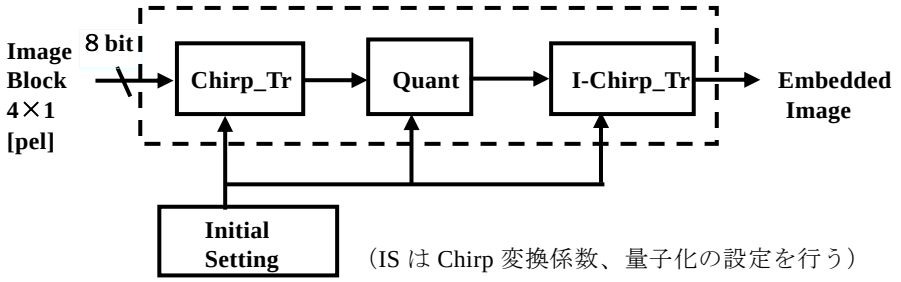



図 1　Input Output Relation of Watermarking.ブロックは 4 画素, 8 ビット/画素

この構成は、計算を表にするため, 線形変換の次数を 4 次又は 8 次に制限してある．この線形変換の変換係数を変化させることで埋込み手法を変え、隠蔽する FingerPrinting の構成ができる。入力画像のモノクロ成分 4 画素を 1 ブロックとし，周波数的な領域へ変換する．中間周波数に相当する 2 個目又は 3 個目の成分を所定の特性で量子化器 Quant で量子化し，逆変換部で逆変換され，画像に戻す．埋込みは量子化後の範囲を調べることにより、有無が判明する。所定の埋込み範囲にあれば、透かしがあり、無ければ無しとなり、埋込みの文字情報が無い場合は、0 ビット埋込みとなる[20]。実際には、埋込みは、複数箇所になされ、その多数決で、優位水準以上に埋込み情報に偏りがあれば、埋込みありとし、偏りが無ければ、無しとなる。偏りは、量子化の場所を 2 種類にすれば、３値となり、無しと 0 ありと 1 ありの 3 種に検出結果が得られる。これは、1 ビットの埋込みになる[21]。

図 2 に 4 画素のデータの変換を 16 ビットから 24 ビットの入力で 2 バイト（16 ビット）出力の 3 個の表により構成する例を示す．これを 4 個の出力に対して 4 種用意する．また，各表の 16 ビットの出力は出力前に値の並べ替えがなされ，この途中結果

Vol.2011-DPS-146 No.29
Vol.2011-CSEC-52 No.29
2011/3/11

の値を観測しても関数関係は推定できないようにする．量子化後の逆変換も同様の構成で実行できる．表 関数の容量の見積もりを表 1 に示す. ROM1 は 128KB で, ROM2,3 が 16MB になる. 量子化器 Quant は最終段の ROM3 に含めることができる. また, ROM1 は ROM2 と統合できるので，変換と量子化までで，1 出力係数につき，32MB となる．逆変換は，各出力を 8 ビットとして，図 2 と同じ構成となり，やはり，ROM1 と ROM2 を統合して，1 出力に対して，32MB となる．以上より，埋込み器の総容量は (32MB+32MB)× 4 出力=256MB となる．検出器の方は，逆変換が不要となるので，32MB× 4 ＝128MB で構成できる．

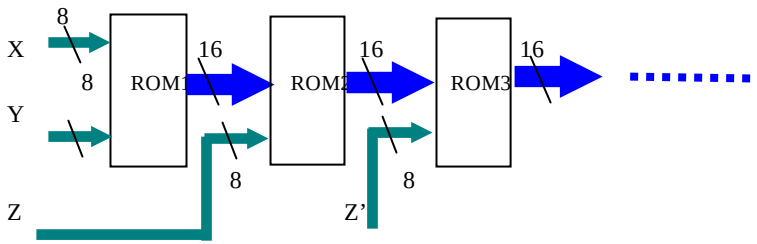



図 2　4 次変換のデータフロー（検出器の場合，埋込みの場合の前半部に相当）

表 1　データ容量の見積もり

| | ROM1 | ROM2 | ROM3 |
|---|---|---|---|
| 入力アドレス | 16 | 24 | 24 |
| 出力 | 16 | 16 | 16 |
| 容量 (bits) | $2^{16}\times16$ $=1048576$ | $2^{24}\times16$ $=16777216$ | $2^{24}\times16$ $=16777216$ |
| 概数 | 128KB | 16MB | 16MB |

このような構成を ROM 方式と呼ぶことにする。一つの変数を複数に分割したり、更に表（Array）に展開する難読化があるので[5]、別名として ROM 方式と呼ぶ。このような ROM 構成の難読化によって埋込み関数の演算を行うとすると、画像ブロックサイズが、当面 4 次か 8 次程度に制約される。例えば、従来からある DCT による変換を使用したとき、4 次では、埋込み位置は、第二成分か第三成分かが候補になる。DC は他の係数より、影響が大きく、通常埋込みを行わない方が良いとされている[22]. しかし、大ブロック係数が多い場合や 2 次元の場合は、埋込み位置は多数の候補があるが、4 次元の場合は、選択肢が無い。DCT 係数は、定義により予め定まった離散的な周波数を表し、画像に適合した周波数乃至は、埋込みに好都合な周波数位置を選ぶことがきない。それならば、DCT という固定の変換を用いずに、第二成分位置に対して最も好適な周波数を有する変換を開発することが有効である。

