

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2016-63475
(P2016-63475A)

(43) 公開日 平成28年4月25日(2016.4.25)

(51) Int. Cl.

H03M 7/40 (2006.01)

F I

H03M 7/40

テーマコード(参考)

5J064

審査請求 未請求 請求項の数 13 O L (全 23 頁)

(21) 出願番号 特願2014-191516(P2014-191516)
(22) 出願日 平成26年9月19日(2014.9.19)

(71) 出願人 000001443
カシオ計算機株式会社
東京都渋谷区本町1丁目6番2号
(74) 代理人 100095407
弁理士 木村 満
(72) 発明者 井手 博康
東京都羽村市栄町3丁目2番1号 カシオ
計算機株式会社 羽村技術センター内
Fターム(参考) 5J064 AA02 BA09 BB05 BC01 BC02

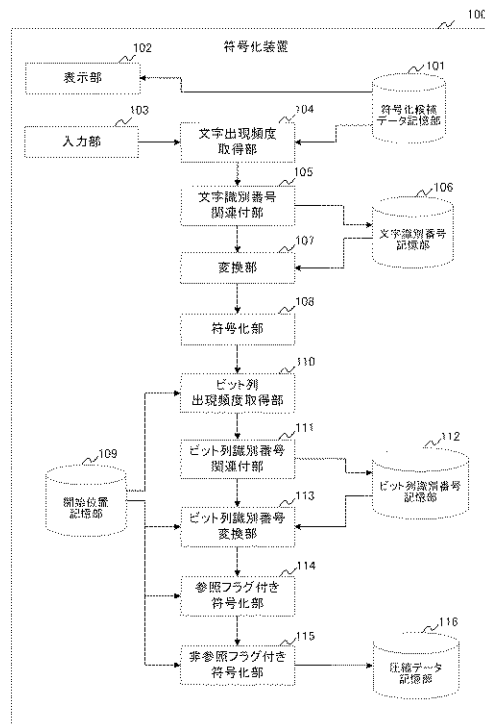
(54) 【発明の名称】 符号化装置、符号化方法、復号装置、復号方法、及び、プログラム

(57) 【要約】

【課題】 符号化効率が高い方法を用いて符号化対象のデータを部分的に復号可能なデータに符号化する符号化装置、符号化方法、符号化効率が高い方法を用いて符号化されたデータを部分的に復号する復号装置、復号方法、及び、プログラムを提供する。

【解決手段】 変換部107は符号化対象データに含まれる各文字を、文字を識別する文字識別番号に変換する。符号化部108は、各文字識別番号をバイナリデータに符号化する。ビット列識別番号関連付部111は、符号化対象のデータが符号化部108によって符号化されたデータに高頻度で出現する各ビット列を、ビット列識別番号に変換する。参照フラグ付き符号化部114は、各ビット列識別番号をバイナリデータに符号化する。

【選択図】 図2



【特許請求の範囲】**【請求項 1】**

符号化対象のデータに含まれる文字と、該文字を識別する文字識別番号と、を関連付けて記憶する文字識別番号記憶部と、

前記文字識別番号記憶部を参照して、前記符号化対象のデータに含まれる文字を、該文字に関連付けられた文字識別番号に変換する変換部と、

前記変換部が変換した文字識別番号をバイナリデータに符号化する符号化部と、

前記符号化部が符号化した文字識別番号のバイナリデータに出現する同一のビット列ごとに、該ビット列を識別するビット列識別番号を関連付けてビット列識別番号記憶部に記憶するビット列識別番号関連付部と、

前記ビット列識別番号記憶部を参照して、前記文字識別番号のバイナリデータに出現する同一のビット列を、該ビット列に関連付けられたビット列識別番号に変換するビット列識別番号変換部と、

前記ビット列識別番号変換部が変換したビット列識別番号に、復号時に前記ビット列識別番号記憶部を参照することを示す参照フラグを関連付けて、該ビット列識別番号をバイナリデータに符号化する参照フラグ付き符号化部と、

を備える符号化装置。

【請求項 2】

前記文字識別番号は、前記符号化対象のデータにおいて出現頻度の高い文字に関連付けられたものほど小さく、

前記符号化部は、前記文字識別番号が小さいほど該文字識別番号を符号化して得られるバイナリデータのデータ量が小さい符号化方法を用いて、前記文字識別番号をバイナリデータに符号化する、

