

1980年暗号と情報セキュリティ

シンポジウム資料

1986年 2月 6日～ 2月 8日

算術符号を利用した 画像深層暗号化

鈴木 寿 有本 卓

大阪大学基礎工学部機械工学科

〒560 大阪府豊中市侍兼山町1-1

06-844-1151 内線4506

あらまし

一見したところ単に画像データを伝送しているようにしか見えないが、実はその陰にもっと重要な別の情報を隠し持つ、というような形態の暗号化、すなわち画像深層暗号化を提唱する。伝送したいデータを攪乱用画像データに混合させて暗号文を生成し、既存の画像伝送系を通じて送る。なるべく画質を落とさず、しかも高い情報伝送率が達成できるように、算術符号を利用した暗号化アルゴリズムが工夫される。

Embedding-in-Image Data Encryption with
Arithmetic Coding

Hisashi Suzuki and Suguru Arimoto

Faculty of Mechanical Engineering
Department of Engineering Science
Osaka University
Toyonaka, Osaka 560, Japan

Abstract

A new method of data encryption of which cryptogram looks like a sequence of pure image data is proposed, which we call the embedding-in-image data encryption. The data desired to transmit are mixed with some image data to produce a cryptogram, which is transmitted through an existing system for digital image transmission. A kind of arithmetic coding is utilized in order to construct an algorithm of encryption that achieves a well-manufactured cryptimage and a high transmission efficiency.

盗聴者が通信系に割り込んで入手した通信文が意味不明な記号列ならば、暗号文であることに気付き解読しようとするだろう。ところが通信文が意味のある記号列ならば、それが通信内容そのものであると確信するに違いない。そこで、表面上は意味のある記号列の体裁をなすが実はその陰にもっと重要な情報を隠し持つ、といった形態の暗号文を生成できれば、盗聴者から解読のきっかけをほとんど完全に奪うことができる。一このような暗号を、通信文の深層部に情報が隠されるというニュアンスで、深層暗号と呼ぶ。

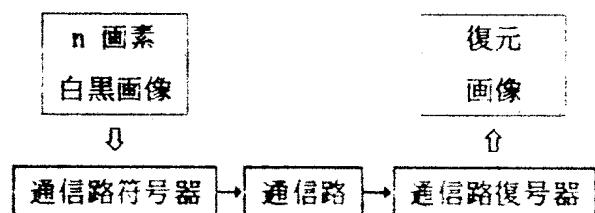
深層暗号は、秘密伝送したい本来の情報を、予め準備した攪乱用情報の中へ紛れ込ませる。その際、前者の情報量に比べ後者の情報量を十分大きくとることによって、後者が暗号文の統計的構造を支配するように仕向ける。それゆえ、深層暗号を実現するためには十分量の攪乱用情報を準備しなければならない。われわれは、ディジタル画像が膨大な情報を有する事実に着目し、画像データを攪乱用情報として利用する深層暗号の一方式、すなわち画像深層暗号を考案した。

今日、集積回路の急速な進歩のおかげで、大量の画像データを手軽に伝送する技術は日常的なものとなった。こうした例は、現代オフィスにかかせないファクシミリ電信、計算機ネットワークを利用したメール、また最近開発されたフロッピー・ディスク記録方式の写真、など数多く見受けられる。さて、既に述べたように攪乱用情報は大量に随伴させなければならないので、深層暗号を載せる通信系もそれに見合うだけの通信容量を持つ必要がある。しかるに、上述のような現存する伝送装置は、まさにこの要請を満たしている。ここで提唱する方式は、そのような既存の伝送装置の入口と出口に、暗号器と復号器を付加するだけで実現できる。

この章では、画像データを利用した比較的簡単な深層暗号を例にあげて、暗号化・復号化のメカニズムを具体的に述べる。

第2-1節 通信系の設定

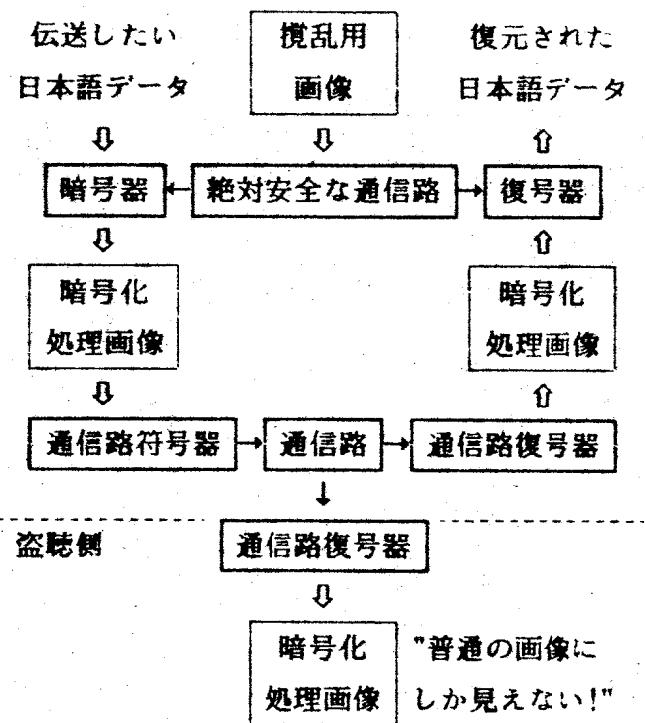
いま、一画面当たり n 画素から成る白黒画像データを何画面も伝送する通信系が与えられたとする。ただし、伝送途中で加わる外乱に対しては誤り訂正符号を導入するなどの十分な対策が講じられており、伝送誤りは、理論上“雑音のない通信路”によってモデル化できる程度にほとんど生じないものとする。



秘密伝送したい情報は、次の 64 個の記号：
 アイオ がウコ オシセツ ナツテ ナヌノ ハツハ
 マミムズ カヨ ラルロ ワン アイエオ ケヤヨ “
 、。？「」（）_

(注：_ は空白の意)
 から成る日本語の文章を、1 記号当り 6 ビットのセグメントに変換することによって得られる二値系列である。以後、このデータを日本語データと呼ぶ。

上述の通信系を基礎にして、次のような暗号通信系を構成する：



が著しいほど、盗聴者が隠れた暗号の存在に気付く可能性は高くなる。

以上の方針において鍵の役目を果たすものは、^① 搅乱用画像データおよび^② u_1, \dots, u_m である。鍵は、暗号文の伝送に先立ち、別の絶対安全な通信路を経由して予め復号器へ配布されなければならない。復号器は、鍵を基にして、次のような手順で日本語データを復元する。

