社団法人　情報処理学会　研究報告
IPSJ SIG Technical Reports

2005－ＣＳＥＣ－30（59）
2005／7／22

非対称電子透かしにおける効率的な多ビット埋め込みについての考察

岡田満雄†　　菊池浩明†

† 東海大学大学院工学研究科　〒259-1292 神奈川県平塚市北金目 1117
E-mail: †{mitsuookada,kikn}@ep.u-tokai.ac.jp

あらまし　本稿では，[1] を拡張し，実用的な多ビット埋め込み法についての非対称電子透かしを考察する．従来の多くの電子透かしは，埋め込み処理と抽出処理が対称であり，透かし埋め込みに用いられた秘密情報をそのまま，抽出に用いていた．従って，抽出時に秘密情報が，検証者に露呈してしまい，透かし情報が除去される危険性がある．この問題に対応すべく Furukawa らが公開鍵暗号方式である Paillier 暗号を用い，非対称な透かし検証プロトコルを提案している [5]．しかしこれを多ビット電子透かしにナイーブに適応すると，復号化処理時間のオーバーヘッドが問題となり，実用的ではない．そこで，多ビット埋め込みへの拡張をしたときの実用的な埋め込み方法について考察する．
キーワード　パッチワーク法電子透かし，Paillier 暗号方式．

On Multiple-bit Embedding in Asymmetric Watermark

Mitsuo OKADA† and Hiroaki KIKUCHI†

† Graduate School of Engineering, Tokai University　1117 Kitakaname, Hiratsuka, Kanagawa, 259-1292 Japan
E-mail: †{mitsuookada,kikn}@ep.u-tokai.ac.jp

Abstract　A new method of efficient multi-bit asymmetric watermark algorithm is proposed in this paper. Our methodology applies to the achievement of efficient multi-bit embedding for secure digital watermark detection that exposes no secret information to a watermark verifier. Furukawa proposed a secure watermark detection scheme [5] in 2004 using the Paillier encryption, but this method is not suitable for multi-bit embedding due to its heavy overhead in extraction processing time. We proposed a consideration for an efficient multi-bit watermark embedding method.
Key words　Patchwork Watermark, Pailler encryption.

1. まえがき

近年，インターネットの爆発的な普及に伴い映像や画像あるいは音楽などのコンテンツのデジタル化が著しい．その理由のひとつとして，デジタルコンテンツは複製，編集，配布，管理などが容易であることがあげられる．しかし，その反面それらの特性を逆手にとった違法行為が後を絶たず，急増し続けている．特に著作権を無視したデジタルコンテンツの不正複製，利用や改竄などが後を絶たず，これらは即急に対処すべき問題である．

そこで，それらの問題に対処する情報秘匿技術の一手法に電子透かし技術がある．電子透かしとは，コンテンツ自身に情報を秘匿する技術であり，著作者情報や秘密情報を埋め込むことで著作権の主張や原本性の主張を可能にし，不正コピーなどの抑止などにも使われる．不正に画像データを CD-R に複製し販売していた男性が，画像に埋め込まれていた透かし情報を根拠に逮捕されたりしている [10]．

しかし多くの電子透かしの欠点はその対称性である．埋め込みと抽出に同じ秘密情報となる鍵が使われていることが多く，透かし埋め込み時の秘密情報が検証者に露呈されてしまう危険性が伴う．

Furukawa らは，この問題に対応すべく公開鍵暗号方式である Paillier 暗号方式と統計的電子透かしの一手法であるパッチワーク法による電子透かし [5] を併用し，非対称的な透かし抽出アルゴリズムを提案した．復号処理時間のオーバーヘッドを削減するために，[1] では処理の軽い El Gamal 暗号で代用する方法の提案と評価が報告されている．

一般に透かしとして埋め込む情報は，著作名や日付，著作権や利用条件などであり，これらには数ビットの情報量がある．ところが，これまでに提案されてきた [3] [5] などは基本的な 1 ビットの埋め込み方法のみが研究されており，多ビットの埋め込みに拡張するのは自明ではない．例えば，[5] を k ビットの埋

め込みに対して k 回ナイーブに埋め込み処理を繰り返すと，埋め込み位置を秘匿した暗号文のサイズが k 倍になり効率が悪い．k 回繰り返し画像を変更すれば，同じ画像を多重に操作してしまう可能性も考慮しなくてはならない．

