

反復型可逆的情報ハイディングを利用した大容量二次元コード

新見道治 九州工業大学情報工学部准教授

1 はじめに

インターネットに代表される情報通信技術の発展によって、大容量の情報を簡単に送受信することが可能になった。その一方で、インターネット上での情報の送受信は第3者に容易に盗聴され、重要な情報の漏えいといった深刻な問題も生じている。この問題の対策として、暗号化した情報を送受信するという方法が挙げられる。しかし、暗号化を施しただけでは、暗号データの送受信自体は第3者に感知されてしまい、暗号を解析される恐れがある。そこで、情報を別の情報に埋め込んで秘匿する技術である情報ハイディングが考案された。例えば、送信するダミー情報の中に重要な情報を隠せば、第3者に情報の存在を気付かせることなく情報の送受信が可能である。

情報ハイディングにおいて、特にデジタル画像に情報を埋め込む場合、埋め込む前の画像をカバー画像、埋め込み後の画像をステゴ画像と呼ぶ。カバー画像に情報を埋め込むと画素値が変化するため、ステゴ画像には少なからず劣化が生じる。このような画像への情報ハイディングのうち、ステゴ画像から埋め込んだ情報を抽出した後にカバー画像を完全に復元することができる技術を可逆的情報ハイディングと呼ぶ。

可逆的情報ハイディングではカバー画像が完全に復元できるので、繰り返し埋め込み処理を実行し、埋め込み量を増やすことができる。もともと、文献[1]の手法をベースに繰り返し埋め込み手法の検討を行っていたが、従来法より格段に埋め込み性能が優れる二値画像に対する可逆的情報ハイディングが実現できた。よって、本報告では、印刷画像や文字画像といった、2値で表わされることの多い画像に対する可逆的情報ハイディングの手法について述べ、さらに繰り返し可逆的情報ハイディングを適用することで、埋め込み容量を増加させる。このことを本研究では反復型可逆的情報ハイディングと呼ぶ。最終的には、QRコードのような二次元コードに対して、反復型可逆的情報ハイディングを適用し、QRコードが持つ情報量を増加させ、大容量二次元コードの実現を目指す。

2 2値画像に対する可逆的情報ハイディング

2.1 埋め込み

2.1.1 埋め込み用ブロック候補

まず、カバー画像を $m \times m$ 画素のブロックに分割する。次に、各ブロックの複雑さ α を計算し、あらかじめ決定しておいた閾値以上の複雑さを持つブロックを埋め込み用ブロック候補とする。この閾値処理で抽出された埋め込み用ブロック候補はノイズ状のブロックである。つまり、このブロックは画素の変化が発生しても視覚的に影響が出にくい領域であり、埋め込みに適している。

2.1.2 埋め込み用画素候補

埋め込み用ブロック候補において、ブロック中の半分の画素を埋め込み用画素候補とする。埋め込み用画素候補は、ブロック左上の画素を含む市松模様状に配置された画素である。

2.1.3 埋め込み用画素の決定

埋め込み用画素候補のうち、特定の近傍のビットパターンを持つものを埋め込み用画素とする。近傍ビットパターンが同じ画素は、同じ画素値を持つ可能性が高い。よって、それらの画素を並べて生成したビット列は圧縮が可能となり、秘密情報を埋め込むスペースを確保することができる。

近傍ビットパターンは、2近傍、3近傍、4近傍あわせて、合計64パターン存在する。本研究では、それらのパターンに一意な番号を付けている。図1に近傍ビットパターンとそのパターン番号の対応関係をいくつか示す。例えば、4近傍全てが黒画素の近傍ビットパターンの番号は15番である。

