平成 13 年度

学士学位論文

日本語文書への秘匿情報埋め込み方式

A Digital Watermarking System onto Japanese Documents

1020281　　河村　智

指導教員　　情報システム工学科 清水　明宏

2001 年 2 月 8 日

高知工科大学 情報システム工学科

要 旨

日本語文書への秘匿情報埋め込み方式

河村　智

近年，コンピュータネットワークの発達とあらゆるコンテンツのディジタル化にともない，誰もが欲しい情報を迅速かつ容易に手に入れることが可能になった．しかし，それに伴い，不正なコピーによる著作権侵害の被害が深刻化してきている．この不正コピーを抑止するためには，模倣された作品が自分のものであることを示し，著作権の侵害を立証しなければならない．

そこで，本稿ではコンテンツ，特に日本語のテキストに透かしを埋め込む手法について提案する．本方式では助詞の同義語置換を行うことによってテキストにビットデータを埋め込む．まず，あらかじめ置換可能な表現をいくつも抽出しておき，さらにどちらがビット０でどちらがビット１に対応するのかを定め，それらを埋め込み規則としてデータベース化しておく．次に，埋め込み手順としては，まずキャリアとなるテキストを文法的に解析し，助詞部分を抜き出す．そして，抜き出した表現を先ほどの辞書データベース内のものと比較し，交換可能な個所があれば候補情報を付加する．次に埋め込みたい透かし情報を２進符号化し，先ほどの候補情報の中からビットデータにあったものを選択し置換を行う．これら一連の処理を繰り返すことによって透かし情報のビットデータが埋め込まれたテキストが完成する．さらに，これらの手順に従い日本語のテキストを解析して，予め用意した辞書データベースに基づいて候補情報を付加し置換を行うプログラムを試作し，本方式の実装を行った．本方式を新聞記事，雑誌のコラム，小説，日記に適用した結果，テキストの内容に左右されることなく，安定した埋め込みを行えることが確認できた．

キーワード　電子透かし，埋め込み，インフォメーションハイディング

Abstract

A Digital Watermarking System onto Japanese Documents

Satoshi KAWAMURA

In recent years, it enabled everyone to get information needed quikly and easily with a spread of a computer network, and digitization of all contents. Therefore, the damage of the literary piracy by the illegal copy is becoming serious simultaneously with it. It proves that the plagiarized work is its thing in order to deter an illegal copy. Furthermore, we must prove inforingement of copyright with it.

Then, I propose a system that embeds a digital watermark in Japanese text. In this system, bit data embedded by replacing the particle of the same meaning. First, much expression of the particle whose replacement is possible is prepared. And it is decided which corresponds to bit 0 in bit 1. And they are registered with a dictionary.

Next, a part of speech of the career text is analyzed and a particle is picked up. Then, the particle refers to the dictionary. And information of the candidate is given to expression of the particle whose replacement is possible. Next, information of the watermark is formed into bit data and the candidates is decided that is becomes same as bit data. Finally, the text in which bit data was embedded by these way. I created the program which embeds bit data in a Japanese text by these way and mounted this system.

key words Watermarking, Embbeding, Information Hiding

目次

第 1 章　はじめに　1

第 2 章　研究の背景　2

2.1　コンテンツの流通と不正コピー 2

2.2　不正コピーに対する試み 3

2.2.1　防止する技術 3

2.2.2　抑止する技術 4

2.3　コンテンツの二次配布の問題 5

2.4　テキストにおけるアナログコピー問題 6

2.5　テキスト配布における透かしの必要性 7

第 3 章　既存の透かし技術　8

3.1　視覚的な埋め込み法 8

3.2　行間制御方式 9

3.2.1　概要 9

3.2.2　評価 10

3.3　英文に対する語間制御方式 11

3.3.1　概要 11

3.3.2　評価 11

3.4　英文に対する改良型語間制御方式 13

3.4.1　概要 13

3.4.2　評価 14

3.5　和文に対する埋め込み方法 16

3.5.1　概要 17

3.5.2 評価 17
3.6 問題点 19

第 4 章 方式の提案 **20**
4.1 助詞置換埋め込み方式の提案 20
4.2 置換対象の選定の理由 21
4.3 埋め込み原理 23
4.4 埋め込み手順 25

第 5 章 方式の実装 **27**
5.1 プログラムの概要 27
5.2 プログラムの動作と外観 29
5.3 評価 31

第 6 章 むすび **33**

謝辞 **34**

図目次

3.1 英文行方向の黒画素ヒストグラムの例 9
3.2 語間制御方式の例 11
3.3 改良型語間制御方式の例 14
3.4 文字の回転と縮小 17
3.5 スキャナ入力画像の例 18

4.1 埋め込みの手順 26
4.2 候補情報の例 26

5.1 PreCheck フェーズ 1 30
5.2 PreCheck フェーズ 2 30
5.3 Check フェーズ 31
5.4 インタフェースの外観 32

表目次

第１章

はじめに

９０年代初頭にインターネットが登場して以来，コンピュータネットワークは驚異的な発展を遂げたきた．それまで，一対一のコンピュータ通信しか行えなかったものが，複数のコンピュータが同時にネットワークに接続しての通信が可能となったのである．そして，この情報通信における革命的とも言える変革を機に一般の家庭にもコンピュータの普及が進み，ますますネットワークの発達に拍車をかけた．

これと同時にメディアの世界も急速に変化していった．コンテンツはその必要や場合に応じて様々なメディアに記憶されていたが，ネットワーク社会の幕開けと共に急速にディジタル化されていった．それまであった音楽 CD などはもちろんのこと，テキスト，画像，動画，音声などあらゆるコンテンツが，それまでのペーパメディアやアナログ磁気テープメディアから，ディジタルメディアへと置き換わっていったのである．そして，このデジタル化されたコンテンツはネットワークを通じて世界中へと配信され，さらに電子的なライブラリや様々なデータベースによる検索により，誰もが居ながらにして求める情報を迅速かつ容易に手に入れることができるようになった．しかし現在，この手軽さが仇となり安易な不正コピーの問題が深刻化している．

そこで本稿ではユーザの安易な不正コピーを抑止するために，コンテンツ，特に日本語のテキストデータに対して，透かしとして何らかの情報を埋め込む方法を提案する．

第 2 章

研究の背景

2.1　コンテンツの流通と不正コピー

近年のコンピュータネットワークの普及とあらゆるメディア，ソフト，コンテンツのデジタル化によって，世界中のどこにあるデータでも手軽に手に入るようになった．デジタルコンテンツはその優れた品質の他に，何度コピーを繰り返しても品質が劣化しない，また取り扱いが簡単で労せず瞬時に複写が可能であり，しかも発達したネットワークによりその流布する速さと範囲は，これまでのアナログ的な伝達手段の比較にならないほど速く広範囲に及ぶ．このような特徴から，デジタルコンテンツを扱うビジネスはアイデア次第で無限の可能性を秘めていると言え，今後の成長が最も期待されている分野のひとつである．

