

JPEG 符号化列へのバイナリデータの埋込み法

小林 弘幸[†] 野口 祥宏^{††} 貴家 仁志^{†††}

A Method of Embedding Binary Data into JPEG Bitstreams.

Hiroyuki KOBAYASHI[†], Yoshihiro NOGUCHI^{††}, and Hitoshi KIYA^{†††}

あらまし JPEG による非可逆符号化において、画像にバイナリデータを埋め込む方法を提案する。提案する方法は、情報を量子化後の AC 係数に埋め込むものである。その際、情報を高域の AC 係数へ埋め込むものとし、その AC 係数に対応する量子化テーブルの値を小さく設定することにより、視覚的にも画質劣化が小さく抑えられる。また、提案法は、復号化の際に埋め込んだ情報を完全に復元することができ、情報を取り出した後の画像に対する影響もほとんどない。更に情報を取り出さない場合や取り出せない環境においても、従来の JPEG 復号器を用いてその符号化列は復号可能であり、その際の画質劣化も小さく抑えることが考慮されている。

キーワード JPEG, 電子透かし, バイナリデータ

1. ま え が き

近年、デジタル画像を取り扱う機会が増加している。またその際、画像に付随する付加的な情報を画像と同時に取り扱いたい場合が多い。例えば、画像を説明する文字や音声情報、画像の符号化において必要となる付加的な情報である。

しかしながら、静止画像の国際規格である JPEG (Joint Photographic Coding Experts Group)[1] では、文字や音声などの情報は取り扱うことは想定されていない。このような JPEG が直接扱わない情報を JPEG の符号化列中に埋め込むことができれば、JPEG のビットストリームとして画像とともに音声や文字の情報をユーザに受け渡すことができる。

このような背景から、本論文では画像以外の付加情報等を画像データの中に埋め込むことを想定し、JPEG 符号化列へのバイナリデータの埋込み法を考察する。その結果、視覚的に目立たない形態で文字情報などのバイナリデータを画像と一緒に符号化することが可能

となる。更に本方法は、非可逆符号化の使用にもかかわらず、復号側において埋め込まれたバイナリデータを完全に取り出すことができる特徴をもつ。

本論文における情報埋込み法は、以下の条件が考慮されている [2]。

(a) 非可逆符号化である JPEG 符号化列に情報を埋め込んだ際にも、埋込み情報が完全に取り出せること。

(b) 埋込み情報が取り出せない環境においても、従来の JPEG 復号化器を用いて復号することが可能であること。

(c) 情報を埋め込むことによる画像の劣化が小さいこと。

(a) の条件は、付加情報としてバイナリデータ等を取り扱うために必要となる。これは、ほとんどの場合においてバイナリデータは 1 bit の欠損も許されないためである。

本論文と類似する研究として、電子透かしという技術がある [3] ~ [6]。これは、主にデジタル画像に対する著作権の問題を解決する方法として、画像に個人的な署名を埋め込むものである。提案法と署名埋込みとは、主に次の 2 点が異なると考えている。

- 署名埋込みは第三者により再圧縮等による署名削除のための攻撃が想定される。そのため、これらの攻撃に対する耐性を考慮する必要がある。

- 多くの署名埋込みは非可逆な画像圧縮を攻撃と

[†] 東京都立工業高等専門学校電気工学科, 東京都 Department of Electronics, Tokyo Metropolitan College of Technology, Tokyo, 140-0011 Japan

^{††} 旭化成工業株式会社 LSI・情報技術研究所, 厚木市 LSI laboratories, Asahi Chemical Industry Co., Ltd., Atsugishi, 243-0205 Japan

^{†††} 東京都立大学大学院工学研究科電気工学専攻, 八王子市 Faculty of Engineering, Tokyo Metropolitan University, Hachioji-shi, 192-0397 Japan

考えている．一方，提案法は情報埋込み者のみが画像圧縮をすることを前提としたものであり，そのような非可逆圧縮のもとでもバイナリデータが完全に取出せることを目的としている．