DCT に対し、それとずれた周波数を生成する候補として、SLANT 変換[13]がある。SLANT 変換はその名の通り斜めに並んだ係数からなる。しかし、文献[13]の SLANT 変換は単に係数が斜めに並んでいるだけで、依然として対称な形式である。4 次の変換を電子透かしの埋込みに用いる場合の問題をより詳しく把握するため、テスト画像を用いて、アダマール変換[18]や DCT 変換を行う予備実験を行った。4 次の DCT を用いた変換後の値の分布は、図 3 のようになっている。この分布は、8 次の JPEG における DCT の変換でも多数示されている[23]。画像によってまた、場所によって分布は異なり、平坦な部分では、DC に集中し、第二成分はほぼゼロになる。このような場合は、埋込みを避けているのが通常の例である[23]。DCT の変換後の AC 係数がゼロになるのは、DCT の変換係数が正負対称で均整がとれているからである。偶数個の画像データでブロックを形成する場合は、変換の方を非均衡にすれば、これが打開できると予想できる。また、画像ブロックを奇数個、例えば 5 個にすれば、DCT であっても必然的に非対称になり均衡は崩れる。ここでは、従来 DCT との直接比較をするためにも、まず 4 次の偶数ブロックのまま、変換の斜交化を行うこととする。埋込み位置を画面全体から広く選べるようにするためには、このような平坦部も埋め込むことができることが望ましい。

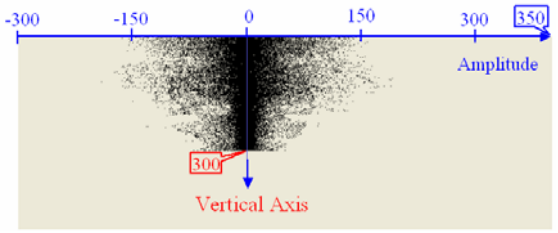



図 3　4 次 DCT の変換後の第二係数の分布

斜交変換としては、画像の圧縮に用いた例がある[14]。斜交になり、画像のブロックが正方の形だけでなく、傾いた斜め線からなるブロックが使用可能になり、しかも斜め線のエッジに対する符号化特性が向上することが確かめられた。また、新規に変

Vol.2011-DPS-146 No.29
Vol.2011-CSEC-52 No.29
2011/3/11

換全体を設計したため、第一成分も均一な係数でなく、曲がっているため、グラデーションのようなやや傾いた平坦部の特性が向上することなどが示されている。この斜交変換は、係数が非均衡であるため、上記の目的に一応適合している。しかし、非均衡であるうえに加えて、上記電子透かしの埋込みの目標は、画像の平坦部も非平坦部も、なるべく広い部分で、埋込みが可能になることである。変換は、入力ブロックの周波数と競合して、変換後の係数の絶対値が大きくなることが望ましい。即ち変換をある周波数に固定すれば、それに適合して、変換後の係数の絶対値が小さい現象が起こりうる。そこで、周波数は、ある固定の最適値があると想定して、それを求めるのではなく、むしろ、1 個の成分に対して、多数の周波数が混ざった変換を実行することがよいことが分かる。以上の考察の元に、Chirp 変換という漸次周波数が上昇していくような変換を導入することにする。Chirp 信号は、アナログ時間信号形式では、

$$C(t) = A \cdot \cos(2\pi f(t) \cdot t + \varphi) \qquad (1)$$

と表される。$f(t)$ が 1 次式のものを線形チャープ信号と呼ぶ。従来の DCT やフーリエ変換などは皆、1 個の基底が単一の周波数から成り、かつ対称性を有している。ここで目指す形状は周波数が複数にわたることと、全体の正負の割合が同じという対称性を有しないことである。このためには、Chirp 信号を正負の割合が一致しないように切り出せばよい。このような形状の波形を電子透かしの埋込みに使用した例として、文献[24]がある。DCT 後の係数を大きさに応じて徐々に緩やかに正負反転する Index 関数により求まる値を透かし値として埋込む。この Index 関数は、Chirp とは逆に周波数が大きくなるほど周波数は低下する。また、2 値に量子化されており、変換では無く、量子化して透かし情報の生成に使用される。また、文献[25]では、乱数から鍵に依存した直交変換を順次作り、変換として使用している。これは、これまでの DCT などの直交変換でなく、推定が難しいと思われる変換を使用している。しかし、変換は乱数を元に設計されており、画像の低域成分の考慮は、設計手順上なされていない。

本論文では、DCT やフーリエ変換の様な低域重視の変換を考慮しつつ、多数の周波数を組み込もうとしている。また、小さいブロックに 1 カ所だけ埋め込むような場合の一箇所の埋込み位置に対応して、効率的な変換基底を設計する。以下 4 次と 8 次の場合の設計を示す。まず、埋込みを第二成分に行うものとして、第二成分を生成する基底を Chirp 変換によって生成する。離散的な 4 点又は 8 点で Chirp 信号を生成するため、DCT の項を参考に低次から中域の周波数を生成するように、係数を定める。高い周波数数は、多くの場合、画像に含まれないため、高い成分を組み込んでも、活用されない。そこで最大でも DCT で言えば中間の周波数に当たる部分までを基にして係数を決める。図 4 に 4 次と 8 次の Chirp 変換の係数例を示す。