請求項 1 に記載の符号化装置。

【請求項 3】

前記文字識別番号のバイナリデータに出現するビット列のうち前記ビット列識別番号が関連付けられていないビット列については、該ビット列の先頭から前記ビット列識別番号が関連付けられたビット列が出現するまでのビットの桁数に、復号時に前記ビット列識別番号記憶部を参照しないことを示す非参照フラグを関連付けて、該ビットの桁数をバイナリデータに符号化する非参照フラグ付き符号化部、

を備える請求項 1 又は 2 に記載の符号化装置。

【請求項 4】

前記ビット列識別番号は、前記文字識別番号のバイナリデータにおいて出現頻度の高い同一のビット列に関連付けられたものほど小さく、

前記参照フラグ付き符号化部は、前記ビット列識別番号が小さいほど該ビット列識別番号を符号化して得られるバイナリデータのデータ量が小さい符号化方法を用いて、前記ビット列識別番号をバイナリデータに符号化する、

請求項 1 乃至 3 の何れか一項に記載の符号化装置。

【請求項 5】

前記符号化対象のデータは、見出し語を含み、

前記同一のビット列の出現頻度は、前記符号化対象のデータにおける前記見出し語の開始位置を跨ぐ該同一のビット列の出現数を含まない、

請求項 4 に記載の符号化装置。

【請求項 6】

前記符号化対象のデータは、見出し語を含み、

前記ビット列識別番号変換部は、前記ビット列識別番号が関連付けられたビット列が、前記見出し語の開始位置を跨ぐ場合、該ビット列を前記ビット列識別番号に変換せず、

前記非参照フラグ付き符号化部は、前記ビット列識別番号が関連付けられたビット列のうち前記ビット列識別番号変換部が前記ビット列識別番号に変換しなかったビット列について、該ビット列の先頭から該ビット列が跨ぐ前記見出し語の開始位置までのビットの桁

10

20

30

40

50

数に、前記非参照フラグを関連付けて、該ビットの桁数をバイナリデータに符号化する、請求項 3 に記載の符号化装置。

【請求項 7】

前記非参照フラグ付き符号化部は、前記ビット列識別番号が関連付けられていないビット列が、前記見出し語の開始位置を跨ぐ場合、該ビット列の先頭から該開始位置までのビットの桁数に、前記非参照フラグを関連付けて、該ビットの桁数をバイナリデータに符号化する、

請求項 6 に記載の符号化装置。

【請求項 8】

前記ビット列識別番号記憶部が記憶する同一のビット列の桁数は、32桁である、

10

請求項 1 乃至 7 の何れか一項に記載の符号化装置。

【請求項 9】

符号化対象のデータに含まれる各文字を識別する各文字識別番号のバイナリデータと、前記各文字識別番号のバイナリデータに出現する同一のビット列を識別するビット列識別番号のバイナリデータと、で構成された圧縮データを記憶する圧縮データ記憶部と、

前記符号化対象のデータに含まれる文字と、該文字を識別する前記文字識別番号と、を関連付けて記憶する文字識別番号記憶部と、

前記各文字識別番号のバイナリデータに出現する前記同一のビット列ごとに、該ビット列を識別する前記ビット列識別番号を関連付けて記憶するビット列識別番号記憶部と、

復号対象のデータを特定するための条件を入力する入力部と、

20

前記圧縮データ記憶部が記憶する圧縮データのうち前記条件を満たすデータを復号対象のデータとして特定し、特定した復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号のバイナリデータとのうち、前記ビット列識別番号のバイナリデータを前記ビット列識別番号に復号するビット列識別番号復号部と、

前記ビット列識別番号記憶部を参照して、前記ビット列識別番号復号部が復号した前記ビット列識別番号を、該ビット列識別番号に関連付けられたビット列に変換するビット列識別番号変換部と、

前記復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号変換部が変換したビット列とを前記文字識別番号に復号する復号部と、

前記文字識別番号記憶部を参照して、前記復号部が復号した前記文字識別番号を、該文字識別番号に関連付けられた文字に変換する変換部と、

30

を備える復号装置。

【請求項 10】

符号化対象のデータに含まれる文字と、該文字を識別する文字識別番号と、を関連付けて記憶する文字識別番号記憶部を参照して、前記符号化対象のデータに含まれる文字を、該文字に関連付けられた文字識別番号に変換する変換ステップと、