しかしながら，デジタルコンテンツはコピーしても品質が劣化しないという特徴から，容易に不正コピーをされてしまうという問題点を抱えている．とくに昨今，電子的なライブラリや様々なデータベースによる情報の検索により，不特定多数の一般ユーザが貴重なデータにふれる機会が増えてきた．これに伴い，著作者の許可を得ない不正なコピーによる，著作権の侵害が問題化している．

2.2　不正コピーに対する試み

このような不正コピーに対して，これまでも様々な方法が試みられてきた．その多くはコピーそのもの防ぐことを目的とした「防止する」技術と，何らかの情報をコンテンツに埋め込む「抑止する」技術に分けられる．ここでは，この「防止する」技術と「抑止する」技術の双方の特徴とコンテンツ保護の観点から見た長所・短所について説明する．

2.2.1　防止する技術

これはコンテンツをコピーそのものを防ぐことを目的とした技術で，当然不正コピーだけでなく正規ユーザも含めたすべてのコピーに対して制限がかかる．この防止する技術の一つとしてコンテンツの “暗号化” があげられる．暗号化とは，第三者に盗み見られたり改竄されたりしないよう，ある規則にしたがってデータを変換する処理のことである．これにより配布の際には，一般にコンテンツを公開鍵暗号方式により暗号化し，正規のユーザのみが正当な手続きを経て鍵を入手して，復号しコンテンツの中身を見ることができる．暗号化したコンテンツはコピーしても暗号化されたままコピーされるので，配布途中で万一悪意のあるユーザに不正に取得されても，コンテンツの中身を見られることはない．

このようにコンテンツの暗号化は，復号されるまでの間は安全を保証することができるが，ひとたび復号されるとまったくの無防備となってしまうため，正規のユーザに復号された後のコンテンツの盗用や転用に対しては有効でないという問題点がある．

暗号化以外にコンテンツを配布する際に，利用者の認証を行う方法がある．この方法ではコンテンツの配布にあたり，その利用者が正当なユーザであるかどうか確認した上でコンテンツの配布を行う．この場合，先の暗号化と同様にコンテンツがユーザの手にわたるまでの部分については安全を保証するが，一度認証を終えてしまうとそれ以降の不正コピーには対応できないという問題点がある．

このように不正コピーを防止する技術は，通常，コンテンツの配布制御により一定期間の安全を保証する一時的かつ電子的な認証手法である．

2.2.2　抑止する技術

これはコンテンツのコピーそのものを防ぐことを目的としているのではなく，署名情報などをコンテンツに忍ばせることによって，それをユーザに意識させることによって不正コピーをしないよう促す技術である．

この抑止する技術の一つとして，“電子透かし” が挙げられる．電子透かしとは，ちょうど紙幣や有価証券に偽造防止用の “透かし” を入れるように，デジタルコンテンツに対してその冗長性を利用して，不正利用から保護する目的で人間に知覚できないように，著作権者を示す情報をそっと忍ばせる技術である．この埋め込まれた情報によって，万が一コンテンツが盗用されたり不正に転用されてたとしても，本当の著作者が誰なのか示すことができ，不正利用者による著作権の侵害を立証することができる．この盗用されたとしてもそれを立証できることによって，利用者の不正行為を抑止する効果がある．

現在，この電子透かし技術は静止画，動画，音声，テキストなど様々なコンテンツに適用が進んでいるが，埋め込まれる情報量を多くすればするほど，あるいは透かし情報が除去できないように埋め込み強度を強くすれば強くするほど，そのキャリアとなるデジタルコンテンツの品質の劣化を招くという問題点もある．

2.3　コンテンツの二次配布の問題

デジタルコンテンツの著作権保護のために様々な方法により対策が講じられてきたが，これらの対策を考えるに当たって無視できない重要な要素がコンテンツの二次的な配布，“二次配布の問題”である．コンテンツの二次配布とは，コンテンツ保護のための配布制御が終わったあとに行われる不正なコピーのことである．すなわち，コンテンツを暗号化する場合においてはユーザが正規の手続きにより鍵を入手し，コンテンツを復号した後に横流しをすることを指し，利用者の認証を行う場合においては，正規の利用者が認証手順を終えてコンテンツを入手したあとに同様に横流しすることを指す．

このような場合，先の，“不正コピーに対する試み”で説明したコンテンツの暗号化や利用者の認証による配布制御などの方法では防ぐことができない．したがって，コンテンツの二次配布の問題においては，コンテンツに対して透かしを用いて著作情報や署名を埋め込むことによってコピーを抑止する手法が有効である．

2.4　テキストにおけるアナログコピー問題

テキストデータにおけるアナログコピーの問題とはテキストデータをペーパメディアへ出力した場合や，またそれをコピー機などによってアナログコピーを行った場合の問題である．

ではまず，テキストデータの性質について考えてみる．画像データとテキストデータの価値について比較した場合，画像データはその外観・見た目による情報，例えば，青い空と建造物が写っている画像の場合，空が青いということと建造物がどんな形状をしているかということ，それと画像としての美しさ，すなわち視覚的な品質が重要である．しかし，それに対してテキストデータの場合は，もちろん文字ひとつひとつがクッキリと美しい方が読みやすいとはいえるが，それよりもその文字によって構成された文章が意味する内容が重要である．たとえ外観が劣化して文字が滲んでいたとしても,「こんにちは」と書かれた文がその通りに読み取れればコンテンツとして劣化していないことになる．したがって，画像データの場合はその “見た目の美しさ” に価値があり，テキストデータの場合は “意味する内容” に価値があることがわかる．

このことを踏まえて，先ほどのアナログコピーの問題を考えてみると，画像データの場合は一旦ペーパメディアに出力されたしまうと，以降はアナログコピーを繰り返す度に品質である “美しさ” は劣化して複写する前の原本よりも価値は目減りする．ところが，テキストデータの場合はたとえペーパメディアに出力されても，またその後にコピー機などによってアナログコピーを繰り返されたとしても，文字が判読できる限り品質である “内容” は劣化しない．したがって複写物は原本と同じ価値を持つことになる．

以上のことから，テキストデータにおける不正コピーの問題を考える場合には，ペーパメディアへの出力とその後のアナログコピーについても考慮する必要がある．

2.5　テキスト配布における透かしの必要性

ここまで，コンテンツの流通と不正コピーの現状，またそれに対する防止・抑止の試み，コンテンツの二次配布問題，テキストデータにおけるアナログコピーの問題について見てきた．これらのことを踏まえて，テキストデータにおける無許可の盗用・転用を防ぐための対策を考えると，テキストデータの性質上ペーパメディアへ出力した場合とその後のアナログコピーが行われた場合においても，不正コピーに対して有効なものでなくてはならないことがわかる．したがって，テキストデータの場合はコンテンツを暗号化するや利用者の認証を行う等の方法ではこれらの配布制御が行われるまでの期間は安全を保証することができるが，これらの処理が終わり，一度正規の方法でコンテンツが入手されたあとのコンテンツの盗用・転用は防ぎきれない．