8 次の場合を見れば、低域の半波長から中域まで周波数が変化していることが分かる。また、各基底の和は、DCT などのようにゼロになっていない。これが非均衡な直

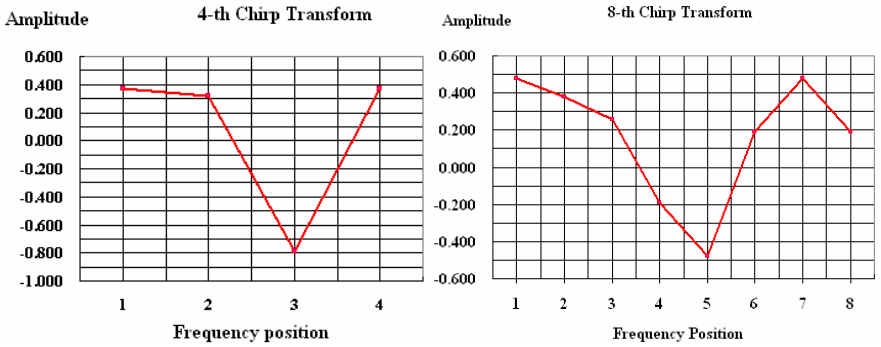


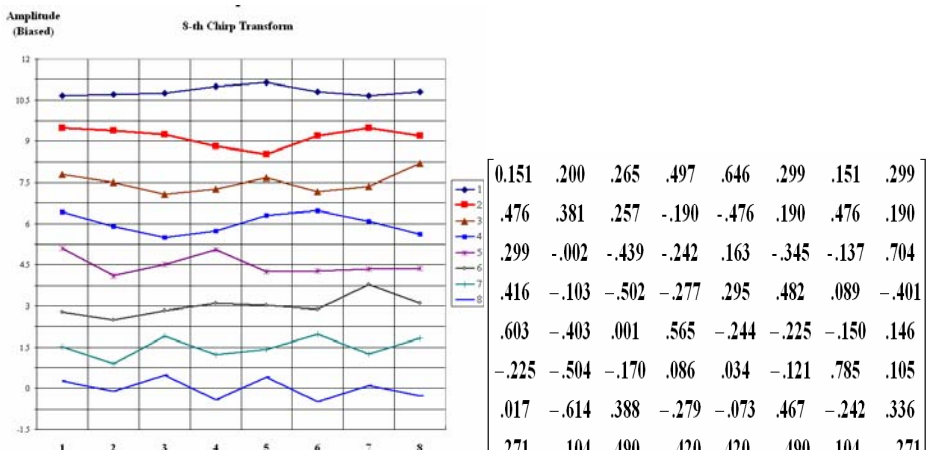

図 4　4 次と 8 次の Chirp 波形図

交変換の第二基底である。次に第一基底を作る。第一基底は通常平坦な直流であったが、必ずしも完全な平坦でなくても直交変換を生成できる。実際画像データから 2 乗誤差最小となる所謂最適な変換 KL 変換を設計すれば、第一成分はかすかに曲がった物になることが多く、完全に平坦な直流になるとは限らない。また、第二成分の形状を優先して先に決定性するため、その係数の合計値が非ゼロであることから、平坦な直流では、直交しないことになる。第一成分は、第二成分と直交する係数で、緩やかな平坦になるように決める。次に第三成分以下を決める。ここで、今回の目的は、画像圧縮などではないので、実際に透かし埋込みに使用する第二成分のみが透かしの埋込みに直接の影響を持つものであり、他の成分は影響度が小さいので、今回は、DCT の第三次以降の係数を種として、持ち込み、Gram-Schumidt の直交化法で正規直交系列に組み上げる。第一成分を適当に作った場合も、Gram-Schumidt の直交化法で、正規化しておく。第三成分は、DCT の第三成分を仮に当て、Gram-Shmidt の直交化法で、正規直交化していく。以下同様に、DCT の係数を借りて、正規直交基底を作り上げる。図 5 にその形状を示す。第二成分以外は、変換に使用されるが、量子化等の変形を受けることなく、透かしを埋込んだ第二成分と共に逆変換される。従って、省略はできないのと、逆変換の時の誤差感度への影響を持つという必要性はあるが、この目的においては、第二成分のみが最も影響が強いので、第二成分の設計に最も注意を払えばよい。また、実際の変換として使用する場合に、係数の絶対値はゼロに近いものは変換において寄与率が小さくなり、情報が無駄になる。そこで、係数の絶対値はなるべく大きい数値になるものが採用されるよう、調整してある。

図 6 にこの変換を用いた時の変換後の係数の分布の様子を示す。先の図 3 と比べると、明らかに分布が広がっていることと、それに伴い、密度が薄くなり、良く分散している様が分かる。実際、空のような平坦部も、実際は、かすかな変動が有るため、

Vol.2011-DPS-146 No.29
Vol.2011-CSEC-52 No.29
2011/3/11

それに呼応して、分布の中心が移動しただけでなく、分散も広がっていることが分かる。全体として変換後は、非ゼロの係数が大幅に減少し、埋込みが画面のより広い部分で埋込みが可能になることが分かる。




| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0.151 | .200 | .265 | .497 | .646 | .299 | .151 | .299 |
| .476 | .381 | .257 | -.190 | -.476 | .190 | .476 | .190 |
| .299 | -.002 | -.439 | -.242 | .163 | -.345 | -.137 | .704 |
| .416 | -.103 | -.502 | -.277 | .295 | .482 | .089 | -.401 |
| .603 | -.403 | .001 | .565 | -.244 | -.225 | -.150 | .146 |
| -.225 | -.504 | -.170 | .086 | .034 | -.121 | .785 | .105 |
| .017 | -.614 | .388 | -.279 | -.073 | .467 | -.242 | .336 |
| .271 | .104 | .490 | .420 | .420 | .490 | .104 | .271 |