□【復号化法 1】

- 1) 搅乱用画像データと m の値を受け取る。
- 2) 暗号文を、暗号化のときと同様に一次元座標軸上に並べる。
- 3) それぞれの $i=1, \dots, m$ に対し、 u_i 上の要素と第 i 番目の搅乱用画像データとの排他的論理和をとり、その結果を第 i 番目の日本語データとする。

第2-2節 初歩的な暗号化法

誰でもまっさきに思いつくのが、伝送したい情報を画像全体に一緒に混入させる、という方法である。

容易に想像がつくように、この方法は簡単に実現できる反面、よほど小さい m を選ぶのでない限り、それほど高い搅乱効果を期待できない。次の例はその欠点を顕著に示している。

□【暗号化法 1】

- 1) 搅乱用画像データを、 $0, 1, \dots, (n-1)$ の座標を持つ一次元座標軸上に並べる。
- 2) 情報伝送率と画質を左右する整数パラメータ m を、 $1 \leq m \leq n$ の範囲で選ぶ。
- 3) m 個の座標 u_1, \dots, u_m をランダムに選ぶ。
- 4) それぞれの $i=1, \dots, m$ に対し、 u_i 上の画像データと第 i 番目の日本語データとの排他的論理和をとり、その結果を u_i へ記録する。
- 5) 座標軸上の要素の並びを暗号文とする。□

ここで情報伝送率とは、日本語データ列の長さと暗号文の長さとの比 $100 \times m/n$ (単位: %) を意味する。また画質の良し悪しは、日本語データが畳み込まれることによって生ずる画像の歪の、感覚的な度合を表わす。暗号化処理による画質の劣化

■《実験 A》

$n=256 \times 256=65536$ および $m=3930$ (655 字に対応、情報伝送率 6%) とおき、SIDBA 画像を二値化したものを使い、"資治通鑑選"の一節を暗号化してみた。

ヨノウニ、コレ_ヨンノ_コ"シクソハ、ミケ_タカフ_マイテルヒ
トビ"トノ_カタ_オモチ_ナリマタ。コウシテ_ミマスト、タカフノ
カソノ_モカ"シニ_ツムケタ"ト_イコトガ"アリカガ"コ"サ
ヤセん。ヒ"シ"ヨ_ホセキニ_ツキマテハ、イ"モ_シノ_ケニノモ
ノテ"ハ_コ"サ"ヤセん。シモ_オモチハ_カズ"オク_ユク_オモチ
ナツ_オラレマス。リニモ_カワラズ"、シ"ンア"ツラ_オモチ_ナツ
_イコトニ_ツ行ハ_ツケ"ハ_カ、テキセツナ_シ"ンフ"ツ"ツ"カ
イ_シ"ンフ"ツ"ツ"ツ"カ"ハ"モ_カク"エス"、タダニイコカ_けか_日
片"モ_ナラズ"、シノ_ケニノ_シッシンテ"ナ代ノハ"ス"テ_ヤ
ル。タカフノカソノモタ_オモチハ_オラレマス。ヨレ"ハれ

シスルモノハ_た外_ウト_ビ"シヨ、オカガク、オカセキノタフ"行
"アリ、カルス"スルモノハ_ヒトヤ_タミテアト_ウコトニ_カリマス。ウタ
ケシハ_マサ_タキ"ヨウニ_キ行_オリマス。「タキシハ_ト」ンナ_ツカ
レ"モ_タニ_ユス"ルコトハ_シイ。ソレニ_アヨウタカト_ナル。コ
ウカ"、ボ"カイヘ_ト"ンナ_チ付_カ"レモ_トイワニ_ウケルも。
ソレニ_アヨウタカト_ナル。オカシ"マハ_ト"ンナ_ミブ"ソノ_ヒクイ
モノテ"モ_カシ"ナイ。

(↑: ここまでで 655 字)

漢字仮名混り文: このように、これら四人のご主君は、皆他国から参っている人々の力をお用いになりました。こうして見ますと、他国からの仕官の者が秦に背くなどということがあるわけがございません。美女や宝石につきましては、いずれも秦の國のものではございません。しかも王様は数多くこれをお用いになっておられます。それにもかかわらず、人物をお用いになるということについてはそうではなく、適切な人物とそうでない人物との区別も考えず、正しいことか否かの論議もなさらず、秦の國の出身でない者はすべてやめさせ、他国よりの仕官の者を追い払っておられます。これでは重んずるものは何かというと美女、音楽、宝石の類であり、軽んずるものは人や民であるということになります。私はまた次のように聞いております。“太山はどんな土くれでも他に譲ることはしない。それゆえあのような高さとなる。黄河、渤海はどんな小さな流れも厭わずに受入れる。それゆえあのような深さとなる。王者はどんな身分の低い者でも軽んじない。”

図A-0は原画像、図A-1は暗号化処理された画像である。このような画像では、細工を施したことがすぐばれてしまう。 ■

■《実験B》

実験Aよりも良好な画質を得ることをもくろみ、M=1110 (185字に対応、情報伝送率 1.7%) において、同じ試料の最初の部分:

ヨヨウニ、コレ … シモ_材マハ_

(ここまでで 185 字: ↑)

を暗号化してみた。

図B-0は原画像、図B-1は暗号化処理された画像である。情報伝送率を 6% から 1.7% へ落としたにもかかわらず、画質はそれほど良くなっていない。 ■

第2-3節. もう少し賢い暗号化法

前節のように、伝送したい情報を画像全体に一様にばらまいたのでは、それほど良い画質が得られない。これに代わる方法は、情報を、画像上の白・黒の変化の境目付近に集中させて置き込むことである。

いま搅乱用画像データを、 $0, 1, \dots, (n-1)$ の座標を持つ一次元座標軸上に並べたとする。このとき、次の条件を満たす座標 u の集合 U を定義する:

条件: 原画像上で、 u に対応する画素の輝度と、それを囲む 8 画素のいずれかの輝度とが反転している。

すなわち、パターン  および  の中心画素に対応する座標は U に属さないが、それ以外のパターン 一例えば  や  などの中心画素に対応する座標は、 U に属する。

新しい暗号化法は、 U を用いて次のように記述される。(暗号化法 1 と異なる部分はステップ 2 と 3 である。)

□【暗号化法2】

- 1) 搅乱用画像データを、 $0, 1, \dots, (n-1)$ の座標を持つ一次元座標軸上に並べる。
- 2) 情報伝送率と画質を左右する整数パラメータ m を、 $1 \leq m \leq |U|$ の範囲で選ぶ。
- 3) U の中から m 個の座標 u_1, \dots, u_m をランダム

に選ぶ。

- 4) それぞれの $i=1, \dots, m$ に対し, u_i 上の画像データと第 i 番目の日本語データとの排他的論理和をとり, その結果を u_i へ記録する.
- 5) 座標軸上の要素の並びを暗号文とする. □

復号化法は, 先の復号化法 1 と同じである.