それに対して，コンテンツに透かしを埋め込む方法ならば，たとえ正規のユーザからコンテンツが第三者に渡ったとしても，透かしが埋め込まれていればそれが誰のものなのかが証明でき，この透かしがペーパメディアに対しても有効なものであるならば，プリントアウト後のテキストやその後コピー機などにより複写が行われたとしても，それらの不正コピー対して作品の著作権を示すことができる．

したがって，テキストデータの場合は，デジタルデータからペーパメディアまでの広い範囲において，コンテンツを不正コピーから守るためには透かしを埋め込む方法が最も効果的であるといえる．なお，この透かしを埋め込む方式は，当然プリントアウトやコピー機による複写などを受けても消失しないものでなければならない．

第 3 章

既存の透かし技術

本章では，現在までに考案されてきたテキストに対する透かしの手法について，紹介する．なお，テキストに対する透かしの方法としては，大分して文字コード等に手を加え，デジタルデータ時にのみ有効である埋め込み方式と，視覚的に手を加えて，デジタルデータ時だけでなくペーパメディアへの出力後にも有効な埋め込み方式とに分けられるが，本稿では先にも述べたように，ペーパメディアへの出力後にも有効な埋め込み方式について提案するため，ここでは後者の視覚的に手を加えて埋め込む方式について紹介する．

3.1　視覚的な埋め込み法

テキストは画像や音声などの他のコンテンツに比べ，冗長性が低いことから透かしを埋め込むことが困難である．さらにペーパメディアへの出力後も有効な埋め込み方式となるとなお少なくなる．ここでは数少ないテキストに対する透かしの埋め込み方式として，代表的なものについて紹介する．本稿では日本語のテキストについて提案しているが，同じ視覚的に埋め込む方法の参考として，まず，英語テキストのような 単語単位が意味を持つ表音文字系のテキストに対する透かしを埋め込む方法について説明する．そして次に，日本語テキスト，和文にのような文字単位でそれぞれが意味を持つ，表意文字系のテキストに対する透かしを埋め込む方法について， 説明する．

3.2　行間制御方式

3.2.1　概要

まず，英語テキストの全体像を大きな視点で見てみると，行間隔の制御により透かしを埋め込む方法が考えられる．図 3.1 に英文のテキストを構成する黒成分の画素数を水平方向から見たヒストグラムを示す．グラフが高い値を示している部分が行で，低い部分が行間に当たる．この行間隔を上下に拡大または縮小することにより，透かしのビットを埋め込むことができる．すなわち，行間の標準画素数を基準にして，ビットが“0”ならば縮小し“1”ならば拡大する．その上下幅の移動量は，復号する際に使用されるスキャナの識別精度に依存するが，できるだけ少ない方が透かしが埋め込まれていることが露呈しにくく，望ましい．

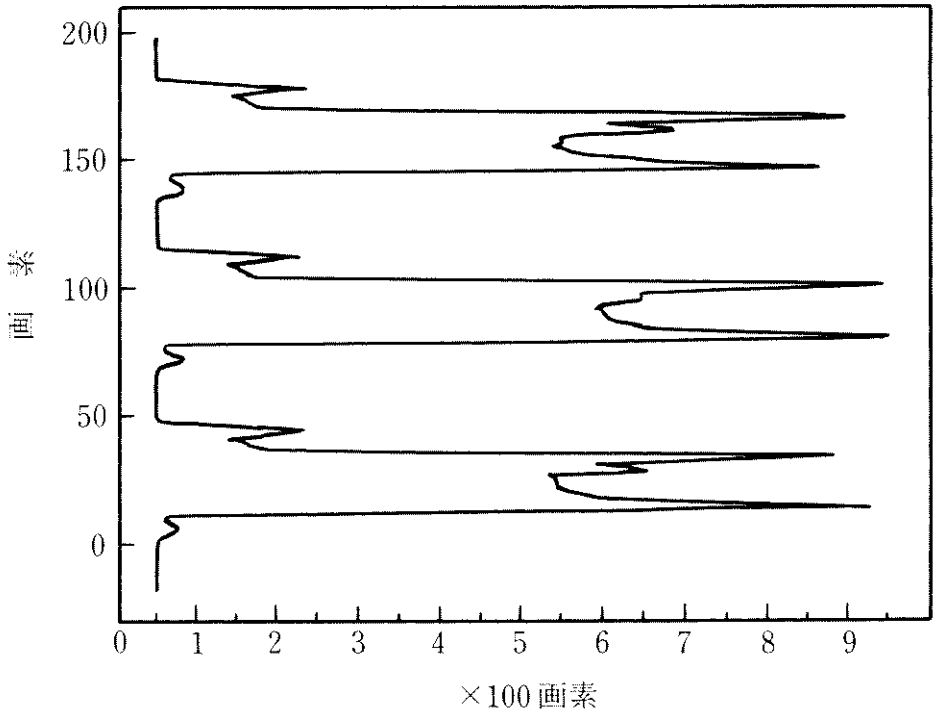


図 3.1　英文行方向の黒画素ヒストグラムの例

3.2.2　評価

この方法では，埋め込みの処理を 1 行単位で行うため，復号時における埋め込まれたビットの識別の信頼性は高くなるが，当然一つのページ内の行数に制約があるため，埋め込み可能な透かしのビット数が少なく，秘匿性に欠ける．また，行間の画素数を拡大または縮小することによって埋め込むことから，復号する際に使用されるスキャナの識別精度を考慮した上で，行間の移動量を決定する必要があり，この移動量があまりに大きくなると人の目にも明らかにその違いがわかることになり，埋め込み個所が露呈してしまい，また，小さすぎると移動量がわからなくなる，などの問題点がある．

3.3 英文に対する語間制御方式

3.3.1 概要

英語などの文章では，単語と単語の間に存在するスペースは区切り符合としての役割を果たしている．しかし，そのスペース長は 1 行に配置できる語数に依存しており，スペーシング機能を利用して美しく配置されるように長短さまざまである．そこで，このスペーシング機能を電子透かしに利用するようにしたものが，語間制御方式である．すなわち，図 3.1 に示すように，埋め込む透かしビットが “0” ならば単語 for の位置ををわずかに前方に移動させる．もし，埋め込む透かしビットが “1” ならばその単語をわずかに後方に移動させる．このようにして透かし情報を 1 行当たり数語程度の割合でランダムに選んでビットを埋め込み印刷出力する方法である．