図５：8 次 Chirp 変換波形図と係数

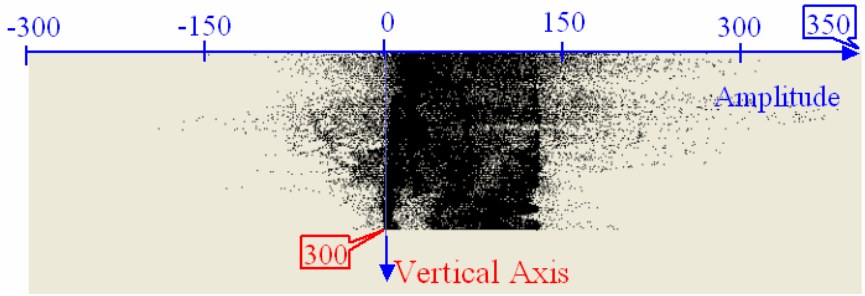



図６　4 次 Chirp 変換後の第二係数の分布

## 4.　実験

図 7 のような構成で、実験を行った。カラー画像の画素成分(R,G,B)は、前処理としてオーバーフロ防止のためヒストグラム分布の調整を行っておく。輝度と色差(Y,I,Q)に色変換され、輝度 Y 成分に透かしが埋め込まれる。ここで、色変換は YCbCr やその他の逆変換が存在するものを用いても良い。RBG から YIQ に変換・逆変換する式は、以下のようになっている。通常 3 桁程度の精度があれば、変換後整数化しても逆変換後の LBS の変化しないが､透かしの埋込みで Y 値が大きく変動するため、精度は高い方が望ましい。

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} = \begin{bmatrix} 1 & 0.954889204321426 & 0.622103935020897 \\ 1 & -0.27135478274584 & -0.647512025865468 \\ 1 & -1.10725100544121 & 1.70246037378756 \end{bmatrix} \begin{bmatrix} Y \\ I \\ Q \end{bmatrix} \quad (3)$$

色差成分.I Q はそのまま保管される。Y 信号の中から 4 又は 8 画素の小ブロックが選択され、DCT 又は Chirp 変換が行われる。第二成分に量子化による埋込みが行われる。量子化による埋込みは、符号化分野でも広く行われてきた基本的な処理で方式というほどのものではない[26]。しかしその後 QIM 方式という名称を附したことにより、説明時に、引用されることが多くなった。量子化は等間隔の幅とランダムな幅など多様なものがあるが、ここでは固定の幅（4,8,16,32,64）を用いた。この量子化が電子透かし埋込みとなる。埋め込みブロックは画像サイズにより、２から 24 箇所になった。埋込み後、逆直交変換され、Ywとなる。更に、保管しておいた I,Q と合わせて逆色変換され、画素値 Rw,Gw,Bw になる。RGB から YIQ への色変換は正則な線形変換で、逆変換により、完全に元のデータ値に戻る可逆な変換である。しかし、透かしを埋込んだ場合は、非可逆である。YIQ 空間は、図８のようになっており、この立体内の点が透かし埋込みで Y 値に変動が出ると、この立体内から外に出て、RGB に逆変換した値が本来の範囲 0 ≦R,G,B≦255　を超えるものになる。この対策として、

[Y1]事後のクリッピングにより、強制的に 0 ≦R,G,B≦255 の範囲に移動してしまう手法と、

[Y2]Y の値を見て、I,Q の値を調節し、図８の範囲内に納まるように I,Q を移動させる手法がある。ここでは、手法[Y2]を I,Q の移動量が 20 以内の範囲で行うこととした。これが、図 7 の Post-Processing である。このあと、逆色変換（ICC）が行われ、RGB の画素値になる。

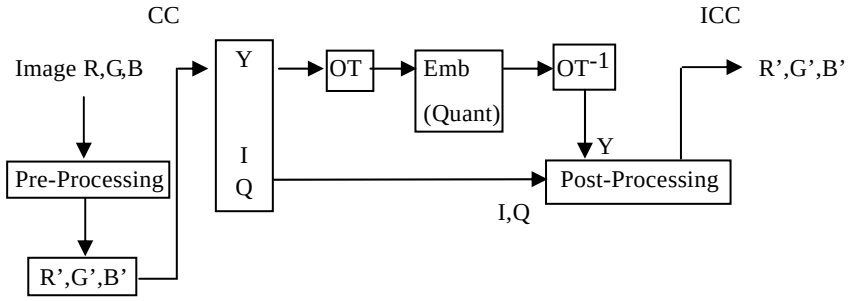
図 9 に攻撃による変形が無い場合、埋め込まれていないところの検出確率を示す。攻撃が無い場合は、検出ができている。埋込み方式が量子化であるので、ランダムな信号に対しても 1/2 の確率で検出がなされる。画像の外枠から少し内側の部分全体に