そこで，本稿では，上記の問題に対して，[5] では画素の位置を表現するのに平文空間中の $-1, +1, 0$ の 3 値しか用いていなかった点に注目し，ここに多ビットを埋め込む方法を試みる．

本稿では，まず 2 章でパッチワーク法と，[5] と [1] によるアルゴリズムを示した後，3 章で提案方式を述べる．さらに，4 章では [1] の実装データを元に，新たな多ビット透かし検出方法を評価する．

2. 準　　備

2.1 パッチワーク法

パッチワーク法電子透かしとは，1995 年に MIT の Bender らが提案した方法であり，広範囲にわたり情報を秘匿し，改竄などのアタックに対して耐性を持たせている方法である．電子透かし抽出時には統計的な偏りを利用している．多くの自然画像は冗長性を持っており，数画素程度の輝度レベルの増減は画像全体ではほとんど影響がでない．画像 からランダムに 2 画素 $(a,\ b)$ を選定し，$(a_i - b_i)$ を n 組生成し，その和を S_n とする．すなわち，

$$S_n = \sum_{i=1}^{n}(a_i - b_i)$$

ここで S_n の期待値 (平均値)$\bar{S_n} = S_n/n$ は，n の増大に伴い 0 に近づく．すなわち，

$$\lim_{n\to\infty} \bar{S_n} \to 0 \qquad (1)$$

となる．

画像の統計的な偏りの特性を利用して，1bit の透かし情報 ω を埋め込む方法を示す．

STEP1　擬似乱数生成器にシードを与え，操作する画素の位置となる擬似乱数系列を生成する．そして，原画像 I から二つの画素値 (a_i, b_i) を n 組選定する．

STEP2　選定された画素に以下の処理を施し統計量を操作する．このとき δ は透かしの強度である．

$$a_i' = a_i + \delta,\ b_i' = b_i - \delta. \qquad (2)$$

STEP3　STEP1 と STEP2 を全ての組に対して行い，原画像 I から埋め込み画像 I' を生成する．このときの $\bar{S_n}$,

$$\bar{S_n} = \frac{1}{n}\sum_{i=1}^{n}(a_i + \delta) - (b_i - \delta) = \frac{1}{n}\sum_{i=1}^{n}(a_i - b_i) + 2\delta = 2\delta$$

は 2δ に限りなく近づく．

lena の原画像 (256×256 画素，256 階調のグレイスケール) と埋め込み画像 (δ=20，n=10000) の統計量の変化を図 1 に示す．本図が示すように，I の期待値は $\bar{S_n} = 0.0121$ であり，(1) を満たしている．また I' は $2\delta = 40$ 右にシフトしている．すなわち，この δ が大きければ検出時の確実性は向上するが，画像に多くの劣化を伴い，逆に δ が小さすぎれば検出時の False Positive や False Negative の確率が上がる．

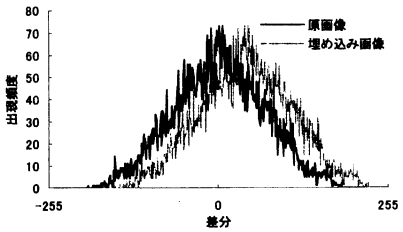

図 1　差分 $(a_i - b_i)$ の分布図 $(\delta = 20, n = 10000)$

抽出時には検査対象画像から同じシードによる擬似乱数系列を復元し，それを元に a_i' と b_i' を選定し，$\bar{S_n}'$ を計算する．次に統計値の偏りを抽出し以下のように判定することにより，透かしビット ω が 0 か 1 かを求める．

$$\omega = \begin{cases} 0 & \bar{S_n}' < \tau, \\ 1 & \bar{S_n}' \geqq \tau. \end{cases} \qquad (3)$$

また，このとき false positive = false negative にするために，閾値 $\tau = \delta$ にする．従って，図 1 の透かしを検出するためには，τ=20 にする．本実験において，lena の期待値は $\bar{S_n}' = 40.0158$, $\bar{S_n}' \geqq \tau = \delta = 20$ となり，(3) の ω=1 の条件を満たす．