□【復号化法2】

- 1) 撥乱用画像データと m の値を受け取る.
- 2) 暗号文を, 暗号化のときと同様に一次元座標軸上に並べる.
- 3) それぞれの $i=1, \dots, m$ に対し, u_i 上の要素と第 i 番目の撥乱用画像データとの排他的論理和をとり, その結果を第 i 番目の日本語データとする. □

次の実験からわかるように, 上述の改良によって画質はかなり向上する.

■《実験A》

2-2節の実験Aと同じ条件(655字, 6%)の下で, 暗号化法2を適用してみた.

図A-0は原画像, 図A-2は暗号化処理された画像である. あたかも質の悪い紙に万年筆で記録したように, 線がにじんでいる. この画像をそのまま伝送したのでは, 暗号化処理を施したことばれるだろうが, 情報伝送率をさらに低く設定しさえすれば実用に耐え得ることが予想される.

■《実験B》

2-2節の実験Bと同じ条件(185字, 1.7%)の下で, 暗号化法2を適用してみた.

図B-0は原画像, 図B-2は暗号化処理された画像である. 線のにじみが多少残っている. ■

第3章. 算術符号を利用した暗号化法

一第二段階

前章では, 画像に情報を畳み込むことによって生ずる歪の視覚的効果を考慮しつつ, 深層暗号の基本的な構成法について述べた. 残る問題は, "白・黒の変化の境目付近に情報を集中投入する" という方針を保ったままで, もっと効率よく情報を送れるような方法を開発することである. われわれは, 算術符号を利用してこの目的を実現する.

第3-1節. 準備

まず, 以前に行った約束のいくつかを修正すると共に, 新たな記号を導入する.

伝送したい情報は日本語データではなく, 0と1を等確率でとる互いに独立な確率変数の列 $\{X_i\}$; $i=1, \dots, \infty$ であると仮定する. これを**情報源データ**と呼ぶ. (実際には, 情報源データすなわち二値乱数は, 日本語データに対し圧縮符号化を施すことによって, あるいはランダムさを増加させる通常の暗号化を施すことによって生成される.)

撥乱用画像において, U に対応するそれぞれの画素の輝度が暗号化処理の結果反転する確率を, ϵ で表わす. このとき, 暗号化法2の下で

$$0.5 \times m/|U| = \epsilon$$

なる関係が成立立つ.

一枚の画像に混入させることができる情報源データの情報量を I (単位: bits) で表わす. さらに, 情報伝送率を

$$R = 100 \times I/n \quad (\text{単位: \%})$$

のように定義する. このとき, 暗号化法2に対し

$$I = m = 2 \epsilon |U| \quad (\text{bits}),$$

$$R = 200 \epsilon |U|/n \quad (\%)$$

となる.

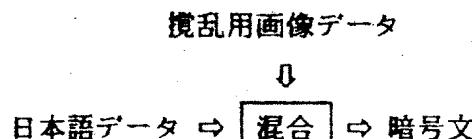
第3-2節 不定長符号の導入

あらゆる有限長二値系列の集合 $\{0,1\}^*$ の部分集合のうち、次の条件を満足するものを、 S で表わす：

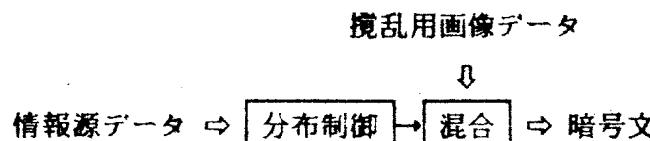
条件：どのような有限長二値系列も、 S 内の要素を用いて一意に分解することができる。ある正整数 k に対し $\{0,1\}^k$ から S の上への一対一写像が存在するとき、これを f で表わし、逆写像を g で表わす。符号化、復号化の数学的定義が f, g であるような種類の符号 (f, g) を、不定長符号 (variable length code) と呼ぶ。

それぞれの $i=1, \dots, k$ に対し、 $X_1 \cdots X_i$ を X^i で表わす。また、 $f(X^k)$ を $Y_1 \cdots Y_L$ あるいは Y^L で表わす。ここで L は $f(X^k)$ の長さをとる確率変数である。 f は、データの確率的構造を変化させる機能を持つ。

暗号化法2は、単に日本語データと画像データとを混合して暗号文を生成するという方法であった。



これに対し、以下で提唱する暗号化法3は、まず情報源データに f を施し、分布を変えておいてから画像データと混合させるという方法である。



後述するように、このような二段階の操作を行うことによって情報伝送率を向上させることができる。

いま、ある不定長符号 (f, g) を用いて、次のような暗号化・復号化法を定義する。

□【暗号化法3】

- 1) 情報源データに f を施すことによって、 Y^L を得る。
- 2) $L=|\mathcal{U}|$ と設定し、 Y^L を日本語データと考えて暗号化法2を適用する。 $(k > |\mathcal{U}|$ の場合は、 Y^L に対し前方から順番に、暗号化法2を繰り返し適用する。)

□【復号化法3】

- 1) 復号化法2を用いて、 Y^L を復元する。
- 2) Y^L に g を施すことによって情報源データを復元する。

さて、任意の (f, g) に対し次式が成立する：

$$I(X^k \wedge Y^L | L) = \sum \Pr\{L=i\} H(Y^i | L=i) \\ \leq \sum \Pr\{L=i\} [H(Y_1) + \dots + H(Y_i)].$$

ここで、和はあらゆる正整数値 i についてとる。もし (f, g) を適切に定めることで画質を暗号化法2のそれに近づけることができるならば、すなわち、 $\{Y_i\}$ が

$$\Pr\{Y_i=0\} = \epsilon, \quad \Pr\{Y_i=1\} = 1-\epsilon$$

なる独立同一分布を達成するような (f, g) を構成できるならば、不等号の間隙は小さくなり、

$$I(X^k \wedge Y^L | L) \times |\mathcal{U}| / EL$$

は

$$h(\epsilon) |\mathcal{U}| \text{ (bits)}$$

に近づく。ここで、 $h(\epsilon)$ はバイナリ・エントロピーを表わす。したがって、情報伝送率 R は

$$100 \times I/n = 100 h(\epsilon) |\mathcal{U}| / n \text{ (%)}$$

に近づき、暗号化法2の

$$\frac{100 h(\epsilon) |\mathcal{U}| / n}{200 \epsilon |\mathcal{U}| / n} = h(\epsilon) / (2\epsilon)$$

$$= (1 - \log \epsilon) / 2 - O(\epsilon)$$

(注：対数の底 = 2)

倍近くまで改善されることとなる。

このようにして $Y^L = 110$ が得られる。

f, g は、次の方法で実現できる：

□【 f : 符号化法】

1) 次のパラメターを決定する：

入力データ長 k ,

希望する出力分布 $(\epsilon, \bar{\epsilon})$ ($\epsilon + \bar{\epsilon} = 1$).