Vol.2011-DPS-146 No.29
Vol.2011-CSEC-52 No.29
2011/3/11

対し、検出を行い、検出率を求めた。画像によってばらつきはあるが、理論的な検出確率 1/2 に近い値が得られている。これから、今回の最大の画像では、誤検出の確率は、

$1/2^{24}$

となる。図 9 に透かしを埋め込んでいない場合の検出テスト結果を示す。0.5 を中心にばらつきは大きいが、0.35-0.6 の間に収まっている。図 10 に JPEG 圧縮攻撃と、他の波形攻撃の有る場合の検出結果を示す。JPEG 圧縮に対しては、DCT と Chirp 変換でほぼ同程度の耐性になっている。一方、波形攻撃に対しては、Chirp 変換の方がステップサイズで 1 区間、従って 2 倍の耐性を有することが見られる。




CC: Color Conversion
OT: Orthogonal Transform(DCT or Chirp_Tr)
Emb:Embedding of Watermark
ICC:Inverse Color Conversion

図 7 RGB:YIQ:DCT/Chirp:EMB

・ 次に、以下の点について、考察し、今後の性能向上の可能性について検討しておく。

[考察]埋込みパラメータの種類と解析攻撃の可能性

以上の埋込みの枠組みにおいて、変換に使用する係数、量子化幅などが個別の FingerPrinting としての透かし埋込みに対して変更可能である。その場合の直流成分の逆変換が最大になっており、例えば 0.5 という係数があった時、平均の画素値 128 の

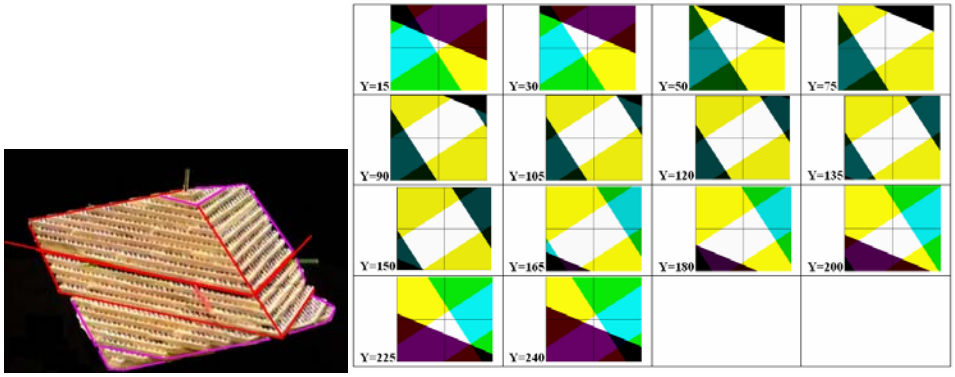



図 8　YIQ 立体：（左）Y が縦軸。（右）Y 値が 15 から 240 までの IQ 平面も断面図.白い部分が IQ の存在範囲.

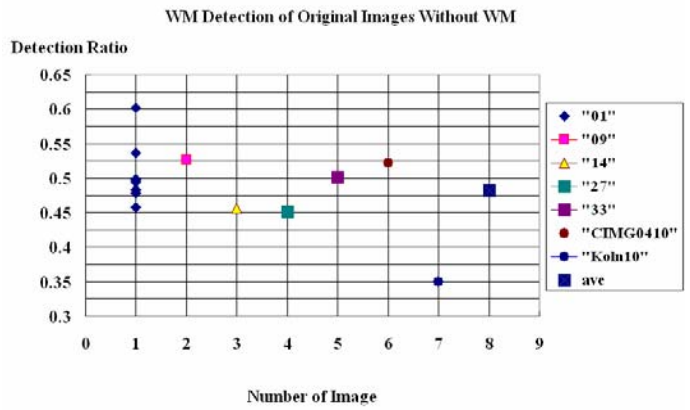



図 9　透かしを埋め込んでいない原画像に対する検出テスト。横軸は画像番号、画像 1 では、量子化ステップサイズ 2,4,8,16,32,64,128,256 を適用、総ブロック数は 2214789.

変換後 DC 値 256（＝128×0.5×4 個）に対し、誤差感度は、1/128 であり、これより小さい係数の変化は、結果に影響がなくなる。従って 0.5 という係数に対して 0.4～0.6 までの許容範囲を設定した場合、0.400,0.404,…0.600 まで約 51 個の種類がある。4 次

Vol.2011-DPS-146 No.29
Vol.2011-CSEC-52 No.29
2011/3/11

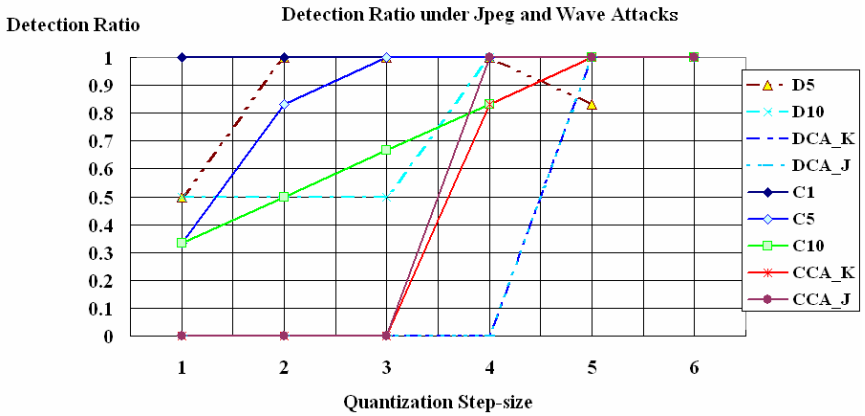



図 10　耐性の実験結果。横軸は、ステップサイズで、4,8,16,32,64,128 に対応。
D：DCT、C：Chirp,C の右の数字は JPEG 圧縮率、CA は他の波形攻撃の場合