2.2 暗号プロトコル

本節では，El Gamal 暗号と Paillier 暗号のアルゴリズムを示す．

El Gamal 暗号方式は，離散対数問題に基づく初めての公開鍵暗号方式であり，パッチワーク法と併用して提案方式に用いる．大きな素数 p を生成し，乗法群 Z_q^* での原始元 g を求める．次に，秘密鍵 $x \in Z_q$ を定め，$y = g^x \bmod p$ を算出する．ここで，秘密鍵は x，公開鍵は y, g, p である．平文 m の暗号文 $E(m) = (c, d)$ は $c = g^m y^r \bmod p$ と $d = g^r$ により計算される．そして，復号は $g^m = D(c, d) = c/d^x \bmod p$ により計算される．

Paillier 暗号方式は [7] により提案された．本方式では，大きな素数 p, と q を生成し，$\gcd(L(g^\lambda \bmod N^2), N) = 1$ を満たす $g \in Z_{N^2}$ を生成する．このとき，$N = pq, \lambda = \mathrm{lcm}(p-1, q-1)$ である．公開鍵は (g, N) で秘密鍵は p, q である．そして乱数 $r \in Z_N$ を生成する．E を暗号化関数，D を復号化関数としたとき，暗号文 e は以下のように求められる．

$$e = E(m) = g^m r^N \bmod N^2. \qquad (4)$$

復号して平文 m' を求めるには以下の処理を施す．

$$m' = D(e) = \frac{L(e^\lambda \bmod N^2)}{L(g^\lambda \bmod N^2)} \bmod N, \qquad (5)$$

ここで $L(t) = \frac{t-1}{N}$ である．

2.3 非対称電子透かし [3]

Minematsu は 2000 年に非対称電子透かしの検出プロトコルを提案した [3]. パッチワーク法による電子透かしに準同型性を満たす公開鍵暗号方式を組み合わせることで, 秘密情報を透かし検証者に知られることなく, 検出を可能にするプロトコルを提案した. この方式では透かし検証者は埋め込み画像から信頼できる第三者 (Key Authority) の助けの元で, 透かしを検証する方式である. 公開鍵暗号には Okamoto-Uchiyama らによる暗号方式が用いられている [4].

2.4 検証可能非対称電子透かし [5]

Furukawa らは Paillier 暗号方式を採用することにより, 安全な透かし検証プロトコルを提案した. 本方式は [3] に近いが, 透かし検証者であるボブが検出結果の有効性を秘密情報の露呈なしに行うことができる. 本プロトコルでは, 操作画素 (A, B) の位置を含む Paillier 暗号文を暗号化されたまま検証することにより透かし抽出処理が行われる. Paillier 暗号方式の性質により, 秘密情報は g のべき指数として暗号化される.

埋め込み者は閾値 τ, 画像の総画素数 l, 操作画素 A, $B \subset \{1, \ldots, l\}$ を定める. そして Paillier 暗号に必要な秘密鍵と公開鍵を生成し, 暗号文 $(e_1, \ldots, e_l)$ を以下の様に計算する

$$e_i = \begin{cases} E[1] & \text{if } i \in A, \\ E[-1] & \text{if } i \in B, \\ E[0] & \text{otherwise.} \end{cases}$$

透かし抽出処理において, $I' = (z_1, \ldots, z_l)$ を有する検証者は, $e = \prod_{i=1}^{l} e_i^{z_i}$ を計算し, e を信頼出来る第三者に送信する. 透かし検出処理では, 検証者が ω を以下のように計算する.

$$\omega = \begin{cases} 0 & \text{if } D(e) < \tau, \\ 1 & \text{if } D(e) \geqq \tau. \end{cases}$$

詳細については [5] を参照されたい.

2.5 El Gamal 暗号による実装 [1]

[5] の欠点は Paillier 暗号の復号処理時間のオーバーヘッドであるが, [1] では El Gamal 暗号を用いることを提案している. なぜならば, パッチワーク法では 0 か 2δ のどちらに近いかだけを検査する必要性しかなく, 期待値が g^0 から $g^{2n\delta}$ の何れかに同定されるかは, 処理の軽い El Gamal 暗号でブルートフォースによる検査法でも十分であるからである. さらに, [1] では実装に基づいて埋め込み処理のパフォーマンスを評価している.