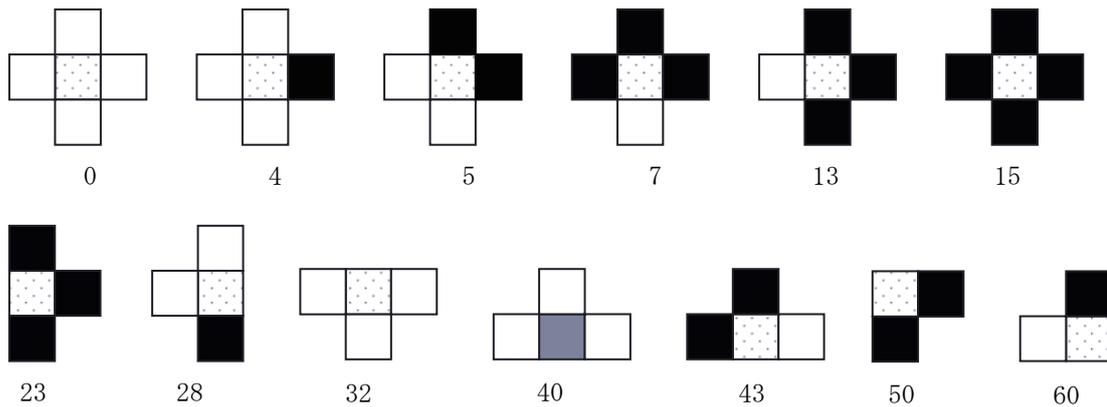


図 1：近傍ビットパターンとパターン番号

2.1.4 埋め込み用ブロックの決定

ある埋め込み用ブロック候補に対して埋め込みを行うと、そのブロック中の埋め込み用画素は変化し、埋め込み用ブロック候補の複雑さも変化する。閾値処理により埋め込みに適したノイズ状のブロックを抽出し、埋め込み用ブロック候補とした。同じように埋め込み用のブロックの複雑さは、埋め込みによって画素の変化が生じたとしても、必ず閾値以上にならない。よって、埋め込み用ブロック候補において、埋め込み用画素をどのように変化させても、その複雑さが常に閾値以上となるブロックを埋め込み用ブロックとする。

2.1.5 圧縮情報の生成と埋め込み

画像中の全ての埋め込み用画素をラスタスキャン順に並べてビット列を生成する。このビット列を算術符号化により圧縮したものを圧縮情報とする。圧縮情報がビット列より小さいとき、圧縮情報と秘密情報からなるビット列と埋め込み用ブロック内の埋め込み用画素の値を書き換えることで、埋め込みを実現する。ここまでの手順を踏まえ、図 3.4 に埋め込み方法のアルゴリズムを示す

2.2 秘密情報の抽出・カバー画像の復元

埋め込んだ秘密情報をステゴ画像から抽出する方法と、カバー画像を復元する方法について述べる。

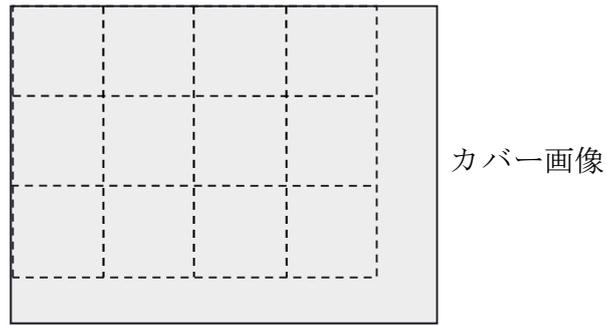
秘密情報の抽出は埋め込み時と同じ流れで行う。まずステゴ画像を $m \times m$ 画素のブロックに分割する。次に、各ブロックの複雑さを計算し、閾値以上の複雑さを持つブロックが埋め込み用ブロック候補、つまり秘密情報が埋め込まれている可能性があるブロックである。そして、埋め込み時と同様の条件を用いて、埋め込み用ブロック候補から埋め込み用ブロックを特定する。その埋め込み用ブロックの埋め込み用画素の値を、ラスタスキャン順に並べビット列を生成することで、埋め込まれた秘密情報と圧縮情報を抽出できる。