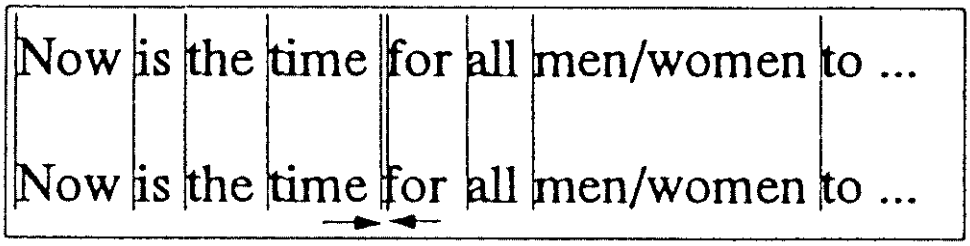


図 3.2 語間制御方式の例

3.3.2 評価

この方式では，復号の際にどの単語が移動しているのかを調べるのに，鍵としてのオリジナルのテキストとの照合が必要となるという問題点がある．しかも，オリジナルと透かしの埋め込んであるテキストとの照合をドット単位で実証しなければならない．これはセキュリティの強度としては原本さえ確実に保管されていればかなり強固なものとなるが，たいへん煩雑な作業である．

さらに，もう一つ問題点がある．英文のワードプロセッサの多くは 1 行の単語数および文

字数がその行にうまく収まらなかった場合，次の行に送ったり単語の間隔を詰めたりする自動機能を備えている．このため，デジタルコンテンツを一度読み込んで編集などしてペーパーメディアへと出力すると，照合の際に原本と一致しない全く別のスタイルのテキストになってしまうという点である．

3.4　英文に対する改良型語間制御方式

3.4.1　概要

先ほどの語間制御方式の問題点は原本との照合を必要とすることによるものであった．この問題点を解決するため，復号の際に原本との照合を必要としない方法として提案されたのが，改良型語間制御方式である．

この方法では，まず，テキスト行のある一つの単語に注目する．ただし，行の冒頭と行末の単語については行わないものとする．その注目した単語の前にあるスペース長を $\mathcal{P}$，その単語の後ろに引き続くスペース長を $\mathcal{S}$ とすると．このとき，$(\mathcal{P},\mathcal{S})$ の組合わせを一つの符合単位と考えて次の式を埋め込み規則として導入する．

埋め込む透かしビットが，“0”ならば，

$$\mathcal{P} \longleftarrow \frac{(1+\rho)(\mathcal{P}+\mathcal{S})}{2}$$
$$\mathcal{S} \longleftarrow \frac{(1-\rho)(\mathcal{P}+\mathcal{S})}{2}$$

とする．一方，透かしビットが，“1”ならば，

$$\mathcal{P} \longleftarrow \frac{(1-\rho)(\mathcal{P}+\mathcal{S})}{2}$$
$$\mathcal{S} \longleftarrow \frac{(1+\rho)(\mathcal{P}+\mathcal{S})}{2}$$

とする．ここで，式内のパラメータ ρ $(0<\rho<1)$ を偏移度とよび，これを透かしを埋め込むための鍵として利用する．この手順を英文に対して一語おきに繰り返して，透かしビットを埋め込んでいく．

また，この偏移度はスペースの増減を示す比であるが，復号時に用いられるスキャナの性能やテキストの予想されるコピー縮尺などを考慮して，それに耐える値に設定しなければならない．

Modern computer networks make it possible to distribute documents quickly and economically by electronic means rather than by conventional paper means. However, the wide spread adoption of electronic distribution of copyrighted material is currently impeded by the ease of illicit copying and dissemination. In this paper we propose techniques that discourage illicit distribution by embedding each document with a unique codeword. Our encoding techniques are indiscernible by readers, yet enable us to identify the sanctioned recipient of a document by examination of a recovered document. We propose three coding methods, describe one in detail, and present experimental results showing that our identification techniques are highly reliable, even after documents have been photocopied.

(a) 原 本

Modern computer networks make it possible to distribute documents quickly and economically by electronic means rather than by conventional paper means. However, the wide spread adoption of electronic distribution of copyrighted material is currently impeded by the ease of illicit copying and dissemination. In this paper we propose techniques that discourage illicit distribution by embedding each document with a unique codeword. Our encoding techniques are indiscernible by readers, yet enable us to identify the sanctioned recipient of a document by examination of a recovered document. We propose three coding methods, describe one in detail, and present experimental results showing that our identification techniques are highly reliable, even after documents have been photocopied.

(b) 順次埋込み

図 3.3 改良型語間制御方式の例

3.4.2 評価

この方式では，先の語間制御方式のように原本と照合する必要はないが，ペーパメディアへの出力後の複数回に及ぶ複写の繰り返すことによって透かし情報が劣化消失するという問題点が挙げられる．これは復号時に用いられるスキャナの読み取り精度と偏移度によって識別可能性が変化することによるもので，先ほどの ρ の値を調整して正しく復号できるようにしなければならない．また，コピーを繰り返すにつれて，行が歪み文字の形が劣化し，ノイズが増えてくる．これによって，単語間のスペース長の長短を正しく識別できなくなる．何世代まで透かし情報が劣化せずに残るべきかは，そのテキストの重要性によって変わってくる．

二つ目の問題点として，拡大コピーと縮小コピーによる透かし情報の劣化消失の問題がある．拡大コピーの場合は単語間スペースの比が変化する可能性は少ない上，絶対値が大きくなることから，読み取り精度の面から見ても都合が良く誤りが発生しにくいが，一方，縮小コピーの場合は，単語間スペースの絶対値が小さくなることから，その絶対値がスキャナの読み取り精度の限界に近づくにつれて，読み取りミスが多くなるという問題点がある．

三つ目の問題点として，埋め込み規則を知った第三者に容易に解読されてしまう点である．この方式では一文字ごとに情報が埋め込まれていることから，行頭から一文字ごとに前後のスペースを調べていけば埋め込まれたビットデータが容易にわかってしまう．

3.5　和文に対する埋め込み方法

ここまで，英文テキストにおけるいくつかの埋め込み方式について，紹介してきた．一般に英語のような表音文字系から構成される文章においては，各文字は単独では機能せず，いくつかかたまり単語となってはじめて一つの意味を持つ，構文となっている．そのため，単語と単語の間に区切り府としてスペースを入れるのである．英文においてはこの空白部分を増減させても全体の意味に何の支障も与えないことから，この部分に透かし情報を埋め込むことができた．

ところが，日本語のように表意文字から構成される文章においては，一文字一文字がそれぞれ単体で意味を持つことから，英文のように単語間に区切り府としてのスペースを入れる習慣がない．このような表意文字文の環境下においては，読者に細工を簡単に見破られてしまうことから，一つたりとも余分な空白もいれることはできないし，また仮に空白があってもそれらを伸長したり短縮したりすることはできない．

そこで，日本語テキストにおいてはこれまでとは違った方法で埋め込む必要がある．まず，テキストを読む立場から考えると，日本語テキストにおける各文字の意味をその形から把握していると思われる．したがって，文字の形から崩れていない限り，テキストを読む際に違和感を与えないものと推測できる．また，人が文章を書くとき，字画の多い漢字を自然と大きく書き，それに対して字画の少ないカナ文字を若干小さめに書くことが多い．これは人が文章を美しく記述しようと無意識のうちにとる行動である．このような日常的に使われている，技法を用いることによって，文字の幅をわずかに広げたり縮めたりしても，文章を読む者に違和感を与えることないと思われる．