しかしながら，電子透かし法の中には，非可逆な画像圧縮のもとでも，提案法と同様にバイナリデータを扱うことができる電子透かし法も存在する．その中で先の (a) ~ (c) の条件を満足する方法に文献 [5], [6] がある．本論文では，この従来法と比較して，多くの場合において提案法の画質劣化が小さくなることが確認できた．

本論文では，2. において，まず JPEG における画像圧縮の手順を簡単に説明し，提案するバイナリデータの埋込みに必要な条件をまとめる．次に 3. では，提案する JPEG 符号化列への情報埋込み法を説明する．4. では，従来法の説明を行い，本方法との相違点の検討を行う．最後に 5. では実際に情報埋込みを行い，従来法との比較を通し提案法の有効性を確認する．

2. 問題設定

ここでは，提案法で前提とする JPEG 符号化手順と，考慮される条件を示す．

2.1 JPEG 符号化の手順 [1]

まず今回使用する JPEG 符号化の手順を簡単に説明する．以下では議論を簡潔にするためにシーケンシャル符号化を例にするが，プログレッシブ符号化においても提案法は適用可能である．

図 1 は JPEG の符号化手順の概略を示したものである．処理すべき画像は 8×8 のブロックに分割され，それぞれのブロックにおいて，DCT，量子化テーブルを用いた量子化，ハフマン符号化が行われる．画像のスペクトルが低域に片寄っているため，高域の DCT 係数は量子化によって零値になることが多い．このような背景から直流以外の係数（AC 係数と呼ばれる）は，図 2 に示される順番でハフマン符号化される．このジグザグスキャンによる AC 係数の最後の零値の集合は符号化されず，ブロックの終了を示す End of Block（以下，EOB と略す）という符号に置き換えられる．

2.2 情報埋込み法の条件

提案法は，以下の条件が考慮されている．

(a) 非可逆圧縮による埋込み情報の保持

本論文では，バイナリデータの埋込みを考慮しているため，1 bit の情報欠落も許されない．非可逆な

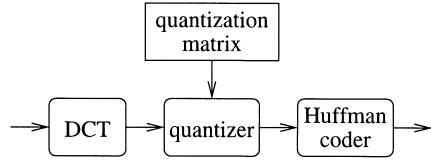


図 1 JPEG 符号化の手順
Fig. 1 JPEG encoder.

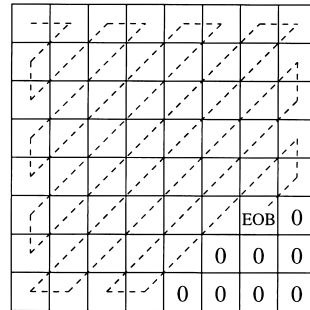


図 2 AC 係数のジグザグスキャン
Fig. 2 Zigzag scan of AC coefficients.

JPEG 符号化においても，復号時に埋込み情報が完全に取出せる必要がある．

(b) 従来のアルゴリズムによる復号

埋め込まれた情報を取り出せない，または取り出す必要がない復号器も想定する必要がある．したがって，情報が埋め込まれた画像を，情報を埋め込まない画像の場合と同様に標準の JPEG アルゴリズムで復号可能でなければならない．

(c) 情報付加による画像劣化

画像に情報を埋め込むことにより，画像は劣化する．情報埋込みの際には，この画像劣化を小さく抑える必要がある．

以上の 3 点が同時に満たされる情報付加の方法を本論文では考察する．

3. 提案する方法

本論文では，前述の条件を考慮した情報付加の方法を提案する．提案法では， 8×8 の各ブロックに最大 1 bit の情報を埋め込むものとする．したがって，例えば 704×480 のグレイスケール画像の場合， $5280(\text{bit}) = 660(\text{byte})$ の情報を最大埋め込むことが可能となる．以降の議論では，ジグザグスキャン順に並べた AC 係数を $AC(k)$ とおく．ただし， k は $1 \sim 63$ である．

3.1 情報追加の方法

提案法では各ブロックにおいて p 番目の AC 係数 $AC(p)$ に注目し、その量子化後の係数値を 1 bit の情報 (1 or 0) で置き換える。ここで p は定数とし、すべてのブロックで同じ場所の AC 係数に対して置換えを行うものとする。その結果、復号側では逆量子化前にその AC 係数を調べることで、埋め込まれた情報を知ることができる。JPEG では、量子化後の処理は可逆であるので、情報が埋め込まれた AC 係数の場所が特定できれば、埋め込まれた情報を復号の際に完全に復元することができる (条件 (a) を満足)。