抽出した圧縮情報を復号化して得られるビット列と埋め込み用画素を置き換えれば、カバー画像を完全に復元することができる。

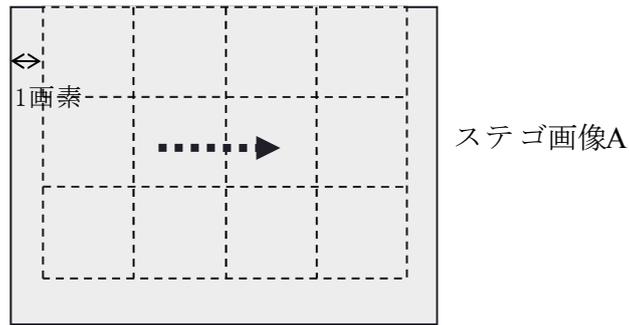
2.3 反復型埋め込み

本研究では、1枚の2値画像に対して、4回の埋め込みを反復して行う。このとき、埋め込みを行うたびに、ブロックを1回目の埋め込みの時に対して右方向に1画素シフト、下方向に1画素シフト、右、下方向に1画素ずつシフトさせる。すると、前回の埋め込み時に利用しなかった画素を埋め込み用画素候補として用いるため、2回目以降の埋め込みにおいて埋め込み効率が低下することを防ぐことができる。

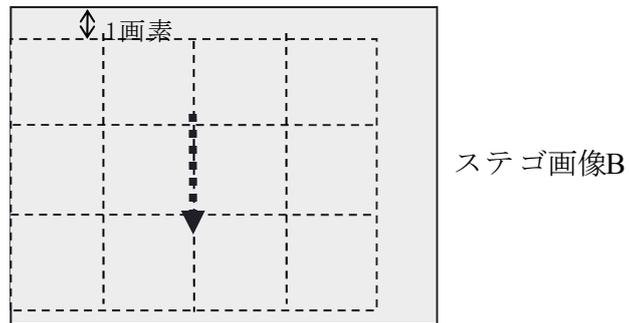
図 2 に反復型埋め込みのイメージ図を示す。ここで、灰色の矩形領域が2値画像、破線の矩形領域が $m \times m$ 画素のブロックの集合、破線の矢印が1回目の埋め込み時に対するブロックのシフト方向を示す。



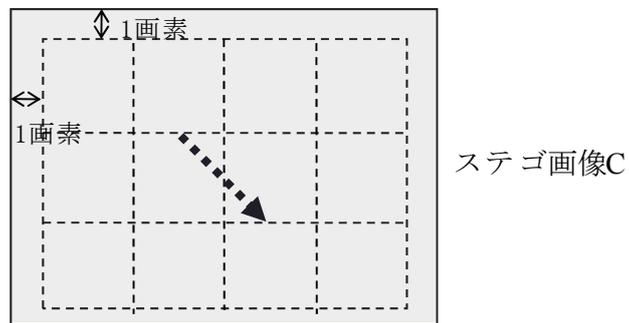
↓ 埋め込み1回目



↓ 埋め込み2回目



↓ 埋め込み3回目



↓

ステゴ画像D

図 2 : 反復型埋め込み

3 実験と考察

3.1 実験条件

3.1.1 カバー画像

実験には、サイズ 816x612 の 8 枚の画像を用いる。これらの画像は 4 種類のグレースケール画像 (flowers, computers, students, characters) に対し、組織的ディザ法、大津の方法の 2 種類の 2 値化方式を施して生成した (以下、組織的ディザ法と大津の方法によって 2 値化された画像を、それぞれディザ画像、2 レベル画像と呼ぶ)。ブロックサイズは 8x8 画素とする。

3.1.2 実験内容

用意した 8 枚の 2 値画像に対して埋め込みを行う。埋め込みは 1 枚の画像に対し、4 回ずつ反復して実行する。実験は 2 種類行い、閾値もしくは近傍ビットパターンの番号と埋め込み量の関係を明らかにする。