2.5.1 モデル

本モデルでは 3 人の登場人物を仮定する. アリスはコンテンツの著作者であり, 透かしを埋め込み処理をする. ボブは透かし検証者で, ケビンは信頼できる第三者で秘密情報の暗号と復号に必要な El Gamal 暗号の秘密鍵 sk と公開鍵 pk を生成し, 保持する. $I = (x_1, \ldots, x_l)$ が原画像とし, $I' = (z_1, \ldots, z_l)$ を埋め込み画像とする. l は I の総画素数である. 以上のモデルを図 2 に示す.

2.5.2 プロトコルの詳細

鍵生成: ケビンは秘密鍵 x のときの El Gamal 暗号の公開鍵 $y = g^x \bmod p$ を生成する.

STEP1 埋め込み法: アリスは δ と $\{1, 2, \ldots, l\}$ からランダムに操作画素 A と B を $A \cap B = \phi$ と $|A| = |B| = n$ の条件を満たすように選ぶ. A と B に従い, アリスは I に透かしを埋め込んだ $I' = (z_1, \ldots, z_l)$ を生成する. ただし,

$$z_i = \begin{cases} x_i + \delta & \text{if } i \in A, \\ x_i - \delta & \text{if } i \in B, \\ x_i & \text{otherwise.} \end{cases}$$

そしてアリスは A と B の暗号文 $e = (c_1, \ldots, c_l,\ d_1, \ldots, d_l)$ を生成する. ここで, $c_i = g^{m_i} y^{r_i}$, $d_i = g^{r_i} \bmod p$,

$$m_i = \begin{cases} 1 & \text{if } i \in A, \\ -1 & \text{if } i \in B, \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

であり, r_i は Z_q の乱数である. 最後にアリスは $I' = (z_1, \ldots, z_l)$ と e をボブに送る.

STEP2 抽出法: ボブは暗号文 e と I' から $e' = (C, D)$ を以下のように変換する.

$$C = c_1^{z_1} \cdots c_l^{z_l} = \prod_{i=1}^{l} g^{m_i z_i} y^{r_i z_i} = g^{\sum^{l} m_i z_i} y^{\sum^{l} r_i z_i} = g^{S_n} y^R,$$

$$D = d_1^{z_1} \cdots d_l^{z_l} = \prod_{i=1}^{l} g^{r_i z_i} = g^R,$$

ここで $R = \sum_{i=1}^{l} r_i z_i \bmod q$ とする. そして, ケビンに e' を送る.

STEP3 ケビンは秘密鍵 x で e' から $M = C/D^x = g^{S_n}$ を計算して復号し, ボブに送り返す.

STEP4 ボブは $M = g^s$ の条件を満たす, べき指数 s を総当りで $s = 1, 2, \ldots, q$ について検査して, s を同定する. これによりボブは秘密情報 ω を以下のように抽出する.

$$\omega = \begin{cases} 0 & \text{if } s < \tau, \\ 1 & \text{if } s \geqq \tau, \end{cases}$$

このとき τ は閾値である.

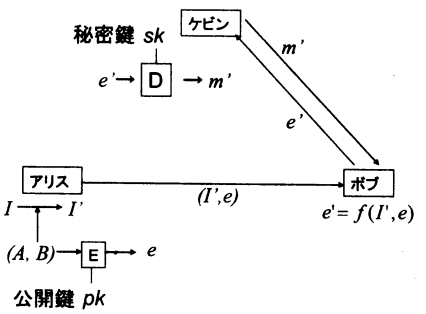

図 2 非対称電子透かし法のモデル

3. 提案方式

3.1 概　　要

従来方式に基づいて，k ビットの透かし情報 ω を埋め込む方法を考える．この ω をどのように符号化して暗号文にするかが，効率に関係してくる．

3.2 ナイーブな方法

k ビット毎に埋め込み位置 $(A_1, B_1), \ldots, (A_k, B_k)$ を独立に生成し，従来方式を繰り返す．すなわち，埋め込み画像は，

$$z_i = \begin{cases} x_i + \delta & \text{if } i \in A_1 \cup \cdots \cup A_k, \\ x_i - \delta & \text{if } i \in B_1 \cup \cdots \cup B_k, \\ x_i & \text{otherwise,} \end{cases} \tag{7}$$