また、この方法は単に AC 係数を変更しただけであり、情報を埋め込んだ符号化列も通常の JPEG 復号器で処理することが可能である (条件 (b) を満足)。

3.2 $AC(p)$ の選択

$AC(p)$ の選択を考慮するにあたり、前述した条件 (c) で示した情報付加による画像劣化について検討する。この情報付加による画像劣化の原因は三つに大別される。第 1 は付加情報を画像に加える際に本来の画像情報の一部を欠落させることによるもの、第 2 は本来の画像にはない情報を加えることによるものであり、第 3 は付加情報の単純な追加は画像全体のデータ量を増加させることによるものである。すなわち、ある一定のファイルサイズを維持するためには、埋め込まない場合に比べ高い圧縮率で圧縮する必要が生じる。以上のような画質劣化を極力抑えるように情報は埋め込まれる必要がある。

以上のことから、情報を埋め込む場所を示す p は、高域の AC 係数を選択することが望ましい。これは、以下の二つの理由による。

- 高域の係数は量子化後に零値となることが多く、係数置換えによる画像劣化を低く抑えられる
- 情報が埋め込まれた画像を通常の JPEG 復号器で復号した際に、低域の係数に埋め込んだ場合に比べ、高域の係数への埋込みは視覚的に目立ちにくい

前者は、画像劣化の第 1 の原因を回避するためのものである。もし、情報を埋め込む前の画像において、AC 係数 $AC(p)$ が零値でなかった場合には、係数の置換えによる情報欠落が生じてしまう。逆に、すべてのブロックにおいて $AC(p)$ が零値となるような p を選択できれば、復号側において埋め込んだ情報を取り除くことにより、係数置換えによる画質劣化を完全に回避できる。

後者は、画像劣化の第 2 の原因を回避するためのも

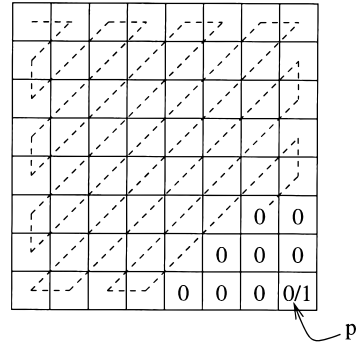


図 3 AC 係数への情報埋込み
Fig. 3 Inserting data into AC coefficients.

のである。通常の JPEG 復号器で復号する場合には、埋め込まれた情報は画像に対するノイズとなってしまう。このノイズの視覚への影響を小さくするために、高域の係数を選択することが望ましい。

また、 p の決定は画像劣化の第 3 の原因である画像符号化列のファイルサイズにも関係がある。図 3 は最高域の係数に情報を埋め込む場合の例である ($p = 63$)。ここで、1 という情報が埋め込まれた場合、埋め込まなかったときに比べ、途中の零値のランレングス符号化分と最後の 1 の係数分だけ符号語が増えることとなる。この増加符号語が符号化列のファイルサイズの増加につながる。もし、ファイルサイズの上限に制限がある場合には、より粗い量子化を行う必要が生じる。

3.3 量子化テーブルの変更

JPEG 符号化器の場合、一般に符号化の際に使用した量子化テーブルをそのままハフマン符号化し、復号側においてその量子化テーブルを利用する。しかし、提案法では符号化の際に使用した量子化テーブルとは異なる量子化テーブルをハフマン符号化する。その理由を以下に示す。

第 1 の理由は、既存の JPEG 復号器による復号画像の劣化を抑えるためである。通常高域の AC 係数における量子化テーブルの値は非常に大きく (表 1)、量子化後に入れた値が 1 という小さい値であっても、逆量子化により非常に大きい値に変換され、復号画像では非常に大きな劣化となってしまう。これが画像劣化の第 2 の原因である。この問題を回避するために、情報を埋め込まれた場所 (p) に相当する量子化テーブルの値を、量子化後に 1 に置き換え、逆量子化の際にそれを利用する (表 2)。