3.5.1　概要

このような観点から，横書きスタイルの日本語テキストについて，文字を傾けたり，行方向にずらすなどして透かし情報の埋め込みを行う．まず，図 3.4 に示すようにテキスト中の特定の文字を角度 θ だけ傾けた場合，および行方向に λ 倍に縮小した場合について，テキストを読むときに気にならない程度に上限値 $(\theta\max, \lambda\max)$ を主観的評価によって求める．つぎに，これを実際に印刷してスキャナで読み取り，埋め込まれた署名情報を誤りなく復号できる下限値 $(\theta\min, \lambda\min)$ を求める．そして，この範囲内で透かし情報を埋め込むためのパラメータ (θ, λ) を決定する．このパラメータの上限値は主観的な評価であり，明確な基準はないが，下限値については原理的な変移量と復号の際のスキャナの解像度に依存する．

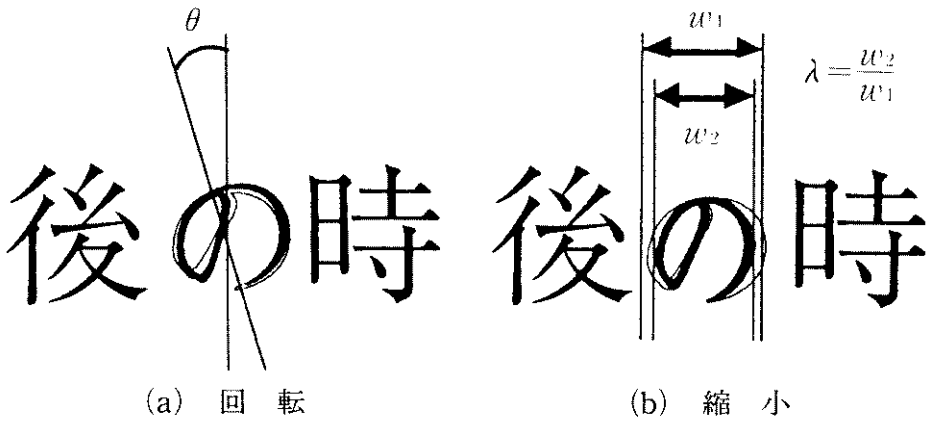
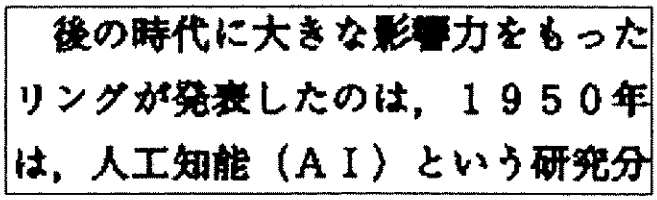


図 3.4　文字の回転と縮小

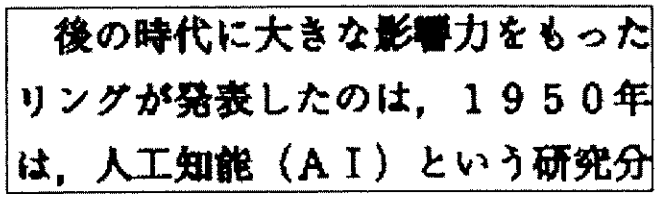
3.5.2　評価

この方式も先のものと同様に原本との照合を必要としないが，文字を回転させたり，縮小したりする場合の変移量が図 3.5 を見てもわかるとおり，1/100 インチ程度と非常に微細であるため，もちろん復号時のスキャナの性能にも因るが，ペーパメディアへ出力後に複数回の複写を受けると文章が劣化し，変移量がわからなくなってしまい，埋め込んだ透かし情報

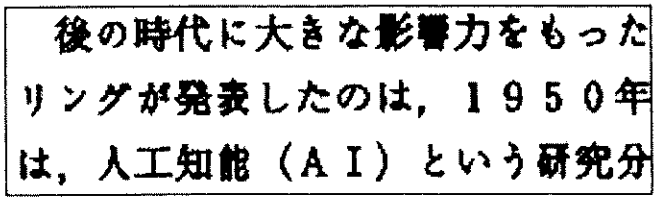
が消失してしまうという問題点がある．また，復号の精度はスキャナの性能に依存することから，劣化消失しないまでも変移量がスキャナの識別限界に近づくにつれて，読み出しミスが増えることになる．



(a)　原画像



(b)　回　転



(c)　縮　小

図 3.5　スキャナ入力画像の例

3.6　問題点

紹介した埋め込み方式のうち，まず日本語テキストに対して適用できる方法を考えてみる．一番最後に紹介した，日本語テキストを想定して提案された，文字の回転や縮小により透かし情報を埋め込む方式は，当然日本語テキストに適用できると考えると，その他の方式ではまず一つ目の行間制御方式は文章のなかの行間にのみ注目し，それを上下に移動させて埋め込むことから，日本語テキストにもそのまま適用できるものと考えられる．次に語間制御方式だが，この方法は表音文字系特有の単語間の区切符であるスペースに注目して埋め込みを行うものである．表意文字系である日本語には文章を書く際に区切符を用いる習慣はないため，この手法は明らかに日本語テキストには適用できない．同様の理由で改良型語間制御方式も日本語テキストには適用できないことがわかる．

次に，日本語テキストに対して適用できる方式についての問題点を考えてみる．行間制御方式の場合，全く同じ原理で日本語テキストにも適用可能であるが，１行毎にビットデータを埋め込むことから，１ページの行の数には限りがあり絶対的な埋め込みデータ量が少ないという問題点が挙げられる．さらに，埋め込み操作時の上下への移動量の問題がある．これは文字を回転・縮小させて埋め込む方式についてもいえることだが，これらの方法は行や文字を一見してわからない程度にわずかに動かすことによって，透かし情報のビットデータを埋め込んでいる．そのため，複数回のアナログコピーを繰り返されると，これらの変移量が劣化し消失してしまうという問題点がある．言い換えれば，これら既存の方式では複写物の文章を人が内容を読み取れるにも関わらず，埋め込まれた透かし情報が消えてしまっているという事態が起る，ということである．

第 4 章

方式の提案

前章であげた問題点を解決するために，本稿では文章内の交換可能な表現を置き換えることによって，透かし情報のビットデータを埋め込む方法を提案する．

4.1 助詞置換埋め込み方式の提案

本方式では，日本語テキストにおいて同じ意味の文でも複数の表現が可能である点に着目し，その複数の表現を置き換えることによって文章にビットデータを埋め込む．例をあげると，「友達と会うため駅に行った」というキャリア文があったとすると，格助詞の「と」と「に」の部分を置き換えることによって 4 通りの 2 進表現ができる．