3.2 実験 1

複雑さの閾値を 0/112 から 80/112 まで変化させ、それぞれの埋め込み回数での閾値 (Threshold) と埋め込み量 (Payload), および埋め込みによって変化した画素数 (Flipped pixels) の関係を求める。なお、埋め込みには 0 番の近傍ビットパターンを用いる。

3.2.1 実験結果

ディザ画像の場合、埋め込みが可能な閾値の範囲が広がった。また、3 回目の埋め込み量は 2 回目の埋め込み量を下回っていた。4 回目の埋め込み量は、他の試行時に比べて大きな値を示していることが多かった。埋め込みによって変化した画素数は、埋め込み量に比例する傾向にあった。

2 レベル画像の場合、埋め込みが可能な閾値の範囲が狭い傾向にあった。また、3 回目の埋め込み量は 2 回目の埋め込み量を上回っていた。4 回目の埋め込み量はほぼ 0 であった。埋め込みによって変化した画素数は 4 回目が高く、3 回目が最も低かった。

3.2.2 考察

ディザ画像に対する埋め込みは、画像の複雑さの平均程度の閾値までならば可能となっている。これは、ディザ画像が複雑なブロックを多く持つ、つまり画像中に埋め込み用ブロックが多く存在するためである。2 レベル画像では、閾値が小さい場合にしか埋め込みが出来ていない。これは、2 レベル画像が複雑さの小さなブロックを多く持つので、ブロックの複雑さが閾値を越える場合が少ないためである。以上より、提案手法はディザ画像のように、ブロックの複雑さの平均値が大きくかつ同じような複雑さのパターンを多く持つ画像に対して有効であるといえる。一方、2 レベル画像のように、ブロックの複雑さの平均値が小さい画像に対しては有効ではない。

試行回数によって埋め込み量が変わるのは、埋め込みごとにブロックをシフトさせ、利用する埋め込み用画素候補を変えるためである。0 番の近傍ビットパターンを用いた本実験では、4 回目の埋め込み量が大きくなる傾向があった。ここで、他の近傍ビットパターンを用いた場合の埋め込み回数と埋め込み量の関係を調べてみた。具体的には flowers のディザ画像に対して 7 番と 13 番の近傍ビットパターンを用い、同様の実験を行った。実験の結果、使用した近傍ビットパターンの番号が 7 番の場合は 3 回目の埋め込み量が、13 番の場合は 2 回目の埋め込み量が大きくなっていた。よって、埋め込み回数と埋め込み量の関係は、使用する近傍ビットパターンによって異なることが分かる。

3.3 実験 2

閾値が 20/112, 30/112, 40/112 の場合において、近傍ビットパターンの番号 (Neighborhood bit patterns) と複雑さの閾値 (Threshold) を変化させ、近傍ビットパターンの番号と埋め込み量の総和 (Sum of payload) の関係を求める。

3.3.1 実験結果

表 1 と表 2 に、それぞれの画像と閾値に対して、埋め込み量の総和が最も大きい場合の近傍ビットパターンの番号、さらにその条件下での埋め込み量の総和と埋め込みによって変化した画素数をまとめた。例えば students (ディザ画像) に閾値 40/112 で埋め込みを行った場合、埋め込み量の総和が最も大きくなるのは、4 番の近傍ビットパターンを利用した場合であった。

近傍ビットパターンの選択が、埋め込み量に影響しているのは明らかであった。例えばディザ画像においては 4, 7, 13 番など、2 レベル画像においては 0, 5, 15 番などの近傍ビットパターンを利用した場合に埋め込み量が増えている。さらに表 4.1 と表 4.2 より、同じ 2 値化方式で生成された 2 値画像でも、画像もし

くは閾値によって、最適な近傍ビットパターンは異なることがわかる。

表1：ディザ画像に対する最適な近傍ビットパターンの番号

画像名	閾値	近傍ビット パターンの番号	埋め込み量の総和	変化した画素数
	20/112	7	26573	16486
flowers	30/112	7	25114	15664
	40/112	13	21318	23431
	20/112	13	18102	19962
computers	30/112	13	21645	24226
	40/112	4	17071	19816
	20/112	13	21706	23820
students	30/112	13	19257	19257
	40/112	4	16500	19397
	20/112	13	29571	30146
characters	30/112	13	26174	25831
	40/112	4	21077	24447