2) x^k を入力する.

3) $\mathcal{S} \leftarrow \{0,1\}^k$ とする. (\mathcal{S} は、長さ k の二値系列から成る集合を表わす "変数" である.)

4) $|\mathcal{S}|=1$ となるまで、以下の操作を繰り返す：

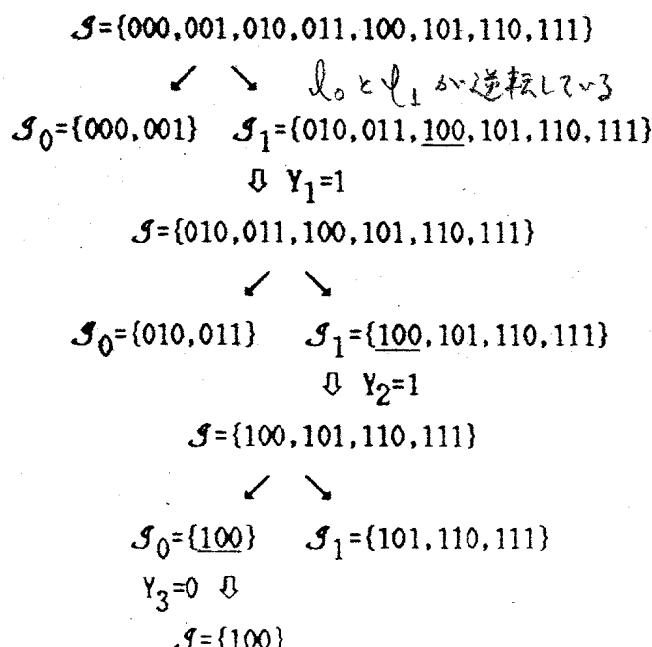
- \mathcal{S} を、空でない 2 つの部分集合 $\mathcal{S}_0, \mathcal{S}_1$ へ分割する. その際 $|\mathcal{S}_0|/|\mathcal{S}|, |\mathcal{S}_1|/|\mathcal{S}|$ ができる限り $\epsilon, \bar{\epsilon}$ を近似するようにする.
- $x^k \in \mathcal{S}_y$ なる y を、 Y^L の構成要素の一つとして出力する.
- $\mathcal{S} \leftarrow \mathcal{S}_y$.

5) 終了. □

要するに、 x^k が属する部分集合を逐次に探索し、その軌跡を Y^L に記す作業を行っていることになる。

■【例】

$(\epsilon, \bar{\epsilon}) = (0.3, 0.7)$, $k=3$ とおく. $x^k = 100$ に対する符号化操作は、下図のように行う：



復号化を行うには、与えられた $Y^L = 110$ を基に \mathcal{S} を絞り込んでゆくことによって、最終的に唯一の要素から成る $\mathcal{S} = \{100\}$ を得、その要素 100 を x^k とみなせばよい。(詳細は、下記のアルゴリズムを参照せよ。) ■

□【 g : 復号化法】

1) 符号化のときと同じパラメターを採用する.

2) $\mathcal{S} \leftarrow \{0,1\}^k$.

3) $|\mathcal{S}|=1$ となるまで、以下の操作を繰り返す：

- \mathcal{S} を、符号化のときと同じ規則に従って、 $\mathcal{S}_0, \mathcal{S}_1$ へ分割する.
- Y^L の構成要素を一個入力し、これを y で表わす.
- $\mathcal{S} \leftarrow \mathcal{S}_y$.

5) \mathcal{S} の唯一の要素を、 x^k として出力する. □

要するに、 Y^L を基にして x^k が属する部分集合を逐次に指定することにより、 x^k の候補を徐々に絞り込んでゆき、最終的に一個の候補を見出すという作業を行っていることになる。

$\mathcal{S}_0, \mathcal{S}_1$ の大きさを上のようにコントロールするとき、 Y^L の第 i 番目の要素 Y_i を処理する時点で、次の関係が成り立つ：

$$\Pr\{Y_i = 0 | x^k \in \mathcal{S}\} = \Pr\{x^k \in \mathcal{S}_0 | x^k \in \mathcal{S}\} = \epsilon,$$

$$\Pr\{Y_i = 1 | x^k \in \mathcal{S}\} = \Pr\{x^k \in \mathcal{S}_1 | x^k \in \mathcal{S}\} = \bar{\epsilon}.$$

右辺の近似は、 $|\mathcal{S}|$ が大きいほど良い。したがって、出力分布をできる限り希望に近づけたければ、 k をできる限り大きくとる方がよい。

ここで問題なのは、 k を大きくとるほど \mathcal{S} の記憶と更新が困難になり、実現性を失っていくという事実である。この欠点は、 \mathcal{S} を二値系列の単なる集合ではなく、二値系列を整数とみたときの区間と考え、 \mathcal{S} から部分区間 $\mathcal{S}_0, \mathcal{S}_1$ への分割操作を、通常の計算機が有する算術演算子を用いて記述することによって、改善できる。すなわち、

\mathcal{S} を最小値と大きさとの組 $(\min \mathcal{S}, |\mathcal{S}|)$ で識別し、 $\mathcal{S}_0, \mathcal{S}_1$ を

$$\begin{aligned}\min \mathcal{S}_0 &\leftarrow \min \mathcal{S}, \\ |\mathcal{S}_0| &\leftarrow |\mathcal{S}| \times \epsilon, \\ \min \mathcal{S}_1 &\leftarrow \min \mathcal{S}_0 + |\mathcal{S}_0|, \\ |\mathcal{S}_1| &\leftarrow |\mathcal{S}| - |\mathcal{S}_0|\end{aligned}$$

によって算出するのである。

このアイデアに基づく符号化・復号化アルゴリズムは、1976年、Pasco[3] と Rissanen[4]により独立に提唱された。（“算術符号”の名は[4]に由来するが、今日では、その後提唱されたさまざまな改良版も含め、算術演算を利用する符号化方式はすべて算術符号と呼ぶ習慣が見受けられる。）ただし、[3]、[4]は、算術符号をデータ圧縮の目的で使用しているのに対し、われわれは分布制御の目的で使用することに注意しなければならない。このような使用目的の違いは、装置を組む上で決定的な差異をもたらす。すなわち上述の f, g は、データ圧縮の際は、符号化、復号化的役割が反転する。（詳細は Guazzo[5] を参照せよ。）