と定められる z_i について $I' = (z_i, \ldots, z_l)$ とする．そして，$A_1, \ldots, A_k, B_1, \ldots, B_k$ の暗号文は，A_j, B_j に (6) 式を適応して定められる e_j によって構成される $e_1, \ldots, e_k$ となる．このとき j は埋め込みビット数のインデックスである．従って，埋め込みビット数 k に比例して暗号文が大きくなり，効率が悪い．

例えば，$l = 65536$ の画像について $k = 3$ ビット埋め込んだ場合，1024 ビットの暗号文に対して El Gamal 暗号からは 2 個の暗号文が生成され，それらの総画素数分の暗号文が必要にるため，結局 e の情報量は $|e|$=1024 · 2lk=50Mbyte になる．さらに多ビット埋め込みの場合，検査画素の k が増加するため，El Gamal 暗号でのブルートフォースによる検査では同定処理に負荷がかかる．そのため，方式をナイーブに用いた埋め込み法は実用的ではない．

3.3 ビット毎に平文を変える方法

(7) 式では A, B に対して，$+1$ と -1 を埋め込んでいたが，これを $+c, -c$ とする．従って，$A_1, \ldots, A_k, B_1, \ldots, B_k$ についての m_i は

$$m_i = \mu_{i,1} + \mu_{i,2}\phi^1 + \cdots + \mu_{i,k}\phi^{k-1} \bmod q$$

となる ϕ 進数に符号化される．ただし，

$$\mu_{i,j} = \begin{cases} +1 & \text{if } i \in A_j, \\ -1 & \text{if } i \in B_j, \\ 0 & \text{otherwise,} \end{cases} \tag{8}$$

である．

この平文に El Gamal 暗号を適応して，$E(m_i) = (c_i, d_i) = (g^{m_i} \cdot y^{r_i}, g^{r_i} \bmod p)$ と定める．この時，A_j, B_j の暗号文のサイズは，従来方式と変わらないが，復号時に誤りが起きないように十分大きな ϕ をどのように定めるかが問題である．一例として，サンプル画像 lena に $n = 2053, \delta = 3$ で $\omega_1 = 01$, $\omega_2 = 10$, と $\omega_3 = 11$ を埋め込んだ時の差分の総和 S_n の確率分布を図 3 に示す．図が示すように分布の重複が多く，大きな抽出誤りが生じることが明らかである．

3.4 生成元を変える方法 (提案方式)

一般に群 Z_p において，位数を q にする生成元は複数存在する．これを，$g_1, \ldots, g_k$ とする．ここで，生成される乗法群は $< g_1 >=< g_2 >= \cdots =< g_k >$ であることに注意する．この

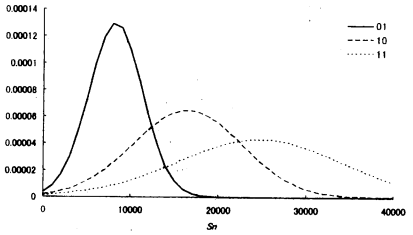

図 3　S_n の確率分布

時，$A_1, \ldots, A_k, B_1, \ldots, B_k$ の平文を $i = 1, \ldots, l$ について，

$$m_i = g_1^{\mu_{i,1}} \cdot g_2^{\mu_{i,2}} \cdots \quad g_k^{\mu_{i,k}} \bmod p$$

と定める．ただし，

$$\mu_{i,j} = \begin{cases} +1 & \text{if } i \in A_j, \\ -1 & \text{if } i \in B_j, \\ 0 & \text{otherwise,} \end{cases} \tag{9}$$

とする．この平文に対して，El Gamal 暗号を適応する時は，$E(m_i) = (c_i, d_i) = (m_i \cdot y^{r_i}, g^{r_i} \bmod p)$ とし，Paillier 暗号の時は，$E(m_i) = m_i r^N \bmod N^2$ とする．(g^{m_i} でないことに注意せよ)．