第 2 の理由は、情報を埋め込んだ AC 係数の場所の

表 1 量子化テーブルの例
Table 1 Example of quantizing table.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

表 2 置き換え後の量子化テーブルの例 ($p = 63$)
Table 2 Arranged quantizing table.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	①

特定のためである。復号側では、符号化列に含まれる係数の置き換えを行った量子化テーブルより、符号化列で選択した p の場所を知ることが可能となる。ただし、この検出のためには p よりも高域の量子化テーブルに 1 という値が存在しないことが必要となる。

以上をまとめると次のようになる。

- 符号化側 (図 4(a) 参照)

(a) 全ブロックの EOB の場所等の情報を参考にして p の場所を決定する。

(b) 該当するブロックに埋め込む情報が 1 である場合には、そのブロックの量子化後に $AC(p)$ に 1 を挿入する。

(c) 使用した量子化テーブルの $AC(p)$ に相当する部分を 1 に変更したものをハフマン符号化する。

- 復号側 1 (付加情報を取り出す場合: 図 4(b) 参照)

(a) 量子化テーブルの 1 に相当する部分から p を得る。複数存在する場合は、最も高域側のものを選択する。

(b) 各ブロックごとに $AC(p)$ を調べ、埋め込まれた情報を判断する。

(c) $AC(p)$ を 0 に戻した後、復号する。

- 復号側 2 (付加情報を取り出さない場合: 図 4(c) 参照)

(a) 標準の JPEG 復号処理と同じ処理を行う。

(b) 各ブロックに付加された情報は単なるノイズとなる。ただし、該当する係数の量子化テーブルが 1

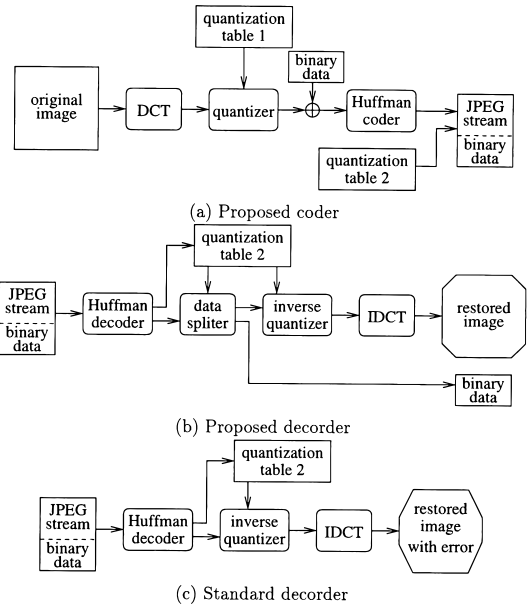


図 4 提案するシステムの入出力関係
Fig. 4 Proposed system.

となっているため、画像に与える影響は小さくなっている。

これらの手順とは別に、符号化-復号間において、どのブロックに情報を埋め込んだかどうかが知らせる必要がある。すなわち、両者の間で埋め込んだブロックの数や使用したブロックの場所に関する同意が必要である。後述するシミュレーションでは、使用するブロックの総数をバイナリデータとともに埋め込む例を示す。

4. 従来法との比較

提案法と同様にバイナリデータを扱うことができる電子透かしの方法が存在する。その中で 2.2. における条件 (a) ~ (c) を満足するものに、文献 [5], [6] がある。ここでは、これらの従来法と提案法を比較検討する。

文献 [5], [6] の方法は、提案法と同様に DCT 領域における方法であり、特定の AC 係数 $AC(k)$ に 1 ビットの情報を挿入する。具体的には、DCT 係数を量子化し丸め操作を行う際、ブロックに埋め込む情報が 0 である場合には、最も近傍の偶数値に設定する。逆に埋め込む情報が 1 である場合には、最も近傍の奇数値に設定する。復号側では注目する AC 係数の最下位ビットより、提案法同様に埋め込まれた情報を完全に復号することが可能である。ゆえにこの方法は量子化後の