上述の方法において、 k はアキュムレーターの長さに応じた制限を受ける。Langdon-Rissanen[6] および Jones[7] は、 \mathcal{S} の大きさを適当なタイミングで回復させることによって k をアキュムレーター長と無関係に大きくとれるような符号化・復号化法を提唱した。以下では、[4]、[5] を参考に著者が考案した方法を紹介する。そこで用いられる変数 a, b は、 $\min \mathcal{S}, |\mathcal{S}| + \min \mathcal{S}$ に相当する役割を担う。（[4]、[5] で指摘された“桁上り (carry over)”に関する未解決問題は、“アンダー・フロー”的名を冠して自然な形で解決されている点に注意せよ。）

まず、次の記法を準備する：正整数 q および実数 $s \in [0, 1)$ に対し $\lfloor \cdot \rfloor_q, \lceil \cdot \rceil_q$ を

$$\begin{aligned}\lfloor s \rfloor_q &= \lfloor s2^q \rfloor 2^{-q}, \\ \lceil s \rceil_q &= \lceil s2^q \rceil 2^{-q}\end{aligned}$$

のように定義する。

この過程を繰り返す。

□【f: 算術符号化法】

1) 次のパラメターを決定する：

- 演算精度 q ,
- アンダー・フロー警告のしきい値 r ,
- 入力データ長 k ,
- 希望する出力分布 $(\epsilon, \bar{\epsilon})$.

ただし q, r, k は正整数、 $\epsilon, \bar{\epsilon}$ は $\{0, 2^{-q}, \dots, 1\}$ 内の実数であって、 $2^{-r} \epsilon \geq 2^{-q}$ なる条件を満足するように選ぶ。

- 2) a および b は、 q -bit 精度の固定小数型演算を行うアキュムレーターである。区間 $[a, b) \subset \{0, 2^{-q}, \dots, 1-2^{-q}\}$ は、 y^L の候補を指示する集合である。まず、 $[a, b) \leftarrow [0, 1)$ のように初期化する。
- 3) c もまた、 q -bit 精度の固定小数点演算を行うアキュムレーターである。先頭の q -bit のデータ $x_1 \dots x_q$ を入力し、
 $c \leftarrow x_1 2^{-1} + x_2 2^{-2} + \dots + x_q 2^{-q}$
のよう初期化する。

- 4) カウンターをリセットする。
(5-1) ~ (5-3)
- 5) カウンターが k になるまで、以下の操作を繰り返す：

- 5-1) もし $|[a, b)| \geq 2^{-r}$ ならば、カウンターを 1 増やし、以下の操作を行う：
 - $[a_0, b_0) \leftarrow [a, a + \lfloor |[a, b)| \epsilon \rfloor_q]$,
 - $[a_1, b_1) \leftarrow [b_0, b)$,
 - $c \in [a_y, b_y)$ なる y を、 y^L の構成要素の一つとして出力する。
 - $[a, b) \leftarrow [a_y, b_y)$.
 - ステップ 5-3 に跳ぶ。

- 5-2) もし $|[a, b)| < 2^{-r}$ ならば、“アンダー・フロー ($[a, b) = \emptyset$ となること) 発生の可能性あり”の警告を発し、以下の処理を行う：
 - $|[a, 1/2)| \geq |[1/2, b)|$ ならば
 $[a, b) \leftarrow [a, 1/2)$,
 - $|[a, 1/2)| < |[1/2, b)|$ ならば
 $[a, b) \leftarrow [1/2, b)$
 とおき直す。

◦ ステップ 5-3 に跳ぶ.

- 5-3) $[a, b] \subset [0, 1/2)$ または $[a, b] \subset [1/2, 1)$ である限り, 以下の操作を繰り返す:

◦ 前者の場合

$$[a, b] \leftarrow [2a, 2b],$$

◦ 後者の場合

$$[a, b] \leftarrow [2(a-1/2), 2(b-1/2)]$$

とおき直す.

◦ データを一個入力し, これを x で表わす.

$$◦ c \leftarrow (2c - \lfloor 2c \rfloor) + x2^{-q}.$$

- 6) $|[a, b]| = 2^{-q}$ となるまで, 以下の操作を繰り返す:

◦ $a' \leftarrow a + \max\{2^{-q}, \lfloor |[a, b]| \epsilon \rfloor_q\}$,

$$[a_0, b_0] \leftarrow [a, a'],$$

$$[a_1, b_1] \leftarrow [b_0, b].$$

◦ $c \in [a_y, b_y]$ なる y を, Y^L の構成要素の一つとして出力する.

$$◦ [a, b] \leftarrow [a_y, b_y].$$

- 7) 終了. □

□ 【g: 算術復号化法】

- 1) 符号化のときと同じパラメターを採用する.
- 2) $[a, b] \leftarrow [0, 1)$.
- 3) カウンターをリセットする.
- 4) カウンターが k になるまで, 以下の操作を繰り返す:

- 4-1) もし $|[a, b]| \geq 2^{-r}$ ならば, カウンターを

1 増やし, 以下の操作を行う:

$$◦ [a_0, b_0] \leftarrow [a, a + \lfloor |[a, b]| \epsilon \rfloor_q],$$

$$[a_1, b_1] \leftarrow [b_0, b].$$

◦ Y^L の構成要素を一個入力し, これを y で表わす.

$$◦ [a, b] \leftarrow [a_y, b_y].$$

◦ ステップ 4-3 に跳ぶ.

- 4-2) もし $|[a, b]| < 2^{-r}$ ならば, "アンダー・フロー ($[a, b] = \emptyset$ となること) 発生の可能性あり" の警告を発し, 以下の処理を行う:

◦ $|[a, 1/2]| \geq |\epsilon, 1/2|$ ならば

$$[a, b] \leftarrow [a, 1/2].$$

◦ $|[a, 1/2]| < |\epsilon, 1/2|$ ならば

$$[a, b] \leftarrow [1/2, b]$$

とおき直す.

◦ ステップ 4-3 に跳ぶ.

- 4-3) $[a, b] \subset [0, \epsilon)$ または $[a, b] \subset [\epsilon, 1)$ である限り, 以下の操作を繰り返す:

◦ 前者の場合, X^k の構成要素の一つとして 0 を出力し,

$$◦ [a, b] \leftarrow [2a, 2b]$$

とおき直す.