友達 と 会うため駅 に 行った $\longrightarrow 00$

友達 と 会うため駅 へ 行った $\longrightarrow 01$

友達 に 会うため駅 に 行った $\longrightarrow 10$

友達 に 会うため駅 へ 行った $\longrightarrow 11$

このように，$00, 01, 10, 11$ の 4 通りのビット列を埋め込むことができる．本方式ではこの基本原理に基づいて，助詞に注目して交換可能な表現の置換を行い，それによってテキストにビットデータを埋め込んでいく．この方法であれば，先に述べたようなテキストの内容を読み取れるにも関わらず，埋め込まれた透かし情報が劣化消失するという問題は生じない．

4.2　置換対象の選定の理由

表現置換の基本原理に基づきビットデータを埋め込むに当たって，まず，その置換を行う対象を定めねばならない．本方式では助詞に注目してこれを置換対象として選定した．

置換対象として助詞を選定した理由としては，まず，一般的な日本語テキストを考えた場合，動詞・名詞・助詞が最も多く出現する．なかでも，動詞と名詞の組合わせが最も総出現数が多くこれを置換対象とすれば，より多くのデータを埋め込むことができる．しかし，ここで動詞と助詞を置換対象とした場合について考察する．先ほどの「意味が変わらない置き換え」という文を例に考えると,「置き換え」という動詞を「置換」という名詞に交換した場合,「意味が変わらない置換」となり，文が表す意味は同じであり読んだときの違和感もない．

しかし，この方法を実際のテキストに適用する場合を考えると，技術論文やマニュアルならばこれでも良いかもしれないが，一般の人が普通に書く文章や口語を書き下した文章の場合,「置換」という表現は通常用いない．同様に例文の「変わらない」という部分を「変化しない」に置換した場合，これも日常的な文章に適用すると必要以上に堅苦しい表現となり違和感がある．普通の人が病人を見舞いにいったとして，家に帰って家族にそのことを伝えたとすると,「昨日とそんなに 変わって なかったよ」とは言うかも知れないが,「昨日とそんなに 変化して なかったよ」とは言わない． このように動詞と名詞においては，状況や話者それにその内容によって意味上は交換可能だとしても，置換すると違和感のある文章となってしまう．

これに対して，助詞の場合は一部の終助詞や接続助詞などを除けば，基本的に文章の内容や話者の影響を受けずいつも同じものが使われる．先の例ならば「意味の変わらない置き換え」「意味が変わらない置き換え」というように，格助詞の「の」と「が」を交換したとしても文全体のニュアンスには何の影響も与えない．また,「置換」と「置き換え」を見てもわかるように，動詞・名詞の同意語で字数が同じものを見つけるのは極めて困難である．しかし，助詞においては，同じ種類の助詞であればほとんどの場合が同じ字数であることから，

置換を行った際にオリジナルのテキストに比べ，極端に文字数が増えたり，また減ったりすることが少ない．

これらのことから様々な分野のテキストに埋め込み方式を適用する場合を考えると，日本語テキスト中に出現する頻度，置換を行った際の文字数の変化の少なさ，置換を行った場合にニュアンスの変化を最小限に押さえることができるということから，助詞が最も適していると考え，本方式の置換対象として助詞を選定した．

4.3　埋め込み原理

4.1 助詞置換埋め込み方式でも述べたように，本方式では日本語テキストにおいて，交換可能な表現についてどちらがビット 0 でどちらがビット 1 かを定め，予め辞書データベースを作成しておき，この辞書に基づき助詞の置換を行って透かし情報を埋め込んでいく．

まず，すべての助詞のなかから意味が同じで表現の交換が可能な助詞とそのペアを探していく．例えば，格助詞の [と ⇔ に]，[の ⇔ が] などが挙げられる．文で見ると「友だちと/に会う」「私の/が書いた作文」といった場合である．このような交換しても文意が損なわれないものに関して，予め格助詞の「と」と「に」は交換可能で [と-0，に -1]，同じく格助詞の「の」と「が」は交換可能で [の-0，が-1] のように定めて辞書データベースに登録しておく，これを実際の埋め込み規則として表すと次のようになる．

「と」と「に」が交換可能で [と-0，に-1] という場合は，

$$\text{と} \longrightarrow \text{に} \ : \ \text{bit1}$$
$$\text{に} \longrightarrow \text{と} \ : \ \text{bit0}$$

同様に「の」と「が」は交換可能で [の-0，が-1] という場合は，

$$\text{の} \longrightarrow \text{が} \ : \ \text{bit1}$$
$$\text{が} \longrightarrow \text{の} \ : \ \text{bit0}$$

このように置換される方向とそれによって埋め込まれるビットについて定めておく．

置換可能な助詞の種類としては主に次のような 3 つのグループに分けられる．

- 通常同義置換グループ

  このグループは最も純粋に同義語置換に対応するもので，格助詞の「が」「に」「と」「へ」「を」「の」，副助詞の「のみ-だけ」「ばかり-ほど」係助詞の「は」，並列助詞の「と-や」「とか-やら」，副詞化の「に」「と」などがある．これらは助詞の種類は違うものもあるが，文法上同じ意味であることから，置換が可能なグループである．例を挙げると，格助詞の「と-に」の場合「友達と／に会う」,「に-へ」ならば「見えないところに／へ隠す」，係助詞の「は」と格助詞の「が」ならば「このことは／が裁判のときに決め手となる」，副詞化の「に-と」であれば「～することが可能に／となる」などである．

- 単純同義語置換グループ

  このグループは同じ種類の助詞動詞で若干表記が違うだけで，ほぼ同じもの同士による置換が可能な助詞のグループである．例を挙げると，格助詞連語の「～ にあたって-～ にあたり」「～ に対し-～ に対して」「～ によって-～ により」，接続助詞の「から-からには」「けれど-けれども」，副助詞の「くらい-ぐらい」「ばかり-ばっかり」などである．

- 省略可能グループ

  このグループは同義語置換ではなく，その語がなくても文章上意味が損なわれない場合に，その語がある・ないによって置換の代わりとなる助詞のグループである．例を挙げると，格助詞の「に」「～ の場合／に，中止とする」，接続助詞の「て」「明かりを消し／て，すぐに寝た」，連体化の「の」「談合など／の，不正行為があった場合～」などである．

これらのグループの特徴にしたがって置換を行うわけだが，これらの中で通常同義語置換グループの格助詞や副助詞などは他の助詞に比べて特に，一つ前にある名詞に合わせたものを選択しなければならないため，これらの置換には注意が必要である．

4.4　埋め込み手順

本方式においてテキストに対して，透かし情報を埋め込む際の埋め込み手順について説明する．図 4.1 を見ると，埋め込の手順としては大まかに 3 つのフェーズに分けられる．第一番目は解析・抽出フェーズ，第二番目は埋め込み個所抽出フェーズ，第三番目は情報埋め込みフェーズである．以下それぞれ順を追って説明していく．