AC 係数の最下位ビットを変更する方法である。

まず、提案法との第 1 の相違点は、情報を埋め込む際に、AC 係数そのものを変更せず、AC 係数の最下位ビットを変更する点にある。この場合、上位ビットの情報は埋込み後もほとんどそのまま残る。そのため、提案法と同様に量子化テーブルを変更してしまうと、逆にもととの AC 係数の値をひずませることになる。したがって、従来法では量子化テーブルの変更は行っていない。しかしながら、高域の係数に情報を埋め込んだ場合、一般に逆量子化の際の量子化テーブルが大きな値となり、最下位ビットの変更であってもその影響は大きくなってしまふ。

このことは、次に述べるシミュレーションにおける検討においても確認される。

5. 情報の埋込み例

5.1 シミュレーション方法

テスト画像としてグレイスケールの Barbara を、埋め込むデータとして本論文日本語アブストラクト (602 byte JIS code テキスト) を使用した。このシミュレーションでは、情報を左上隅のブロックから順番に埋め込むものとした。またその際に、埋め込むブロックの個数を 16 ビットのバイナリで示し、テキストデータの先頭にヘッダとして付加し、画像に埋め込んだ。したがって、今回の例では、全 5280 ブロックのうち、4832 ブロック (ヘッダ 16 ビット + テキストデータ 4816 ビット) を使用している。

使用する量子化テーブルは、表 1 を基本とし、Q-factor に従って各係数を Q-factor/50 倍して得られる。Q-factor が増大するほど、量子化ステップが増大するため、圧縮率が高くなり、低精度な画像となる。

ここでは最も簡単な例として最終 AC 係数 ($p = 63$) に信号を埋め込む場合を考える。この場合、復号側では EOB の存在の有無により、1 or 0 を判断することもできる。表 2 は、Q-factor = 50 のときの復号側における量子化テーブルである。

一方、従来法としては、4. で示した LSB にビットを埋め込む方法を実行した。文献中では秘^{とく}匿性を考慮し、情報を埋め込む AC 係数の位置を鍵を用いて可変にしている。しかしながら、今回はバイナリデータの埋込みを目的としているため、提案法と同様にすべてのブロックにおいて固定な AC 係数に情報を埋め込むこととした。ただし、従来法ではどの AC 係数に情報を埋め込んだかという情報は符号化列には埋め込ま

れていないため、符号側・復号側両者の間に同意がとれているものと仮定した。情報を埋め込む AC 係数として、63, 31, 15 の 3 箇所を選び、提案法との比較を行った。

5.2 情報を取り出さない環境における復元

最初に復号側で付加情報を取り出さない場合を検討する。この場合、付加された情報は実際の画像に対するノイズとなる。ここでは、情報を付加した符号化列に対し、標準の JPEG の復号処理を行った場合の画質劣化について調べる。

表 3 は原画像に対する圧縮画像の PSNR を比較したものである。左側は情報を付加しなかった場合の JPEG 符号化列の PSNR であり、中央は情報を埋め込んだ場合の JPEG 符号化列の PSNR である。信号埋込みを行った場合でも行わなかった場合とほとんど同程度の PSNR であり、画質劣化はほとんど見られないことがわかる。

次に、量子化テーブル入れ換えの効果を考察する。表 3 の右列は、符号化側で量子化テーブルを変更しなかった場合の PSNR である。これは、従来法で情報を埋め込む係数を 63 にした場合に相当する。テーブルを変更した場合に比べ、大きく画像が劣化していることがわかる。特に、圧縮率が大きくなると、大きな値の量子化テーブルを使用することになるため画像の劣化が顕著になる。以上より、量子化テーブルの変更が有効であることが確認できる。

また、提案法及び従来法において情報を埋め込む位置を変更した場合の表を表 4 に示す。提案法は AC 係数の置換えをしてしまうため、低域の係数に埋め込んだ場合に画質劣化が大きくなることがわかる。したがって、提案法では 3.2 で示したようになるべく高域の係数を使用すべきであることがわかる。逆に、従来法は量子化テーブルの値が小さくなるために、低域の

表 3 JPEG 復号による画像劣化の評価 (埋込み情報あり): PSNR[dB]

Table 3 Evaluation of image distortion (data embedded): PSNR[dB].