前記変換ステップで変換した文字識別番号をバイナリデータに符号化する符号化ステップと、

前記符号化ステップで符号化した文字識別番号のバイナリデータに出現する同一のビット列ごとに、該ビット列を識別するビット列識別番号を関連付けてビット列識別番号記憶部に記憶するビット列識別番号関連付ステップと、

40

前記ビット列識別番号記憶部を参照して、前記文字識別番号のバイナリデータに出現する同一のビット列を、該ビット列に関連付けられたビット列識別番号に変換するビット列識別番号変換ステップと、

前記ビット列識別番号変換ステップで変換したビット列識別番号に、復号時に前記ビット列識別番号記憶部を参照することを示す参照フラグを関連付けて、該ビット列識別番号をバイナリデータに符号化する参照フラグ付き符号化ステップと、

を備える符号化方法。

【請求項 11】

復号対象のデータを特定するための条件を入力する入力ステップと、

50

符号化対象のデータに含まれる各文字を識別する各文字識別番号のバイナリデータと、前記各文字識別番号のバイナリデータに出現する同一のビット列を識別するビット列識別番号のバイナリデータと、で構成された圧縮データのうち、前記条件を満たすデータを復号対象のデータとして特定する特定ステップと、

前記特定ステップで特定した前記復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号のバイナリデータとのうち、前記ビット列識別番号のバイナリデータを前記ビット列識別番号に復号するビット列識別番号復号ステップと、

前記各文字識別番号のバイナリデータに出現する前記同一のビット列ごとに、該ビット列を識別する前記ビット列識別番号を関連付けて記憶するビット列識別番号記憶部を参照して、前記ビット列識別番号復号ステップで復号した前記ビット列識別番号を、該ビット列識別番号に関連付けられたビット列に変換するビット列識別番号変換ステップと、

前記復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号変換ステップで変換したビット列とを前記文字識別番号に復号する復号ステップと

、
前記符号化対象のデータに含まれる文字と、該文字を識別する前記文字識別番号と、を関連付けて記憶する文字識別番号記憶部を参照して、前記復号ステップで復号した前記文字識別番号を、該文字識別番号に関連付けられた文字に変換する変換ステップと、

を備える復号方法。

【請求項 1 2】

コンピュータを、

符号化対象のデータに含まれる文字と、該文字を識別する文字識別番号と、を関連付けて記憶する文字識別番号記憶部を参照して、前記符号化対象のデータに含まれる文字を、該文字に関連付けられた文字識別番号に変換する変換部、

前記変換部が文字識別番号をバイナリデータに符号化する符号化部、

前記符号化部が符号化した文字識別番号のバイナリデータに出現する同一のビット列ごとに、該ビット列を識別するビット列識別番号を関連付けてビット列識別番号記憶部に記憶するビット列識別番号関連付部、

前記ビット列識別番号記憶部を参照して、前記文字識別番号のバイナリデータに出現する同一のビット列を、該ビット列に関連付けられたビット列識別番号に変換するビット列識別番号変換部、

前記ビット列識別番号変換部が変換したビット列識別番号に、復号時に前記ビット列識別番号記憶部を参照することを示す参照フラグを関連付けて、該ビット列識別番号をバイナリデータに符号化する参照フラグ付き符号化部、

として機能させるためのプログラム。

【請求項 1 3】

コンピュータを、

復号対象のデータを特定するための条件を入力する入力部、

符号化対象のデータに含まれる各文字を識別する各文字識別番号のバイナリデータと、前記各文字識別番号のバイナリデータに出現する同一のビット列を識別するビット列識別番号のバイナリデータと、で構成された圧縮データのうち、前記条件を満たすデータを復号対象のデータとして特定する特定部、

前記特定部が特定した前記復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号のバイナリデータとのうち、前記ビット列識別番号のバイナリデータを前記ビット列識別番号に復号するビット列識別番号復号部、

前記各文字識別番号のバイナリデータに出現する前記同一のビット列ごとに、該ビット列を識別する前記ビット列識別番号を関連付けて記憶するビット列識別番号記憶部を参照して、前記ビット列識別番号復号部が復号した前記ビット列識別番号を、該ビット列識別番号に関連付けられたビット列に変換するビット列識別番号変換部、

前記復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号変換部が変換したビット列とを前記文字識別番号に復号する復号部、