◦ 後者の場合, 1 を出力し,

$$◦ [a, b] \leftarrow [2(a-1/2), 2(b-1/2)]$$

とおき直す.

5) 終了. □

X^k および Y^L は先頭から末尾へ向かって少しずつ処理されるので, 暗号化・復号化法 3 をわずかに修正すれば, 分布制御と混合とを並列に実行できるようになる. この場合, $k=\infty$ とおくことが許される.

■ 《実験 A》

2-2節および2-3節の実験 A と同じ情報伝送率 6% の下で, 算術符号化法と共に暗号化法 3 を適用してみた. $\epsilon=0.05$.

図 A-0 は原画像, 図 A-3 は暗号化処理された画像である. 2-2節の実験 B (1.7%) にほぼ匹敵する画質が得られる. ■

■ 《実験 B》

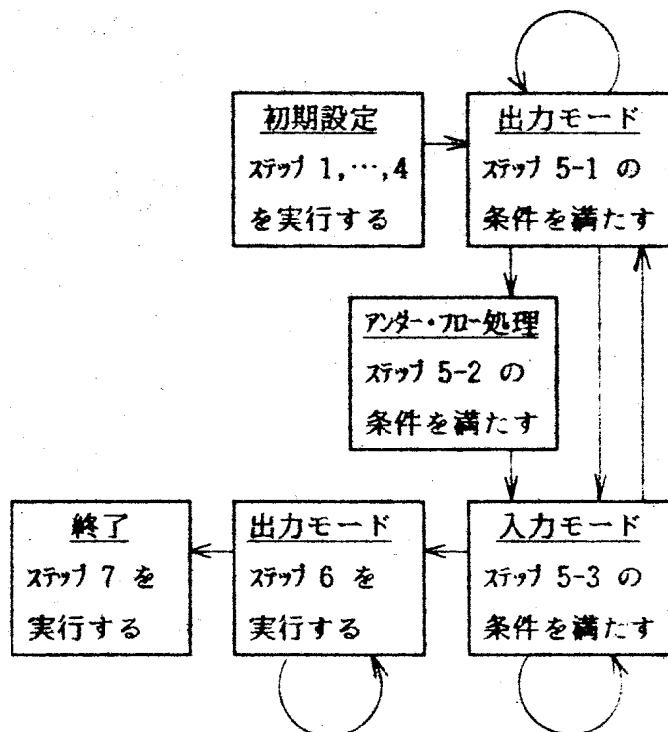
2-2節および2-3節の実験 B と同じ情報伝送率 1.7% の下で, 算術符号化法と共に暗号化法 3 を適用してみた. $\epsilon=0.01$.

図 B-0 は原画像, 図 B-3 は暗号化処理された画像である. ほとんど見分けがつかない. ■

第3-4節 算術符号化法の妥当性

前節の算術符号化法を用いて確かに希望通りの分布が達成できることを示そう。

符号化は、下図のような状態遷移に沿って進行する：



ある $\underline{x} \in \{0,1\}^k$ に対し、符号化操作を行ったとする。一度出力モードに入った後、そこを抜け出すまでに出力した過去の総文字数を $\xi(1)$ で表わす。続いて入力モードに入った後、そこを抜け出すまでに入力した過去の総文字数を $\xi(1)$ で表わす。再び出力モードに入った後、そこを抜け出すまでに出力した過去の総文字数を $\xi(2)$ で表わす。（すなわち、2回目に出力モードに入った時点から数えて、そこを抜け出すまでに出力した文字数は $\xi(2)-\xi(1)$ である。）同様にして、 $\xi(2), \xi(3), \dots, \xi(v), \xi(v+1)$ を定義する。ここで、 v は入力モードへの総入出回数を表わす。出力モードへの総入出回数は、ステップ 6 が寄与するため、入力モードのそれよりも一回多くなる。さらに、 $\xi(0)=\xi(0)=0$ と約束す

る。既出の記号との関わりから、 $\xi(v)=k$ 、かつ $\xi(v+1)$ は $f(\underline{x})$ の長さに等しい、という条件が自然に課せられる。

$p(0)=\epsilon, p(1)=\bar{\epsilon}$ と定義する。さらに、それぞれの $\underline{y}=y_1 \cdots y_i \in \{0,1\}^*$ に対し

$$p(\underline{y}) = p(y_1) \times \cdots \times p(y_i)$$

と定義する。

□【定理1】

2-2節の算術符号化法において、 $k=\infty$ と設定する。また、任意の正整数 d を固定する。 q と r の比を一定に保ちつつ q を増加させるとき、十分大きい i に対し

$$\Pr\{Y_{i+1} \cdots Y_{i+d} = \underline{y}\} \rightarrow p(\underline{y}), \quad \forall \underline{y} \in \{0,1\}^d$$

となる。 □

■《注意》 \rightarrow VF へ変更すると
問題なし

k を有限に設定すると、 $f(\underline{x}), \underline{x} \in \{0,1\}^k$ の長さがまちまちになり、その最小値・最大値間の i に対し “ Y_i の分布” の実際的な意味が不明瞭になる。 ■

■《定理1の証明》

ある $\underline{x} \in \{0,1\}^k$ に対し、符号化操作を行ったとする。系列 $f(\underline{x})$ の構成要素を $f^i(\underline{x}), i=1, 2, \dots$ で表わす。 $[a, b]$ を更新した各時点で、過去に入力、出力した総文字数がそれぞれ i, j のとき、 $[a, b]$ を $\mathcal{J}(i, j)$ で表わす。（アンダーフロー処理のときに限り、更新前の $[a, b]$ と更新後のそれとに同一の記法 $\mathcal{J}(i, j)$ が割り当てられることに注意せよ。）

入力モードにおいて、それぞれの (i, j) ：
 $i = 1, 2, \dots; j = \xi(i-1)+1, \dots, \xi(i)$

に対し

$$\frac{|\mathcal{J}(j, \xi(i))|}{|\mathcal{J}(j-1, \xi(i))|} = 2$$

が成立立つ。

同様に、出力モードにおいて、それぞれの

(i, j) :

$$i = 1, 2, \dots; j = \zeta(i-1)+1, \dots, \zeta(i)$$

に対し

$$\begin{aligned} & |\mathcal{S}(\zeta(i-1), j)| \\ & > |\mathcal{S}(\zeta(i-1), j-1)| p(y_j) - 2^{-q} \end{aligned}$$