Q-factor	埋込みなし	埋込みあり	
		テーブル変更あり	テーブル変更なし
5	44.2404	44.2039	43.1998
10	40.7913	40.7844	38.9603
20	37.5403	37.5372	34.4426
40	34.3546	34.3539	29.5005
60	32.4469	32.4459	26.471
80	31.0561	31.0556	24.2079

表 4 従来法との比較 (埋込み情報あり): PSNR[dB]
Table 4 Comparing with conventional method (data embedded):PSNR[dB].

(a) 提案法

Q-factor	k = 63	k = 31	k = 5
5	44.2042	38.7115	36.8265
10	40.7849	37.3302	35.9344
20	37.5388	35.2434	34.4449
40	34.3541	32.3469	32.1978
60	32.4456	30.1858	30.4457
80	31.0558	28.5206	28.9972

(b) 従来法

Q-factor	k = 63	k = 31	k = 5
5	43.4703	43.9173	44.2099
10	39.4239	40.2855	40.7480
20	35.1689	36.6749	37.4668
40	30.4971	32.8609	34.2221
60	27.5946	30.4271	32.2480
80	25.4150	28.6238	30.8242

係数に埋め込むほど画質劣化が少なくなることがわかる。圧縮率が低く、低域の係数に埋め込んだ場合は、提案法で $k = 63$ に埋め込んだ場合と同程度の画質劣化であるが、圧縮率が高くなると画質劣化が顕著に現れることが確認できる。

5.3 情報を取り出せる環境における復元

次に復号側で付加情報を取り出す場合を検討する。この場合、付加情報を完全に取り出すことは可能である。しかしながら、最終係数が零でない場合、すなわちブロックに EOB が含まれない場合には、そのブロックでは画像情報の欠落が生じる。表 5 に Q-factor と圧縮率及び EOB の含まれないブロックの数の関係を示す。圧縮率が低い場合には、高域の成分も値をもつために、EOB が存在しないブロックが数 % 存在する。しかしながら、1/5 程度の圧縮率であれば、画像情報を欠落することなく情報を付加することができる。実際に専用の復号器を用いて情報を取り出した後の PSNR を表 6 に示す。Q-factor が 20 以上の場合にはもとの JPEG 画像への劣化が生じてないことがわかる。

5.4 情報付加によるファイルサイズの増加

次に提案法を用いて作成した JPEG 符号化列のファイルサイズと情報を埋め込まなかった場合の符号化列のファイルサイズの比較を図 5 に示す。情報を付加したことにより、ファイルサイズが増加していることがわかる。これは最終係数に 1 が入った場合に、途中の零値のランレングスを符号化する必要があるためであ

表 5 Q-factor と EOB 不在のブロック数の関係
Table 5 Relationship between Q-factor and the number of blocks without EOB.

Q-factor	EOB 不在のブロック数
5	219 (4.15 %)
6	90 (1.70 %)
7	43 (0.81 %)
8	23 (0.44 %)
9	10 (0.19 %)
10	5 (0.09 %)
15	1 (0.02 %)
20	0 (0 %)
25	0 (0 %)

表 6 情報取出し後の画像劣化の評価: PSNR[dB]
Table 6 Evaluation of image distortion (data extracted):PSNR[dB].

Q-factor	埋込みなし	情報取出し後
5	44.2404	44.2146
10	40.7913	40.7901
20	37.5403	37.5403
40	34.3546	34.3546
60	32.4469	32.4469
80	31.0561	31.0561

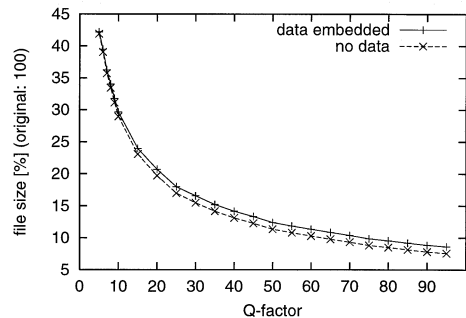
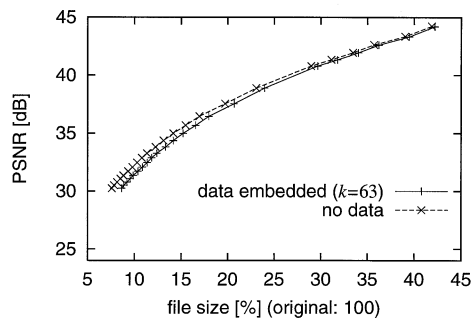