表2：2レベル画像に対する最適な近傍ビットパターンの番号

画像名	閾値	近傍ビット パターンの番号	埋め込み量の総和	変化した画素数
	20/112	15	10672	12709
flowers	30/112	15	2627	3871
	40/112	15	1513	261
	20/112	0	3490	4027
computers	30/112	0	427	787
	40/112	5	2	422
	20/112	15	3030	3093
students	30/112	0	214	378
	40/112	(埋め込み不可)		
	20/112	0	10730	10086
characters	30/112	0	1945	2106
	40/112	0	81	197

3.3.2 考察

ディザ画像は複雑さが大きく、白画素と黒画素が入り混じっている。よって4, 7, 13番など、近傍に白と黒の画素を両方とも持つ近傍ビットパターンが埋め込みに適している。2レベル画像は複雑さが小さく、同じ画素で構成される領域が多い。よって、近傍が全て白画素である0番、もしくは全て黒画素である15番の近傍ビットパターンが埋め込みに適している。以上より、画像の複雑さに基づいて、埋め込みに適切な近傍

ビットパターンを決定する必要があるといえる。

3.4 追加実験

埋め込みを行ったとき、埋め込みに必要な画素数と埋め込み後に反転した画素数を調べることで、埋め込み後に画像の劣化が生じるかを検証する。画像のハッシュ値の埋め込みを仮定し、提案手法を利用して 8 枚の 2 値画像に対して 128bit の情報を埋め込む。このとき、埋め込みに必要な画素数と埋め込みによって変化した画素数を調べる。なお本実験では、埋め込みには 0 番の近傍ビットパターンを用いる。閾値は、埋め込み量が 128bit 以上の場合で最も小さいときの値を設定する。表 3 にディザ画像、2 レベル画像に対する 128bit 情報の埋め込み結果を示す。

表 3 : 2 値画像への 128bit 情報の埋め込み

	ディザ画像		2 レベル画像	
	埋め込みに必要な画素数	埋め込みによって変化した画素数	埋め込みに必要な画素数	埋め込みによって変化した画素数
flowers	1017	491	1232	625
computers	538	265	854	386
students	533	262	588	290
characters	314	146	587	291

埋め込みによって反転した画素は最大でも画像全体の 0.0013%程度であり、ごくわずかであるといえる。また、カバー画像とステゴ画像を比較しても、画像劣化を視認することはできなかった。よって、画像上の視覚的に影響が出にくい領域に効果的に埋め込みが行われているといえる。