1. 解析・抽出フェーズ

   このフェーズでは，まずはじめにオリジナルのテキストを文法的に読み取るために形態素解析をする．これによりテキストのなかのすべての単語について品詞を調べる．そして，その調べたものの内から助詞の部分だけを抽出し，次のフェーズへと進む．

2. 埋め込み可能個所抽出フェーズ

   このフェーズでは先ほど形態素解析と抽出処理によって取出した助詞の部分を，埋め込み規則として予め登録しておいた辞書データベースのなかにある置換可能な表現と照らし合わせ，テキストの中に置換可能な表現が含まれていた場合は，全体の文意を損なわないかどうか十分に注意して，置換しても問題なければ，置換可能箇所に図 4.2 のような候補情報を付加する．

3. 情報埋め込みフェーズ

   このフェーズで先ほど置換可能箇所に付加された，候補情報の中から候補を決定し，情報の埋め込を行う．ここで，透かし情報をテキストに埋め込むために 2 進符号化する．そして，この 2 進符号化したビットデータに合わせて，候補情報の中から候補を選んで決定する．これによって，透かし情報がテキストに埋め込まれる．

これらの一連の処理を終えると，透かし情報が埋め込まれたテキストが完成する．

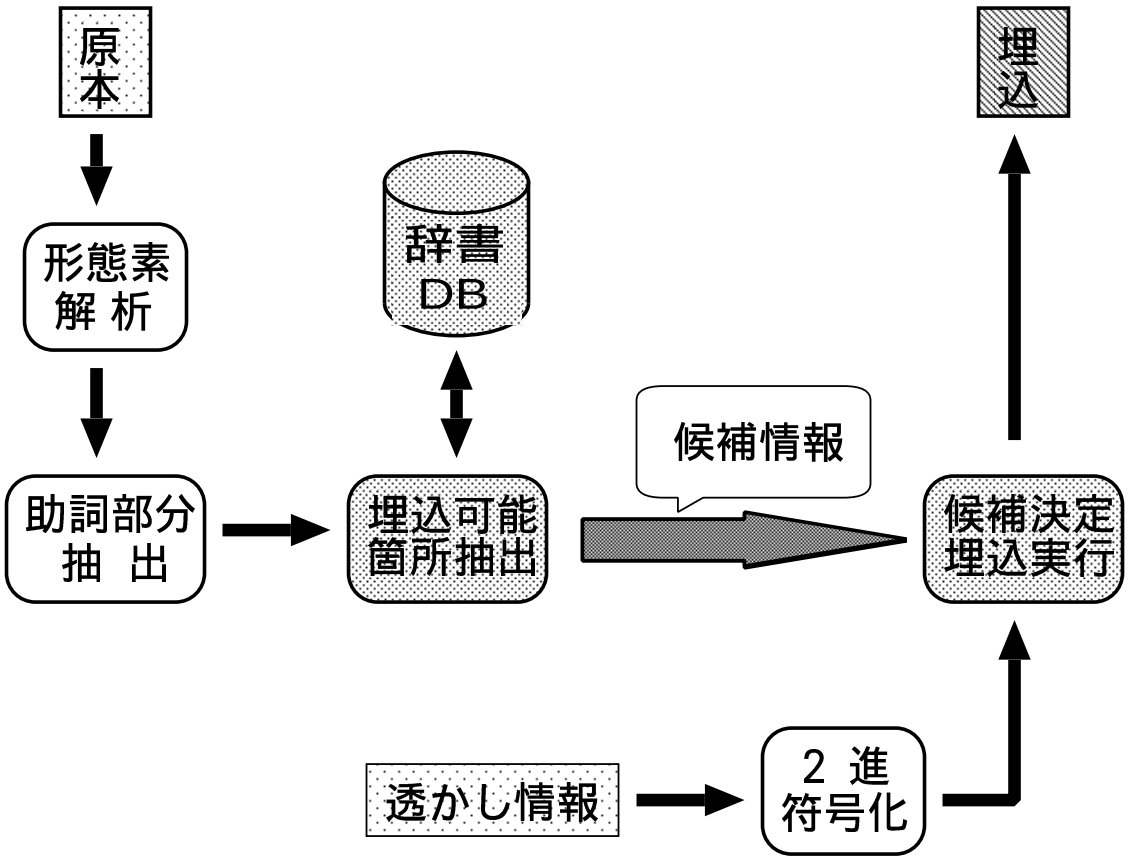


図 4.1　埋め込みの手順

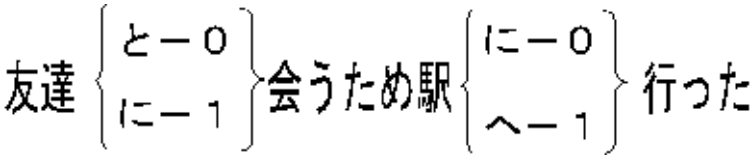


図 4.2　候補情報の例

第 5 章

方式の実装

前章で提案した助詞置換埋め込み方式の埋め込み原理と埋め込み手順に基づいて，日本語のテキストデータに対して透かし情報のビットデータを埋め込むプログラムを試作し，本方式の実装を行った．実装に当たっては JAVA 言語を使用し，日本語の文法的な解析には形態素解析プログラム Chasen を使用した．

5.1 プログラムの概要

4.3 の埋め込み規則と 4.4 の埋め込み手順に基づいて，日本語テキストに対して透かし情報のビットデータを埋め込むプログラムを作成した．

1. ファイルの読み込みと助詞の抽出フェーズ

   本プログラムの概要としては，まず，指定されたファイル名のテキストを読み出し，外部コマンドとして形態素解析プログラム Chasen を実行して，読み出したテキスト内のすべての単語の品詞・活用を調べ，その中から助詞の表現箇所だけをピックアップする．助詞以外の品詞の部分には何の処理も行わずそのままファイルへと出力する．

2. 埋め込み可能箇所の抽出フェーズ

   次にその調べた助詞の情報をもとに予め作成しておいた辞書と照らし合わせ，置換可能な表現がないかどうか，チェックする．置換可能な表現でない助詞に関してはその他の品詞のものと同様，何も処理を行わずファイルへと出力する．

3. 候補の決定とビットの埋め込みフェーズ

   辞書と照らし合わせた結果，置換可能な表現だった場合，辞書に登録された埋め込み規則にもとづいてビットデータに合わせた候補を決定する．この候補の決定に際しては最も有力な候補をユーザに提示して，それに問題がなければそのまま採用され，もし文意が損なわれるような問題があった場合は，他の正しい候補および「置換しない」という項目のいずれかをユーザに選択してもらい．その選択通りに処理を行う．

以上のような流れとなっている．埋め込むべきビットデータにしたがって置換候補を選択するようになっているものの，テキスト全体の意味を損ねないように置換を行うのはこんなんであり，最終的な決定にはユーザの手を借りることとなった．