10

20

30

40

50

前記符号化対象のデータに含まれる文字と、該文字を識別する前記文字識別番号と、を関連付けて記憶する文字識別番号記憶部を参照して、前記復号部が復号した前記文字識別番号を、該文字識別番号に関連付けられた文字に変換する変換部、
として機能させるためのプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、符号化装置、符号化方法、復号装置、復号方法、及び、プログラムに関する。

【背景技術】

10

【0002】

データ量を削減するために、符号化対象のデータを符号化する（圧縮する）符号化方法、及び、復号対象のデータを符号化前の元データに復号する復号方法が知られている。

【0003】

例えば、非特許文献1は、符号化対象のデータに繰り返し出現する文字列について、2回目以降に出現する文字列を、最初に出現する文字列のデータ内における位置及び文字列の長さを示すデータ（以下、頻出文字列メタデータと言う。）に変換して符号化対象のデータを符号化するLZ符号化方法を開示している。また、非特許文献1は、復号対象のデータ（LZ符号化方法で符号化されたデータ）に含まれる複数の頻出文字列メタデータのそれぞれを、符号化前の元データにおいて繰り返し出現する文字列のうち最初に出現する文字列に置き換えて復号対象のデータを復号する復号方法を開示している。

20

【先行技術文献】

【非特許文献】

【0004】

【非特許文献1】岡野原大輔著 「高速文字列解析の世界」岩波出版、2012年

【発明の概要】

【発明が解決しようとする課題】

【0005】

しかしながら、従来の符号化方法・復号方法では、部分的な復号が必要とされるデータを符号化・復号の対象とした場合、符号化効率が低い。例えば、辞書データの復号では、ユーザが検索対象とした見出し語単位での復号が必要とされるが、このような部分的な復号を従来の符号化方法・復号方法が実現しようとする、辞書データを見出し語単位で符号化しなければならない。このことを、上述のLZ符号化方法を例にとって具体的に説明すると、辞書データ全体がLZ符号化方法で符号化されている場合、符号化された辞書データの一部分には、前述の頻出文字列メタデータの置き換えに必要となる、符号化前の元データに繰り返し出現する文字列のうち最初に出現する文字列が含まれない可能性が高い。そのため、辞書データ全体がLZ符号化方法で符号化されている場合、辞書データの部分的な復号はほぼ不可能である。従って、見出し語単位での辞書データの復号には、見出し語単位での符号化が必要となる。また、他の符号化方法・復号方法についても同様の理由から、復号対象のデータの部分的な復号には、符号化前の元データについて部分的な符号化が必要となる。そして、このような部分的な符号化は、符号化効率を著しく低下させる。

30

【0006】

本発明は、以上のような課題を解決するためのものであり、符号化効率が高い方法を用いて符号化対象のデータを部分的に復号可能なデータに符号化する符号化装置、符号化方法、符号化効率が高い方法を用いて符号化されたデータを部分的に復号する復号装置、復号方法、及び、プログラムを提供することを目的とする。

【課題を解決するための手段】

【0007】

上記目的を達成するために、本発明の第1の観点に係る符号化装置は、

50

符号化対象のデータに含まれる文字と、該文字を識別する文字識別番号と、を関連付けて記憶する文字識別番号記憶部と、

前記文字識別番号記憶部を参照して、前記符号化対象のデータに含まれる文字を、該文字に関連付けられた文字識別番号に変換する変換部と、

前記変換部が変換した文字識別番号をバイナリデータに符号化する符号化部と、

前記符号化部が符号化した文字識別番号のバイナリデータに出現する同一のビット列ごとに、該ビット列を識別するビット列識別番号を関連付けてビット列識別番号記憶部に記憶するビット列識別番号関連付部と、

前記ビット列識別番号記憶部を参照して、前記文字識別番号のバイナリデータに出現する同一のビット列を、該ビット列に関連付けられたビット列識別番号に変換するビット列識別番号変換部と、

前記ビット列識別番号変換部が変換したビット列識別番号に、復号時に前記ビット列識別番号記憶部を参照することを示す参照フラグを関連付けて、該ビット列識別番号をバイナリデータに符号化する参照フラグ付き符号化部と、

を備える。

【 0 0 0 8 】

上記目的を達成するために、本発明の第 2 の観点に係る復号装置は、

符号化対象のデータに含まれる各文字を識別する各文字識別番号のバイナリデータと、前記各文字識別番号のバイナリデータに出現する同一のビット列を識別するビット列識別番号のバイナリデータと、で構成された圧縮データを記憶する圧縮データ記憶部と、