すなわち

$$\begin{aligned} & \frac{|\mathcal{S}(\zeta(i-1), j)|}{|\mathcal{S}(\zeta(i-1), j-1)|} \\ & > p(f^j(\underline{x})) - \frac{|\mathcal{S}(\zeta(i-1), j-1)|}{2^{-q}} \\ & \geq p(f^j(\underline{x})) - 2^{-q+r} \\ & \geq p(f^j(\underline{x})) \times (1 - 2^{-q+r}/\epsilon) \end{aligned}$$

が成り立つ。

また、アンダー・フロー処理を一回行うたびに、 $|\mathcal{S}(\zeta(i-1), \zeta(i))|$ は

$$|\mathcal{S}(\zeta(i-1), \zeta(i))| / 2$$

以上の値へ書き換えられる。

これら三つの関係に基づいて、可能な $i, j, i''(>i), j''(>j)$ の組合せに対し

$$\begin{aligned} & 2^{-i''+i} \times |\mathcal{S}(i'', j'')| / |\mathcal{S}(i, j)| \\ & > p(f^{j+1}(\underline{x}) \cdots f^{j''}(\underline{x})) \\ & \quad \times (1 - 2^{-q+r}/\epsilon)^{j''-j} \times 2^{-\mu(j, j'')} \end{aligned}$$

を得る。ここで、 μ はアンダー・フロー処理の回数を表わす。したがって、任意の $y \in \{0, 1\}^{j''-j}$ に対し

$$\begin{aligned} & \Pr\{Y_{j+1} \cdots Y_{j''} = \underline{y}\} \\ & = \sum_{\substack{x: f^{j+1}(\underline{x}) \cdots f^{j''}(\underline{x}) = \underline{y}}} \Pr\{X^k = \underline{x}\} \times \Pr\{X_{j+1} \cdots \overset{\uparrow}{X_{j''}} \in \mathcal{S}(\zeta(i''), j'')\} \\ & & \quad \downarrow \quad \downarrow \quad \downarrow \\ & & \quad X_{j+1} \cdots X_{j''} \in \mathcal{S}(\zeta(i), j)\} \\ & > p(\underline{y}) \times (1 - 2^{-q+r}/\epsilon)^{j''-j} \\ & \quad \times \sum_{i=0, 1, \dots} \Pr\{\mu(j, j'') = i\} \\ & \quad \quad f^{j+1}(x^k) \cdots f^{j''}(x^k) = \underline{y}\} 2^{-i} \end{aligned}$$

となる。

一方、ほとんど同様なやり方で、任意の $y \in$

$\{0, 1\}^{j''-j}$ に対し

$$\begin{aligned} & \Pr\{Y_{j+1} \cdots Y_{j''} = \underline{y}\} \\ & < p(\underline{y}) \times (1 + 2^{-q+r}/\epsilon)^{j''-j} \end{aligned}$$

を導くことができる。(この場合、上界を押えることが目的なので、因子 $2^{-\mu(j, j'')}$ は不要である。)

結局、 $Y_{j+1} \cdots Y_{j''}$ の分布を上方と下方どちら押えることができた。したがって、 $j''-j=d$ なる j, j'' に対し、 q が増大するにつれ、 $\mu(j, j'')$ > 0 となる確率が 0 に近づくことを示せば、証明は完了する。ここでしばらくの間、定理の証明を中断し、そのようになることを確認しよう。

《証明一時中断》 ■

いま、

$$\begin{aligned} \gamma &= \min\{p(0), p(1)\} \times (1 - 2^{-q+r}/\epsilon) \\ &= \epsilon (1 - 2^{-q+r}/\epsilon) \end{aligned}$$

および

$$\begin{aligned} \kappa &= \max\{i: 2^{-r} \gamma^{-i} \leq 2^{-2}\} \\ &= \lfloor (r-2)/(\log \gamma) \rfloor \end{aligned}$$

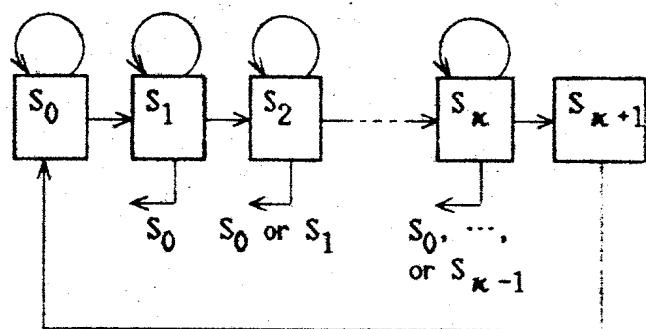
とおく。さらに、入力モードまたは出力モードにおいて $[a, b]$ が更新されるその都度 $|[a, b]|$ を観測することにより、次の $\kappa+2$ 個の状態を定義する: $i=0$ に対し

$$\begin{aligned} S_i &= \{2^{-r} \gamma^{-\kappa} \leq |[a, b]|, \\ & i=1, \dots, \kappa \text{ に対し} \\ S_i &= \{2^{-r} \gamma^{-\kappa-i} \leq |[a, b]| < 2^{-r} \gamma^{-\kappa-i-1}\}, \\ & i=\kappa+1 \text{ に対し} \end{aligned}$$

$$S_i = \{|[a, b]| < 2^{-r}\}.$$

状態遷移は、 X^k の構成要素を一個入力する、または Y^L の構成要素を一個出力する毎に起こるが、以下の特別な場合については約束を行う: 初期設定を終えた時点で、まず S_0 にいるものと考える。また、 $S_{\kappa+1}$ に移るということは、その後アンダー・フロー処理を行ってから入力モードに入り $[a, b]$ の大きさを 2^{-2} 以上まで回復するということを意味するので、引き続く時点では S_0 に移るものと考える。

状態遷移系は X^k によって駆動されるが、単に $S_i, i=0, \dots, \kappa+1$ 間の遷移と見なせば、下図のような 1 次マルコフ系をなす：



状態 S の定常分布を $P(S)$ と記す。また、現時点で状態 S にいるとき、次の時点で状態 S'' へ遷移する確率を $P(S \rightarrow S'')$ と記す。状態遷移確率に関し、少なくとも

$$P(S_i \rightarrow S_{i+1}) \leq \epsilon, \quad i=0, \dots, \kappa,$$

$$P(S_{\kappa+1} \rightarrow S_0) = 1$$

であることがわかっている。

□【補助定理2】

$|j''-j|$ を固定する。 $f(x)$ の構成要素を $j''-j$ 個出力する間に、状態 S_{k+1} を通過する確率は、 q/r を一定に保ちつつ q を増大させると 0 に近づく。 □