の変換では、全部で 16 個の係数があるので、51 の 16 乗の種類になる。これらは当初設定した正規直交変換から逸脱したものであるが、変換は正規性や直交性が必須ではなく、逆変換が存在する正則行列でありさえすれば良い。従って、これらの摂動処理で、ほぼ正則性は保たれると考えられるので、これら全てのパターンは使用可能と予想される。また 8 次の場合は、同じく、誤差感度が約 1/140 になり、約 70 種類となり、70 の 64 乗になる。また、量子化は今回 4 ～64 程度の固定の幅を実験したが、耐性の強いものとして、16 から 64 程度の幅が望ましい。このなかで、任意の数を量子化値として設定可能である。実際は、倍数の関係にある 2 個の量子化値は、片方の検出が包含されるので、識別性が劣る値になる。確実に識別性があるのは素数であるので、この間の素数の個数を求めると、11 個あるので、組合せの数は上記の値の約 10 倍あることになる。

次に、構成を基に攻撃における探索回数を検討する。図 11 に攻撃の観点での本構成の検出器の未知パラメータの配置を示す。画像 G を入力し、真の変換係数を求めることを試みることとする。現在、変換係数に関する手がかりは全くないため、個別の変換係数候補を用いて、検出 ROM と同じ結果になるものを探索していく。この探索回数は、概数で上記考察から、4 次の場合で、51 の 16 乗、8 次の場合で、70 の 64 乗となる。一方、1 回の埋込み処理は、4 次又は 8 次の行列積が 2 回と、逆変換を求める逆行列計算が 1 回となる。検出処理は行列席 1 回である。量子化は演算量としては無視することにする。現在の数値計算の分野から、まず、線形変換、量子化そして再度の線形変換の直列構成に入力と出力を与えて、線形変換係数を求めることは、ニューラルネットワークにおける 3 層構成のパーセプトロンに対応している。中間層の量子化が無い場合は、解を求めることができるが、非線形な量子化が中間層に有る場合は、解が求まらないことが証明されている[27]。更に数値計算を行っても、回数を削減すれば、多くの実験により 100%の解に到達しないことが、経験的にも知られている。

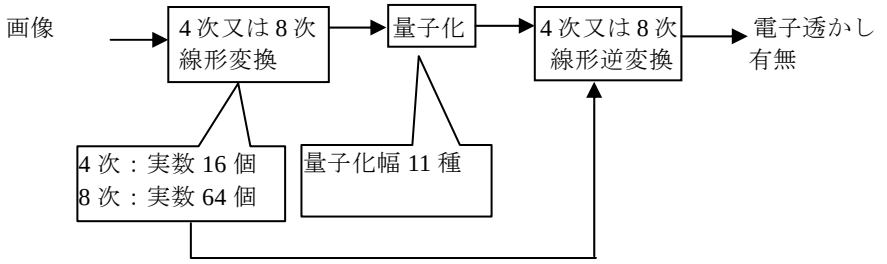



図 11　電子透かし埋込み器のパラメータと自由度

図 11 の構成は、連続関数に対する連立非線形方程式とも異なり、量子化が非連続性を示すので、現状では、全探索するしかないと仮定して、全探索の回数を見積もることにする。表 3 は最近の数値計算分野での行列積と逆行列の演算時間を実測した例である。今回の 4 次又は 8 次は数値計算の分野では、小さすぎる方であるため、大きいサイズの結果から、ほぼ同じ割合で削減した数値を用いた。行列のサイズが 100 以上では、サイズが 2 倍になると、 7 － 8 倍の計算量になっていることが多い。そこで、サイズを半減するたびに、計算量を 1/7 にし、8-16 から下へ向かう小サイズでは、オーバーヘッドが増加するため、それぞれ 1/6,1/5 とした。また、各種高速計算手法が開発されており、計算量は多種あるが、標準的なものを基準に取る観点で、多数有るものは、平均をとってある。高速計算では、10 倍近く早くなるものもあるが、特殊な用途物もあり、高速化は全体の見積もりの外で、付加的に検討するようにした。表 3（d）は文献[28]より引用したが、2000 次において、計算手法により、行列積では、BLAS 法が 14.686（秒）、MKL 法が 1.263 となっていた。また、逆行列では、LAPACK が 22.121、MKL が 1.901 でそれぞれこの平均値を使用している。このデータは現在よりも少し前のプロセッサを用いた計算であるが、その処理速度は、表 3 (c)で 15GFLOPS,表 3 (d)で 21GFLOPS であり、現在商用で多く使用されている、Intel プロセサが 50GFLOPS 程度であることから[29]、これも時期により、補正を加えることとして、除外する。表 2 の 4 次 8 次の最小計算時間に、上記探索回数を乗ずると、表 4 のように、4 次、8