5.2　プログラムの動作と外観

プログラムの動作についてそれぞれのフローチャートと共に示す．実装に当たり，まずテキストを形態素解析して助詞部分を抜き出し，辞書と比較して置換可能箇所の抽出を行う，5.1 プログラムの概要の 1. ファイルの読み込みと助詞の抽出フェーズ及び，2. 埋め込み可能箇所の抽出フェーズにあたる部分，を PreCheck フェーズとし，3. 候補の決定とビットの埋め込みフェーズに当たる部分を Check フェーズとして実装を行った．

- PreCheck フェーズ

  PreCheck フェーズでは指定されたテキストファイルを読み込んで Chasen を実行することによって品詞の解析を行うが，このとき，一度に全テキストを読み込んで解析を行うと，テキストファイルのサイズが大きくなった場合に危険であるため，テキストファイルを一行ずつ読み込んで解析し一次ファイル pretmp ファイルに結果を出力していく．この一次ファイルには CSV 形式で単語毎に品詞と活用が出力される．

  この処理が終わると，次にこの pretmp ファイルを一単語ずつ読み込んで，辞書データと比較していき交換可能な表現かどうか調べる．もし交換可能ならば順番に番号を振り印をつけ，二次ファイル tmp ファイルへ出力する．交換可能でない場合はそのまま何もせずに二次ファイルへ出力する．

  これらの処理が終わると，最後に置換可能な部分に印をつけた二次ファイルを表示する．

- Check フェーズ

  Check フェーズでは再び一次ファイル先頭から読み出して，交換可能な表現があった場合には PreCheck フェーズで置換可能箇所に付加した番号と埋め込まれるビットに合わせた交換可能な候補を表示し，ユーザに確認を促す．このとき OK ボタンが押された場合はそのまま置換を行い最終ファイル outfile に出力する．一方，問題がありユーザが他の候補や「置換しない」を選んだ場合にはその入力にしたがって置換を行い，最終

ファイルに出力する．

以上が本プログラムの動作である．以下にそれぞれのフェーズのフローチャートとインタフェースの外観を示す．

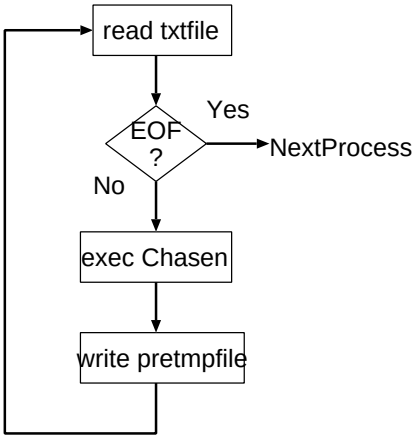


図 5.1　PreCheck フェーズ 1

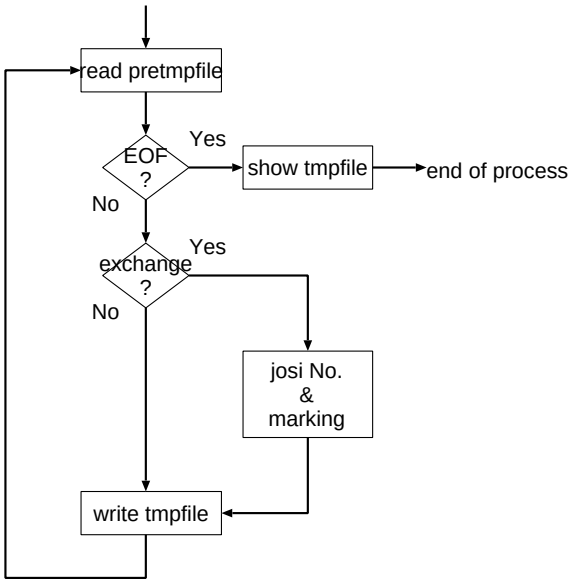


図 5.2　PreCheck フェーズ 2

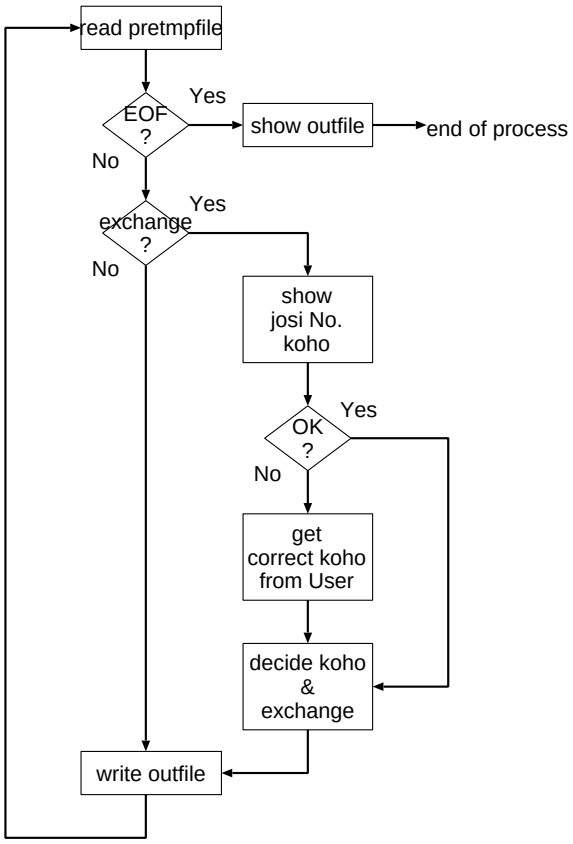


図 5.3　Check フェーズ

5.3　評価

本方式を実装したプログラムを新聞の記事，雑誌のコラム，小説，日記，それぞれ約１０００文字に適用して評価を行った．その結果，埋め込み可能なビット数は，もっとも多く埋め込めた小説では８２ビット，最も少なかった新聞の記事では６３ビットとなった．このことから本方式はテキストの内容によって極端に埋め込みビット数が増減することが確認できたが，適用したサンプル数が少ないためさらに多くのテキストに適用して検証を行わなければならない．

図 5.4　インタフェースの外観

第 6 章

むすび

日本語のテキストに対して，デジタルデータ時だけでなくペーパーメディアへの出力後も有効な，透かしの埋め込み方式について提案・実装した．しかしながら，方式の模索に多くの時間を費やしたために，当初の計画よりも実装が大幅に遅れ結果として十分な評価を行うことができなかった．今後，さらに多くのサンプルに適用して本方式の有効性を検証し，さらに，本稿では透かし情報の埋め込みのみの提案となったため，埋め込んだ透かし情報を取出す抽出方法の考案，さらに埋め込み規則を知ったうえで第三者が改竄を行った場合に，どの程度有効でるかの検証も行っていく予定である．

謝辞

研究全般にわたり，本学情報システム工学科　清水明宏教授には，御指導，御鞭撻を賜った．ここに謹んで深謝申し上げる．

また，本学情報システム工学科　妻鳥貴彦氏には，研究途上において有益な御助言をいただいた．ここに心から御礼申し上げる．

清水研究室の方々には，研究途上において有益な御議論をいただいたこと，心から感謝する．