■《補助定理2の証明》

仮に、それぞれの $i=0, \dots, \kappa$ に対し

$$P(S_i \rightarrow S_{i+1}) = \epsilon,$$

$$P(S_i \rightarrow S_i) = \bar{\epsilon}$$

であり、かつ

$$P(S_{\kappa+1} \rightarrow S_0) = 1$$

であると仮定すると、

$$P(S_i) = 1/(\kappa + \epsilon), \quad i=0, \dots, \kappa,$$

を得る。それゆえ、実際には

$$P(S_i) \geq 1/(\kappa + \epsilon), \quad i=0, \dots, \kappa,$$

であることが知られる。

次に、 S_{k+1} から S_0 へのルートを切断し、 S_{k+1} を吸収状態と考える。 $f(x)$ の構成要素を

$j''-j$ 個出力する間の遷移を \leftrightarrow で表わすとき、

$$i+j''-j \leq \kappa$$

なる i に対し

$$P(S_i \leftrightarrow S_{\kappa+1}) = 0$$

である。

以上より、

$$P(\cdot \leftrightarrow S_{\kappa+1}) = \sum_{i=0}^{\kappa} P(S_i) P(S_i \leftrightarrow S_{\kappa+1})$$

$$\leq (j''-j)/(\kappa + \epsilon) = 0(q^{-1})$$

を得る。したがって、主張が成立する。 ■

□《定理1の証明の続き》

さて、 $\{0,1\}^{j''-j}$ の部分集合 S_q の系列
 $\{S_q\}, q=1, 2, \dots$

のうち、条件

$$\lim_{q \rightarrow \infty} \Pr\{Y_{j+1} \cdots Y_{j''} = \underline{y}\} > 0, \quad \forall \underline{y} \in S_q$$

かつ

$$\lim_{q \rightarrow \infty} \Pr\{Y_{j+1} \cdots Y_{j''} \in S_q\} = 1$$

を満たすものを考える。このとき、補助定理2に基づき、任意の $\underline{y} \in S_q$ に対し

$P(\cdot \leftrightarrow S_{k+1} | f^{j+1}(x^k) \cdots f^{j''}(x^k) = \underline{y}) \leq 0(q^{-1})$ でなければならない。これは、任意の $\underline{y} \in S_q$ に対し

$$\Pr\{\mu(j, j'') = 0 | f^{j+1}(x^k) \cdots f^{j''}(x^k) = \underline{y}\} \leq 0(q^{-1}),$$

$$|\Pr\{Y_{j+1} \cdots Y_{j''} = \underline{y}\} - p(\underline{y})| \leq 0(q^{-1}).$$

したがって

$$\Pr\{Y_{j+1} \cdots Y_{j''} \in S_q\} \rightarrow p(S_q)$$

であることを意味する。左辺は 1 に近づくので、 $p(S_q) \rightarrow 1$ でなければならない。すなわち、 q を十分大きく選ぶとき $S_q = \{0,1\}^{j''-j}$ でなければならない。結局、任意の $\underline{y} \in \{0,1\}^{j''-j}$ に対し

$$|\Pr\{Y_{j+1} \cdots Y_{j''} = \underline{y}\} - p(\underline{y})| \leq 0(q^{-1})$$

となることがわかる。したがって、主張が成立する。 ■

第4章. 応用

われわれが提唱した方法は、濃淡画像 (=白黒多値画像)、カラー画像、カラー濃淡画像、などを搅乱用情報とする方式へ拡張できる。先の二値出力の算術符号化法に修正を施し、多値出力の算術符号化法を構成することは容易である。問題は、画像中に情報源データをどのような分布で埋め込めばよいか、という点にある。

濃淡画像については、人間工学の知見より次のことがいえる：普通、輝度を 2^3 ないし 2^4 段階程度確保すると、画面を見たとき不便を感じなくなる、といわれている。これを逆の立場から解釈するならば、輝度が 2^4 段階かそれよりも少ないときには、むやみに輝度を揺らすべきではない。ということを示唆する。この場合は経験的に、暗号化法3のように、輝度の変化点付近へ、 \pm をなるべく小さく抑え、かつ、輝度がたかだか ± 1 の範囲で揺れるように情報を混入させるとよいようである。

輝度が 2^5 段階かそれよりも多いときは、それぞれの画素の輝度を適切な分散を持つガウス分布に従って揺らせば、特に輝度の変化点付近に限らず画面全体に情報を混入させても、不自然さは現れない。輝度が α 段階あるとき、ある画素の輝度の揺れが $\pm \alpha/2^4$ 以下である確率が、例えば 95% 以上となるように、標準偏差を

$$\sigma = \alpha/32$$

と選んだ場合、画質の劣化を知覚することは実際難しい。

カラー画像やカラー濃淡画像を搅乱用情報として用いる場合の視覚的効果は、未検討である。

- [1] W.Diffie and M.E.Hellman, "New Directions in Cryptography," IEEE, IT-22, pp.644-654, 1976.
- [2] 喜安善市 他, "暗号", 別冊数理科学, サイエンス社, 1982.
- [3] R.C.Pasco, "Source Coding Algorithms for Fast Data Compression," Ph.D.thesis, Dept.of Electrical Engineering, Stanford Univ., 1976.
- [4] J.J.Rissanen, "Generalized Kraft Inequality and Arithmetic Coding," IBM J. Res.Develop., vol.20, pp.198-203, 1976.
- [5] M.Guazzo, "A General Minimum-Redundancy Source-Coding Algorithm," IEEE IT-26, pp.15-25, 1980.
- [6] G.G.Langdon and J.Rissanen, "Compression of Black-White Images with Arithmetic Coding," IEEE COM-29, pp.858-867, 1981.
- [7] C.B.Jones, "An Efficient Coding System for Long Source Sequences," IEEE IT-27, pp.280-291, 1981.

1167. Other standard charts,
used for quality measurements.

tended for use
at for evaluation



图A-0

1167. Other standard charts,
used for quality measurements.

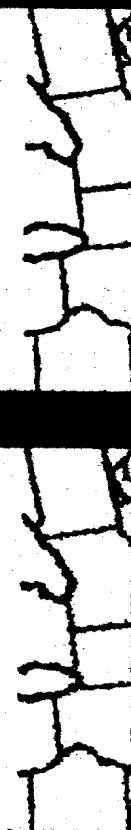
tended for use
at for evaluation



图A-1

1167. Other standard charts,
used for quality measurements.

tended for use
at for evaluation



图A-2

图A-3

1167. Other standard charts,
used for quality measurements.

tended for use
at for evaluation



图R-0

1167. Other standard charts,
used for quality measurements.

tended for use
at for evaluation



图B-2

1167. Other standard charts,
used for quality measurements.

tended for use
at for evaluation



图B-3