透かしの検出方法は公開鍵アルゴリズムにより異なる．El Gamal 暗号の場合，復号によって m_i が得られるため，各ビットの値に展開するためには，

$$D(e') = g_1^{S_1} \cdot g_2^{S_2} \cdots g_k^{S_k} \bmod p$$

を満たす $(S_1, \ldots, S_k)$ を総当りで求める必要がある．一方，Paillier 暗号の場合は，

$$D(e') = S_1 + \phi_2 S_2 + \cdots + \phi_k S_k \bmod N$$

となる $(S_1, \ldots, S_k)$ を求めればよい．ここで，ϕ_j は，$g_j = g^{\phi_j} \bmod N$ を満たす定数であり，公開しておいてよい．方式 2 との違いは，ϕ のべき指数で表現するか，ビット毎に独立の ϕ_j を用いるかと言ってよい．

3.5 画像劣化低減の為の拡張方式

埋め込みによる画像劣化を最小限に抑えるため，異なるビット間で以下の処理を施す．まず異なるビット間での A_i と B_j の重複を避け，抽出時の誤り率を低下させる．ここで i と j はそれぞれ，異なるビットのインデックスである．さらに A_i と A_j や B_i と B_j の重複の場合には重複したビットの δ の操作を無効にすることにより画像の劣化を最小限にする．なぜならば，異なるビット間で同じ方向に生じた変化は，1 回分の画素値の変更量が両ビットに効くからである．この関係を図 4 に示す．例えば，ある画素が $i \in A_1 \cup A_2$ に属していたとする．この時，$c_i = g_1 g_2 \cdot y^{r_i}$ であるため，$z_i = x_i + \delta$ とすれば，$c_i^{z_i}$ により g_1 と g_2 の両者に累乗される．この効果を見るために，サンプル画像について多ビットを埋め込んだ時の PSNR を図 5 に示す．以上の方法をまとめた比較を表 1 に示す．

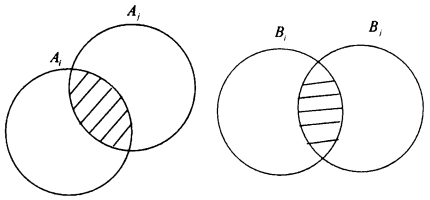

図 4　拡張方式の透かし位置割り当て

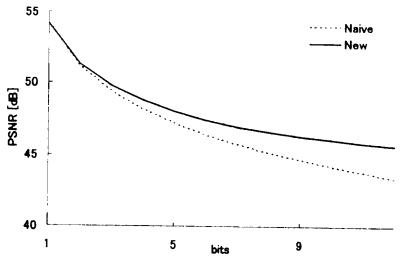

図 5　多ビット埋め込み画像の PSNR の変化

表 1　方式比較

	方式 1	提案方式 (2 と 3)
画像 I'	$z_1,\ldots,z_l$	$z_1,\ldots,z_l$
暗号文 e のサイズ	$1024\cdot 2\cdot lk$	$1024\cdot 2\cdot l$
透かし抽出法	STEP2 と 3 を k 回繰り返す	STEP2 と 3 を 1 度実行し，2^k 通りから透かしを検出

4. 評　　価

4.1 安 全 性

パッチワーク法による電子透かしは安全性に関して以下の特性がある．最初に広範囲性であるが，操作画素 A と B が $\{1,\ldots,l\}$ の広範囲にわたり拡散している．広範囲性による埋め込みの効果を図 7 の A,B の分布から見ることが出来る．以上により I' の埋め込み鍵を知らずに透かし除去をすることは困難である．次に原本非参照性を満たしている．これは抽出時に原本を必要とせず，透かし情報抽出時に原画像の漏洩などのリスクが無く，差分攻撃などを避ける．最後に，不可視性を確認するため，図 6 に 256×256，255 階調，$n=2053,\delta=3$ で，8 ビット埋め込まれた画像を示す．またこのときの SN 比は 46.5767[dB] である．

次に El Gamal 暗号とコンテンツ改竄に対しての安全性について述べる．離散対数問題に基づく難しさにより，I' と暗号文 e からボブは埋め込み時の秘密情報 A と B による情報は何も得られない．ボブから受信した暗号文 $e=(C,D)$ から，ケビンは 暗号文の生成に使われた I' や埋め込み鍵を算出することは出来ない．