Vol.2011-DPS-146 No.29
Vol.2011-CSEC-52 No.29
2011/3/11

表 3 (a)　M-1800/20(1995 年)16 次以上が実測データ［33］

| 次数 | 行列積 | 連立一次方程式 |
|---|---|---|
| 4 | 1.01000E-05 | 8.13333E-06 |
| 8 | 5.05000E-05 | 4.06667E-05 |
| **16** | **0.000303000** | **0.000244000** |

表 3 (b)　VP2600/10(1995) 16 次以上が実測データ［33］

| 次数 | 行列積 | 連立一次方程式 |
|---|---|---|
| 4 | 3.40000E-06 | 6.73333E-06 |
| 8 | 1.70000E-05 | 3.36667E-05 |
| **16** | **0.000102000** | **0.000202000** |

表 3 (c)　Core2Duo,1.86GHz(2009) 14.88GFLOPS,1280,5120 が実測データ［34］

| 次数 | 逆行列 | 次数 | 逆行列 |
|---|---|---|---|
| 5 | 3.46933E-07 | 320 | 0.040816327 |
| 10 | 2.42853E-06 | 640 | 0.285714286 |
| 20 | 1.69997E-05 | **1280** | **2** |
| 40 | 0.000118998 | 2560 | 15 |
| 80 | 0.000832986 | **5120** | **100** |
| 160 | 0.005830904 | | |

表 3 (d)　Core2Duo,2.66GHz(E6700) 21.28GFLOPS、2000 が実測データ[35]

| 次数 | 行列積 | 連立一次方程式 |
|---|---|---|
| 4 | 3.2259E-07 | 4.86152E-07 |
| 8 | 1.61295E-06 | 2.43076E-06 |
| 16 | 9.6777E-06 | 1.45845E-05 |
| 32 | 6.77439E-05 | 0.000102092 |
| 65 | 0.000474207 | 0.000714643 |
| 125 | 0.00331945 | 0.005002499 |
| 250 | 0.023236152 | 0.035017493 |
| 500 | 0.162653061 | 0.245122449 |
| 1000 | 1.138571429 | 1.715857143 |
| **2000** | **7.97** | **12.011** |

次共に 1 年の秒数 3.2E7 を大幅に上回っており、探索には不可能な時間がかかることが分かる。

表 4　全探索の計算時間見積もり結果（埋込み器の場合）

| 次数 | 計算時間 |
|---|---|
| 4 | (3.2E-7 ×2+4.9E-7)×51^16=1.1E-7 ×1.0E27=1.1 E 20 |
| 8 | (1.6E-6 ×2＋2.4E-6)×70 ^ 64＝5.6E-6×1.0E118=5.6E112 |

## 5. まとめと今後の課題

埋込み検出処理ソフトを難読化し、小サイズブロックにも耐性のある新しい疑似 Chirp 変換を用いた埋込みを行う電子透かし方式を提案し、良好な埋込み特性と耐性を確認した。新たに開発した疑似 Chirp 変換は、複数の周波数成分を有するため、変換後の絶対値は値が大きくなり、その分散も大きくなることがわかった。これにより、画面内の中で、透かしの埋込み可能領域が増加し、従来不可能だった平坦部でさえも埋込むことが可能になった。用いた難読化は計算時間が短い、計算結果が同一である。また、埋込み器の難易度に関しては、現在可能な全探索の攻撃では、探索時間が年単位以上で、十分長い。プログラムサイズは、増大するが、現在の PC 環境では実現可能なサイズである。

今後は、埋込みブロックサイズの拡大と、関数処理の難読化を電子透かし処理から一般的処理関数への拡張と成立条件の検討などを行っていく。

## 参考文献

1) G. Fahmy,1 M. F. Fahmy, and U. S. Mohammed, "Nonblind and Quasiblind Natural Preserve Transform Watermarking", EURASIP Journal on Advances in Signal Processing Volume 2010 (2010), Article ID 452548, 13 pages

2) Mehemed Bashir Aliwa, Tarek El-Ahmady El-Tobely, Mahmood M. Fahmy, Mohamed EL Said Nasr and Mohamed Hashem Abd El-Aziz, "A New Novel Fidelity Digital Watermarking Based on Adaptively Pixel- Most-Significant-Bit-6 in Spatial Domain Gray Scale Images and Robust", American Journal of Applied Sciences (7): 987-1022, 2010

3) Katzenbeisser, S. and F.A.P. Petitcolas, "Information Hiding Techniques for Steganography and Digital Watermarking. 1st Edn.", Artech Print,Canton Street Norwood, MA., 1580530354, pp: 220. 1999

4) Collberg, C. Thomborson, C. Low, D. "Manufacturing cheap, resilient, and stealthy opaque constructs", Proceedings of the 25th ACM SIGPLAN-SIGACT, pp.184 - 196, 1998.

5) Drape, S.,"Intellectual Property Protecting Using Obfuscation", CS-RR-10-02, March 2009. Research sponsored by Siemens AG, Munich.