3.5 QR コードをカバーとする反復型埋め込み

サイズ 57x57 と 114x114 の二つの QR コードをカバー画像として (図 3 参照), 反復型可逆的情報ハイディングを適用した。

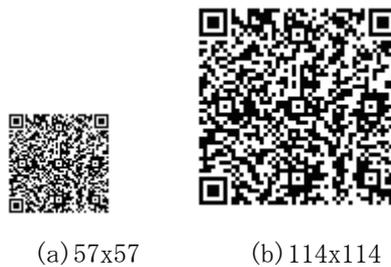
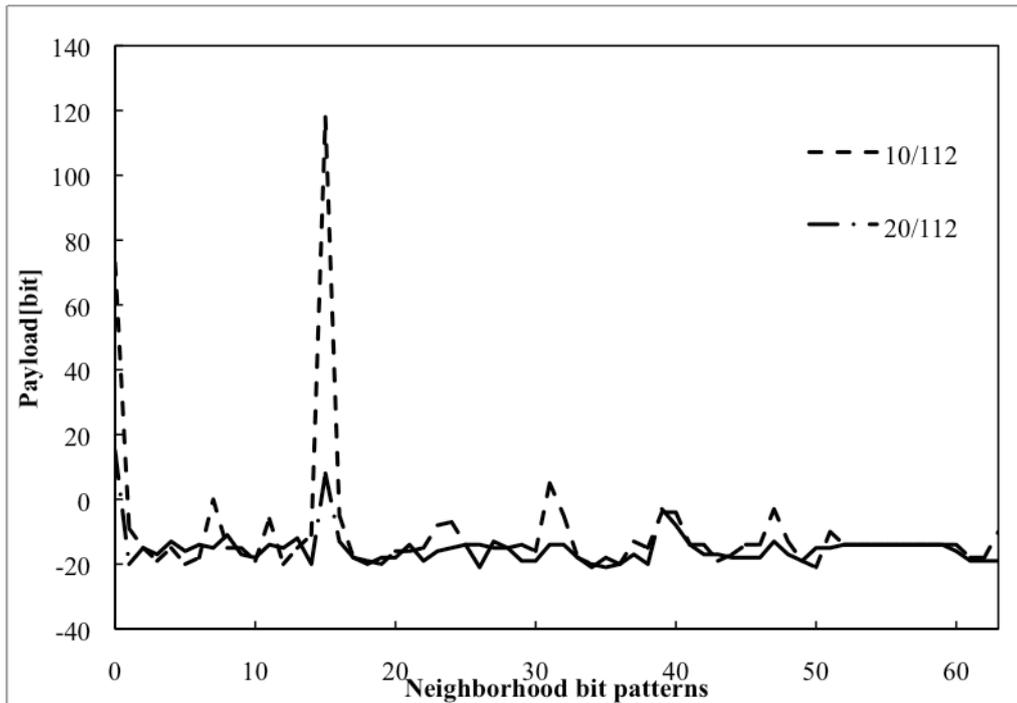
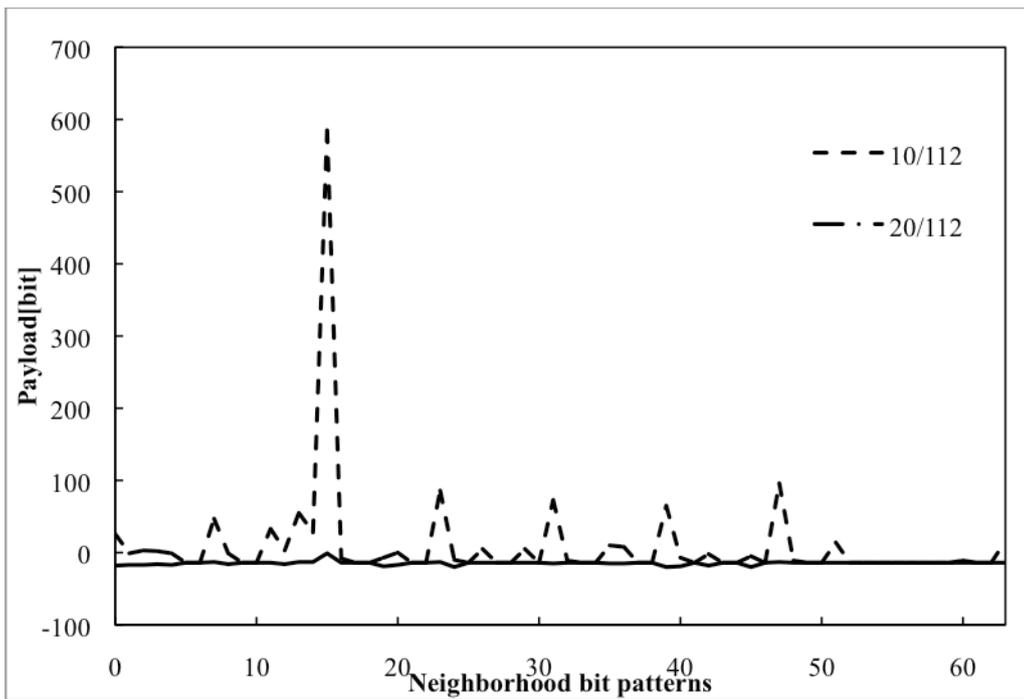