4.2 最適なパラメータ

本節では最適なパラメータ δ の選定方法について述べる．埋め込み時の δ と操作ビット数 n を与えた時，確率論に基づき，95%の確からしさで透かしが判定できるための δ の最小値は以下のように求められる．

図 6　8bit 埋め込み画像

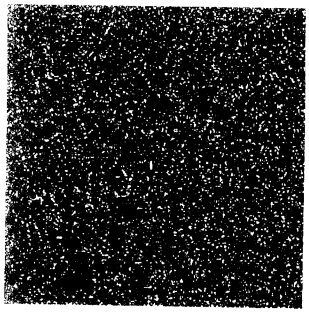

図 7　操作ビット A と B の分布図

$$\mu=\bar{S}_n,\sigma=\frac{\sigma'}{\sqrt{n}},\delta=2\sigma.$$

ただし、ここで，σ' は n 個のサンプルの (a_i-b_i) の標準偏差，σ は平均値 $\bar{S}_i$ の標準偏差である．本実験では総和の平均値 μ、標準偏差 σ'、母集団の標準偏差 σ を求め，算出した δ を表 2 と図 8 に示す．このときの最適な埋め込み画像は 2σ 右にシフトするため，δ とは区別することが出来る．

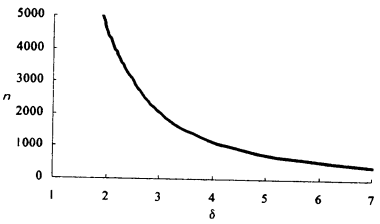

図 8　δ の最適値

δ の定義のために，操作画素の数 n と画質の関係を考察する．画質は以下の SN 比により定義される．

$$\mathrm{PSNR}=10\cdot\log_{10}\frac{255^2}{\mathrm{MSE}^2}=10\cdot\log_{10}\frac{255\cdot 255}{\sum(x_i-x_i')^2},$$

表 2　n についての最適なパラメータ δ

n	μ	σ'	σ	δ
4613	0.8847	67.4449	0.4769	2
2053	1.9206	67.9670	1.5000	3
1165	-0.4335	68.2865	2.0007	4
757	-1.3805	68.8136	2.5011	5
539	-2.0260	69.7601	3.0048	6

このとき MSE は I と I' の間に発生する平均 2 準誤差である．256×256 の lena 画像を表 2 のパラメータを用い実装した時の n についての PSNR を図 9 に示す．このとき $n = 2053$ と $n = 4613$ の間には差が生じていないことを示している．

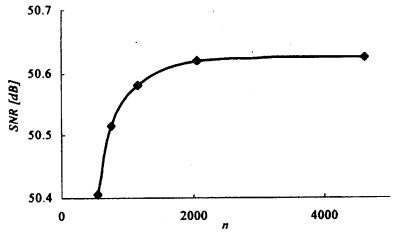

図 9　操作画素数 n と SN 比の関係

4.3　実装に基づく性能評価

提案方式の性能を評価するため，グレイスケール画像に対する透かし埋め込み抽出システムを C 言語で，暗号計算を Java で実装した．表 3 に実装環境を示す．

表 3　実 装 環 境

詳細	仕様
CPU	Xeon 2.3GHz
OS	Redhat 9.0, Linux 2.4.20
メモリ	1GB
暗号アルゴリズム	1024-bit El Gamal, 1024-bit Paillier
プログラミング言語	J2SDK 1.4.2, gcc (GCC) 3.3.3

本実装において lena をホスト画像 I として，異なるサイズ l =64×64, 128×128, 256×256 で埋め込み，暗号化，復号化，そして抽出を実装した．

1. 透かし埋め込み法

C 言語による画像サイズ l についての，1 ビットの埋め込み処理時間を図 10 に示す．処理時間は画像サイズ l に比例している．

2. El Gamal 暗号と Paillier 暗号による鍵生成

El Gamal 暗号による 1024-bit の暗号化と復号処理時間は，0.104 [s] と 0.077 [s] であり，Paillier 暗号は 33.03[s] と 21.27[s] である．これらのデータから各画像サイズに対しての鍵生成プロセス時間の見積もりを表 4 に示す．これにより，鍵生成は画像サイズ l に比例することが示されている．