図 5 情報付加によるファイルサイズの増加

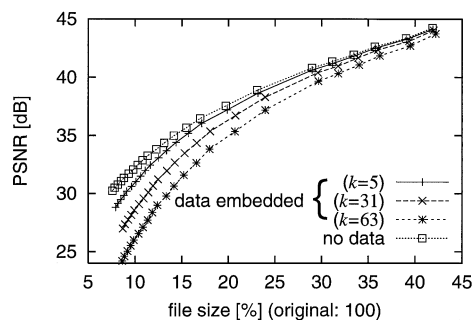
Fig. 5 Relationship between the Q-factor and the file size of the bitstreams.

る。したがって、大きな値の Q-factor を選択するほど、データを挿入しない場合に比べ AC 係数の零値の数が増えるために、ファイルサイズの増加が激しい。もしファイルサイズに制限がある場合には、情報付加によって増えてしまうサイズ分だけ、大きな Q-factor を用いて量子化する必要がある。

図 6 は表 4 の結果を整理して、JPEG 符号化データのファイルサイズとデータを取り出さない場合の PSNR の関係を示したものである。比較のため情報を埋め込まなかった場合も併記した。同一ファイルサイズで比べた場合には、提案法と従来法はともに大きな



(a) 提案法



(b) 従来法

図 6 ファイルサイズと PSNR の関係

Fig. 6 Relationship between the file size of the bitstreams and the PSNR.

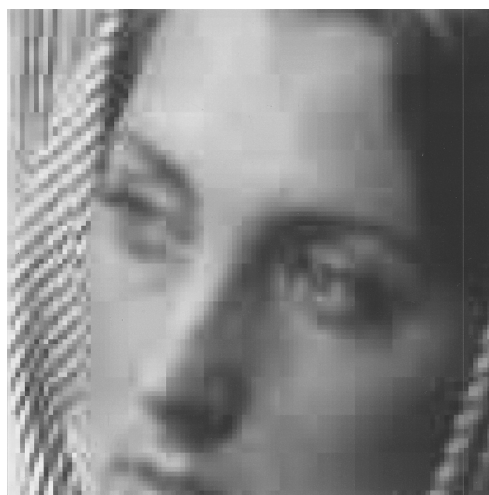
Q-factor を使用する必要があるために、情報を付加しなかった場合よりも PSNR が低下する．特に大きな Q-factor，すなわち小さなファイルサイズを選択するほど、両者ともその傾向は顕著になる．従来法において大きな Q-factor の選択は、逆量子化の際、挿入したデータによるひずみを拡大する．また、埋め込む場所を高域にする、すなわち k の値を大きくするほど画質劣化が大きくなっていることがわかる．これは、逆量子化におけるテーブル値が高域になるほど大きくなるためである．一方、提案法において Q-factor が大きい場合は、情報埋込みにより $k = 63$ に有意係数が存在するために JPEG の EOB (エンド・オブ・ブロック) コードによる情報量圧縮が働かないので、埋込みを行わないときよりもファイルサイズが増大する．したがって、ファイルサイズを一定に制限する場合には、非埋込み時よりも大きな Q-factor を選択する必要がある、これが画質劣化の原因となっている．

例として、実際にファイルサイズを約 1/11 に制限した場合の詳細を図 6 から選び、結果を表 7 に示す．

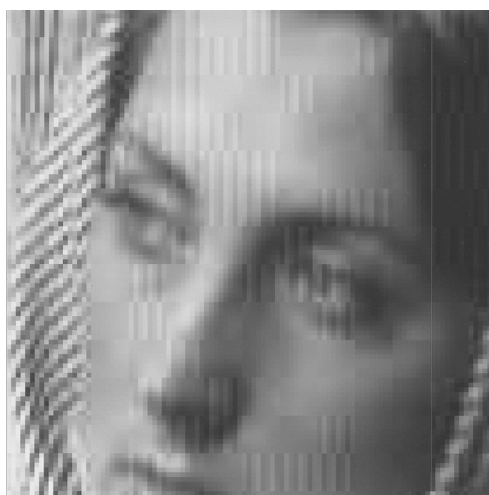
表 7 ファイルサイズと PSNR の関係

Table 7 Relationship between file size of the bitstreams and the PSNR.