図 3 : QR コード (カバー画像)

使用する近傍ビットパターンを変化させたときの埋め込み容量を調べた。実験結果を図 4 に示す。



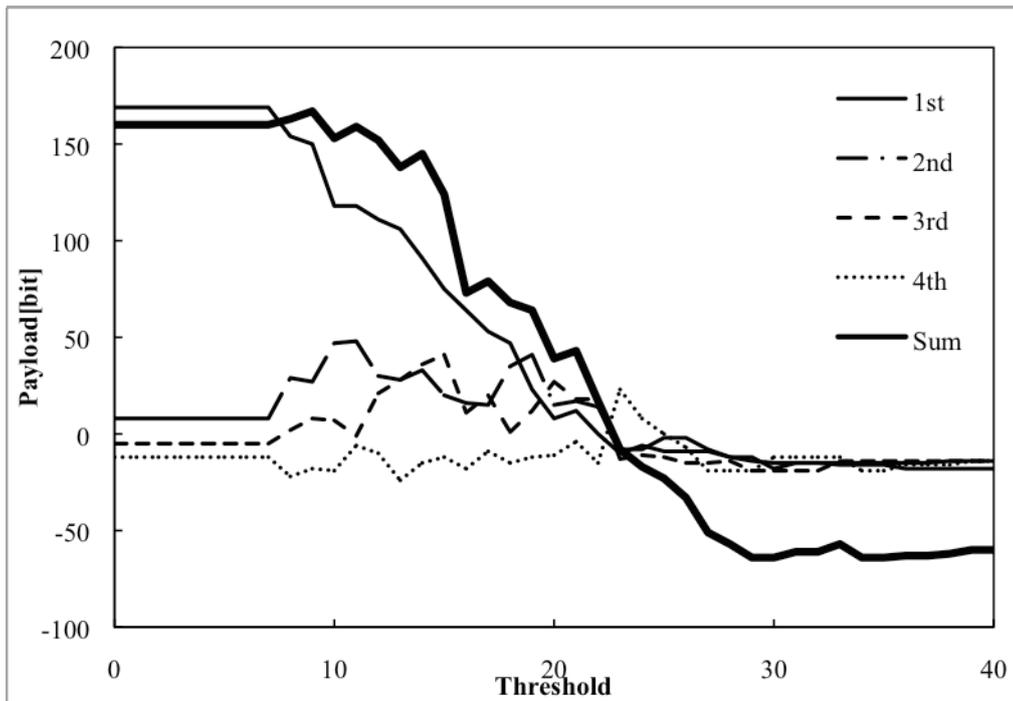
(a) 57x57 に対する結果



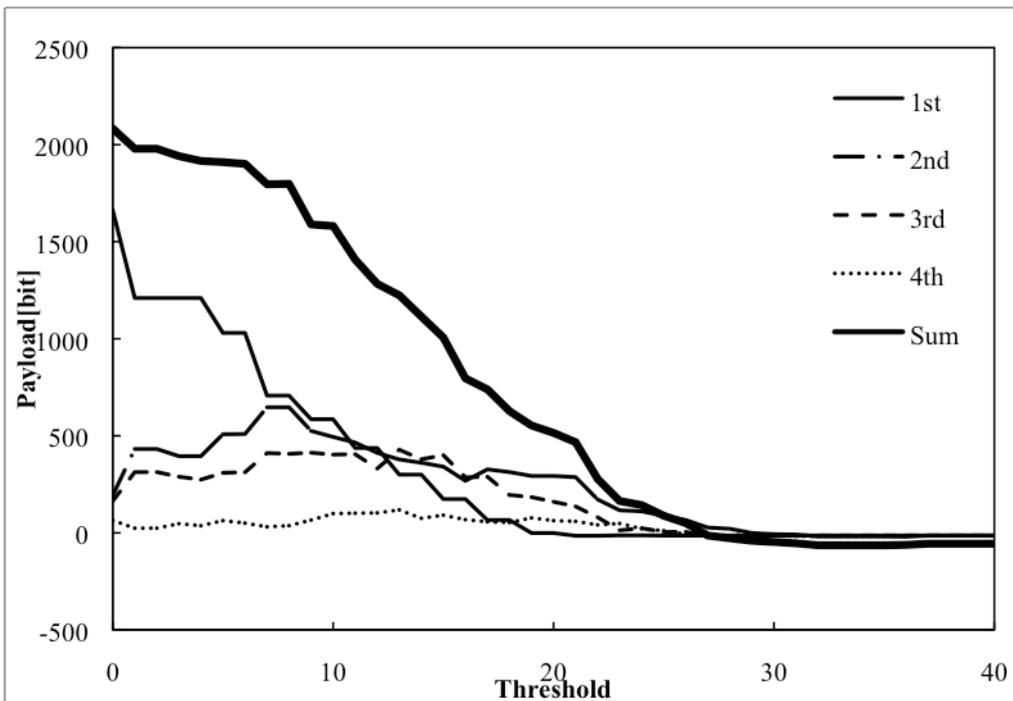
(b) 114x114 に対する結果

図 4 : 近傍ビットパターンを変化させたときの埋め込み容量の変化

次に、閾値を変化させた時の埋め込み容量の変化を図 5 に示す。



(a) 57x57 に対する結果



(b) 114x114 に対する結果

図 5 : 閾値を変化させたときの埋め込み容量の変化

114x114 の QR コードに対しては、4 回繰り返して埋め込むと 2000 ビット程度埋め込めることが分かる。これは、1 回埋め込む量の約 1.5 ~ 2 倍であり、反復型可逆的情報ハイディングが有効に機能していることが分かる。

4 まとめ

提案手法では、画像をブロック分割し、閾値処理により画像上のノイズ状のブロック、すなわち内部の画素が変化しても視覚的に認識しにくい領域を抽出する。そしてノイズ状のブロックから、近傍ビットパターンを利用することにより圧縮しやすい画素を選択する。また埋め込みを反復して4回行い、埋め込みを行うたびにブロックをシフトさせる。以上の手順を経ることで、画質の劣化を抑えながらも、より大容量の秘密情報を埋め込むことを実現した。ただし使用する近傍ビットパターンによって、埋め込み回数と埋め込み量の関係や反復埋め込みでの埋め込み量の総和は異なる。ゆえに効率的な埋め込みを行うには、画像の複雑さに応じて埋め込みに使用する近傍ビットパターンを選択する必要がある。

また、QRコードに対して反復型可逆的情報ハイディングを適用してみた。その結果、1回しか適用しないときよりも埋め込み量が増え、約1.5～2倍程度の埋め込み量を実現できた。その量は約2000ビットであり、QRコードに付加情報を持たせる新たな応用分野が期待できる。

今後の課題として、埋め込みの効率を上げることが挙げられる。そのためには、埋め込みに用いる近傍ビットパターンをさらに有効に活用することが重要である。例えば、埋め込みに複数の近傍ビットパターンを混用する、画像ごとに異なる最適な近傍ビットパターンを自動的に求めるアルゴリズムを開発することなどが考えられる。

【参考文献】

[1]

Reversible Information hiding using complexity measure of binary images, Michiharu Niimi, Tetsuya Nakamura, Hideki Noda, Proceedings of 2006 International Workshop on Advanced Image Technology, pp.209-214,2006

〈発表資料〉

題名	掲載誌・学会名等	発表年月
Reversible Information hiding for binary images based on selecting compressive pixels on noisy blocks	Processing of 2009 International Workshop on Advanced Image Technolo	2009年1月
2値画像に対する可逆的情報ハイディング	2009年電子情報通信学会総合大会, pp.S21—S22	2009年3月