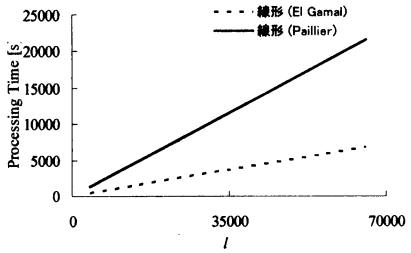

図 10　埋め込み処理時間

表 4　鍵　生　成

l	64×64	128×128	256×256
El Gamal [s]	425.9840	1703.936	6815.744
Paillier [s]	1353.03168	5412.12672	21648.50688

4.4　Furukawa 法と [1] の検出処理時間の比較

両者の本質的な違いは非対称電子透かしと併用する暗号アルゴリズムである．1 ビット埋め込み時の両者の透かし検証に要する総処理時間の比較を図 11 に示す．このときのデータは，4.2 節に示されているように全ての n に対して異なるシードで 10 個のデータを採取した平均値である．El Gamal 暗号の処理時間は n と比例して増大するのに対して，Paillier 暗号は一回分の復号で透かしを抽出することが出来る．図 11 で用いた値を表 5 に示す．1 ビットでは処理時間が図 11 に示したように n に比例すると仮定して，Paillier が El Gamal を下回るのは，n^*=7403 の時である．これを元に多ビットの場合の検出処理時間を見積もると，El Gamal 暗号の場合，検出に要する処理時間は 1 ビット時の処理時間 v に対して v^k に増加していく．Paillier 暗号の場合，図 11 で示した v に総当りで k^2 個の組み合わせの中から検出結果を同定する処理時間が追加される．

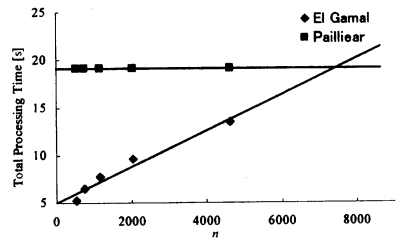

図 11　[1] と [5] の処理時間の比較

表 5　透かし検証の処理時間

n	539	757	1165	2053	4613
[1] (El Gamal)	5.279	6.475	7.697	9.590	13.47
[5] (Paillier)	19.11	19.11	19.11	19.11	19.11

5. おわりに

本稿では，非対称電子透かしにおける，多ビットを埋め込む問題について検討した．べき指数表現や独立した生成元を生成することにより，埋め込み位置の暗号文のサイズがビット数に依存しない方式を提案した．今後の課題として，誤り訂正符号などの補助的な技術を組み合わせることにより，透かし検証率が向上することが期待でき，電子透かしの信頼性が向上すると考えられる．さらに高解像度の 32 ビットなどの画像に対応させることにより統計的に有利な結果が得られると考えられる．

文　献

[1] M. Okada and H. Kikuchi, Secure Detection of Asymmetric Digital Watermark, DICOMO 2005. (発表予定)

[2] Bender, Technique for Data Hiding, SPIE, vol.2020, pp. 2420-2440, 1995.

[3] K. Minematsu, On a Secure Digital Watermark Detection Protocol Using Patchwork Watermarking, ISITA 2000, pp. 673-676, 2000.

[4] T. Okamoto and S. Uchiyama, A New Public-key Cryptosystem as Secure as Factoring, Enrocrypt' 98, LNCS 1403, pp. 308-318, 1998.

[5] J. Furukawa, Secure Detection of Watermarks, IEICE Trans., vol. E87-A, no. 1, pp. 212-220, 2004.

[6] El Gamal, T., A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms, IEEE Trans, IT-31, 4, pp. 649-472, 1985.

[7] P. Paillier, Public-key Cryptosystems based on Composite Degree Residuosity Classes, Eurocrypt'99, LNCS 1525, pp. 223-238, 1999.

[8] Petitcolas, Anderson, and Kuhn. Attacks on Copyright Marking Systems, Information Hiding, Second International Workshop IH'98, LNCS 1525, Springer-Verlag, pp. 219-239, 1998.

[9] Petitcolas, Watermarking Schemes Evaluation, IEEE Signal Processing, vol.17, no. 5, pp. 58-64, 2000.

[10] ACCS ニュース, (http://www2.accsjp.or.jp/news/news050608.html), 2005 年 6 月参照.