	埋込みなし	提案法 ($k = 63$)	従来法 ($k = 5$)
ファイルサイズ [Byte]	29778	29774	29824
(オリジナル比)	(8.81 %)	(8.81 %)	(8.82 %)
Q-factor	75	91	80
PSNR [dB]	31.3423	30.4426	29.8259



(a) 提案法 ($k = 63$)



(b) 従来法 ($k = 5$)

図 7 表 7 の条件の下で埋込みを行った復号画像

Fig. 7 Restored Images for the conditions of Table 7.

また、その条件のもとでデータを取り出さずに復号した画像を図 7 に示す．提案法は、従来法で埋め込む位置を $k = 5$ に選択した場合よりも 0.6 [dB] 程度良い

結果が得られている。また、図 7 より、どちらの画像も Q-factor が大きいために、ブロックひずみが多少見られるが、従来法ではブロックひずみ以外にも視覚的に目障りなひずみの発生が確認できる。これは、従来法は視覚的に目立ちやすい低域の係数へ情報を埋め込んでいるためである。

6. む す び

本論文では、JPEG 符号化列へのバイナリデータの埋込み法について検討した。本方法では、埋め込んだ符号化列からデータを完全に復元可能である。また符号化列は標準の復号器でそのまま復号が可能であり、その際にも画質の劣化が視覚的に目立ちにくい。これは、提案法が視覚的に影響が少ない高域の係数に情報を埋め込んでいること、また量子化テーブルの値を変更していることによるものである。

謝辞 本研究の一部は文部省科学研究費補助金萌芽的研究課題番号 11875088 の援助によるものである。

文 献

- [1] 安田 浩 (編), マルチメディア符号化の国際標準, 丸善, 1991.
- [2] 貴家仁志, 小林弘幸, 野口祥宏, “領域再分割のための JPEG 画像へのデータ埋め込み法,” 信学技報, CAS 98-6, June 1998.
- [3] 高橋史忠, “欧州の産学協同プロジェクト, 電子透かしで映像の著作権保護,” 日経エレクトロニクス, no.680, pp.11-12, 1997.
- [4] 井上 彰, 電子透かし—マルチメディア時代の暗号システム, 丸山学芸図書, 1997.
- [5] 松井甲子雄, 画像深層暗号—手法と応用, pp.139-147, 森北出版, 1993.
- [6] 松井甲子雄, 電子透かしの基礎—マルチメディアのニュープロテクト技術,” pp.57-60, 森北出版, 1998.

(平成 11 年 5 月 20 日受付, 11 月 17 日再受付)



野口 祥宏

1985 神戸大卒。同年旭化成工業 (株) 入社。1994-1996 カリフォルニア大学客員研究員。画像処理, VLSI 設計に関する研究に従事。IEEE 会員。



貴家 仁志 (正員)

1980 長岡技科大・工・電気電子システム卒。1982 同大学院修士課程了。同年東京都立大工学部電気工学科助手。現在、同大学院工学研究科電気工学専攻教授。1995 年 10 月~1996 年 3 月シドニー大 (オーストラリア) 客員研究員, 工博。マルチレート信号処理, 適応信号処理及び画像処理に関する研究に従事。IEEE Transaction on Signal Processing, 本会論文誌 (A) 編集委員。著書「高速フーリエ変換とその応用」, 「デジタル信号処理」, 「マルチレート信号処理」。電子画像学会, 映像メディア学会, IEEE 各会員。



小林 弘幸 (正員)

1992 東京都立大・工・電気卒。1997 同大学院博士課程了。同年東京都立工業高等専門学校電気工学科講師。現在同高専助教授。博士 (工学)。マルチレート信号処理及び画像処理に関する研究に従事。