6) Boaz Barak, Oded Goldreich, Rusell Impagliazzo, Steven Rudich, Amit Sahai, Salil Vadhan and Ke Yang, "On the (Im)possibility of Obfuscating Programs", Advances in Cryptology — CRYPTO 2001, Lecture Notes in Computer Science, 2001, Volume 2139/2001, pp.1-18,
7) http://www.cs.princeton.edu/~boaz/Papers/obf_informal.html
8) 大関和夫，叢力，「計算量的難読化を仮定した，不特定第三者の認証に依存する電子透かし方式 in Japanese 」情報処理学会，研究報告，コンピュータセキュリティ研究会，CSEC-32, pp.61-66,2006 年 3 月.
9) Stanley Chow, Philip Eisen, Harold Johnson and Paul C. Van Oorschot, "White-Box Cryptography and an AES Implementation", Lecture Notes in Computer Science, 2003, Volume 2595/2003, pp.250-270.
10) D. Upham. Steganographic algorithm JSteg. http://zooid.org/ paul/crypto/jsteg.
11) Kodovský, J. and Fridrich, J. "Quantitative Structural Steganalysis of Jsteg", IEEE Transactions on Information Forensics and Security, pp.681-693, Dec. 2010.
12) Kodovský, J. and Fridrich, J., “Quantitative Steganalysis of LSB Embedding in JPEG Domain”, *Proc. ACM Multimedia and Security Workshop*,, pages 187-198, Sept., 2010.
13) Pratt, W., Chen, W., Welch, L. "Slant Transform Image Coding", IEEE Trans. COM-22, pp.1075-1093, Aug.1974.
14) 山根延元，森川良孝,成相剛士,鶴原篤,”斜交軸上の DCT による画像の高能率符号化法",電子情報通信学会論文誌. B-I,J81-B-1(2), pp.110-117, 1998 年 2 月
15) Shilpashree Srinivasamurthy, David Q. Liu, “Survey on Cloud Computing Security”, 2nd IEEE International Conference on Cloud Computing Technology and Science (Cloud Com )cloudcom2010_submission_67.pdf
16) Jian Li Ming Xu Ning Zheng Jian Xu ,"Malware Obfuscation Detection via Maximal Patterns",IEEE Intelligent Information Technology Application, 2009. IITA pp.324 - 328
17) Ceccato, M.; Di Penta, M.; Nagra, J.; Falcarin, P.; Ricca, F.; Torchiano, M.; Tonella, P.; Fondazione Bruno Kessler, IRST, Trento, "The effectiveness of source code obfuscation: An experimental assessment, IEEE 17th International Conference on Program Comprehension, ICPC '09.pp.178 - 187, May, 2009.
18) Ohzeki et al., “Obfuscation of Software for Watermark Detector Using Table Function “, IPSJ SIG Technical Report Vol.2010-CSEC-51 No.1pp.1-6, Dec. 2010.
19) Yarlagadda, R. Hershey, J., “A naturalness-preserving transform for image coding and reconstruction“, IEEE Trans. ASSP Vol.33, pp.1005-12, 1985.
20) Furon, T., “A Constructive and Unifying Framework for Zero-Bit Watermarking”, IEEE Transactions on Information Forensics and Security, Vol.2 pp. 149 – 163, 2007.
21) Kazuo Ohzeki, Cong Li, "Consideration on Variable Embedding Framework for Image Watermark against Collusion Attacks", Wavilla Challenge (WaCha) 2005, Proceedings of the WAVILA Workshop on Watermarking Fundaentals D.WVL.2-1.0.pdf, pp.54-62., June 8-9, 2005.
22) 安達丈晴,長谷川まどか,加藤茂夫,"DCT を利用した静止画像の電子透かし法についての検討",Technical report of IEICE. HCS 99(384), 17-22, 1999 年 10 月
23) Jan Kodovský, Jessica Fridrich, "Calibration Revisited,"Proc. ACM Multimedia and Security Workshop, Princeton, NJ, September 7–8, pp. 63–74, 2009.
24) Jiri Fridrich, "Combining low-frequency and spread spectrum watermarking", Proc. SPIE Int. Symposium on Optical Science, Engineering, and Instrumentation, pp.2-12.July, 1998.
25) Jiri Fridrich, Lt Arnold C. Baldoza and Richard J. Simard, "Robust Digital Watermarking Based on Key-Dependent Basis Functions", Proc. 2nd Information Hiding Workshop, LNCS vol. 1525, Springer-Verlag, New York,pp. 143-157 1998.
26) M. D. Swanson, B. Zhu, and A. H. Tewfik, “Robust data hiding for images”, *Proc. of the IEEE Digital Signal Processing Workshop*, pp. 37–40, Loen, Norway, September 1996.
27) Minsky, M. and S. Papert., Perceptrons; an introduction to computational geometry, MIT Press, 1969（＝中野馨, 阪口豊訳『パーセプトロン』改訂版, パーソナルメディア, 1993）
28) 渡部 善隆: 「数値計算と速度の話～スーパーコンピュータはどれくらい速いか～」 九州大学大型計算機センター広報, Vol.28, No.3 (1995), pp.207-231.
29) 山本 有作、「マルチコア・メニーコアプロセッサ向けの固有値計算アルゴリズム」 *筑波*大学 計算科学研究センター(CCS)主催、分野横断型研究会「アルゴリズムによる計算科学の融合と発展、2009 年 4 月.
30) 楠原 文雄**,**「数値計算ライブラリ」http://www.rcs.arch.t.u-tokyo.ac.jp/kusuhara/fswiki/wiki.cgi の数値計算ライブラリ 2009 年 3 月 21 日
31) インテル® Core™2 Duo プロセッサー http://www.intel.com/jp/support/processors/sb/cs-023143.htm