前記符号化対象のデータに含まれる文字と、該文字を識別する前記文字識別番号と、を関連付けて記憶する文字識別番号記憶部と、

前記各文字識別番号のバイナリデータに出現する前記同一のビット列ごとに、該ビット列を識別する前記ビット列識別番号を関連付けて記憶するビット列識別番号記憶部と、

復号対象のデータを特定するための条件を入力する入力部と、

前記圧縮データ記憶部が記憶する圧縮データのうち前記条件を満たすデータを復号対象のデータとして特定し、特定した復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号のバイナリデータとのうち、前記ビット列識別番号のバイナリデータを前記ビット列識別番号に復号するビット列識別番号復号部と、

前記ビット列識別番号記憶部を参照して、前記ビット列識別番号復号部が復号した前記ビット列識別番号を、該ビット列識別番号に関連付けられたビット列に変換するビット列識別番号変換部と、

前記復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号変換部が変換したビット列とを前記文字識別番号に復号する復号部と、

前記文字識別番号記憶部を参照して、前記復号部が復号した前記文字識別番号を、該文字識別番号に関連付けられた文字に変換する変換部と、

を備える。

【 発明の効果 】

【 0 0 0 9 】

本発明によれば、符号化効率が高い方法を用いて符号化対象のデータを部分的に復号可能なデータに符号化することができる。また、符号化効率が高い方法を用いて符号化されたデータを部分的に復号することができる。

【 図面の簡単な説明 】

【 0 0 1 0 】

【 図 1 】 本発明の実施形態に係る符号化装置の物理構成を示す図である。

【 図 2 】 本発明の実施形態に係る符号化装置の機能構成を示す図である。

【 図 3 】 本発明の実施形態に係る符号化処理の流れを示すフローチャートである。

【 図 4 】 辞書データ（符号化対象のデータ）を示す図である。

【 図 5 】 各文字の辞書データにおける出現頻度を示す図である。

【 図 6 】 文字と文字識別番号との関連付けを示す図である。

10

20

30

40

50

【図 7】最終符号化前バイナリデータを示す図である。

【図 8】各ビット列の最終符号化前バイナリデータにおける出現頻度を示す図である。

【図 9】ビット列とビット列識別番号との関連付けを示す図である。

【図 10】本発明の実施形態に係る復号装置の物理構成を示す図である。

【図 11】本発明の実施形態に係る復号装置の機能構成を示す図である。

【図 12】本発明の実施形態に係る復号処理の流れを示すフローチャートである。

【発明を実施するための形態】

【0011】

以下、本発明の実施形態について、図面を参照して説明する。

【0012】

従来の符号化方法・復号方法によれば、データを部分的に復号したい場合は、符号化前の元データを部分的に符号化しておかなければならない。従って、符号化に手間がかかり、符号化効率は低い。この点に関して、本願発明に係る符号化装置 100、復号装置 200 は、符号化前の元データをまとめて符号化してしまっても部分的な復号を可能とする。以下、このような符号化装置 100、復号装置 200 の物理構成及び機能構成を説明する。

10

【0013】

本実施形態に係る符号化装置 100 は、物理的には図 1 に示すように構成される。即ち、符号化装置 100 は、ROM (Read Only Memory) 10 と、RAM (Random Access Memory) 11 と、外部記憶装置 12 と、入力装置 13 と、表示装置 14 と、CPU (Central Processing Unit) 15 と、を備える。

20

【0014】

ROM 10 は、各種初期設定、ハードウェアの検査、プログラムのロード等を行うための初期プログラムを記憶する。RAM 11 は、CPU 15 が実行する各種ソフトウェアプログラム、これらのソフトウェアプログラムの実行に必要なデータ等を一時的に記憶する。

【0015】

外部記憶装置 12 は、例えば、ハードディスクであって、各種ソフトウェアプログラム、データ等を記憶する。これらソフトウェアプログラムの中には、アプリケーションソフトウェアプログラム、OS (Operating System) のような基本ソフトウェアプログラム等が含まれている。

30

【0016】

入力装置 13 は、キーボード、マウス、トラックパッド等を備え、ユーザからの入力を受け付ける。入力装置 13 は、キーボード、マウス、トラックパッド等からの入力に基づいて信号を生成し、CPU 15 に供給する。

【0017】

表示装置 14 は、液晶ディスプレイ等の画面を備え、CPU 15 から供給されたテキストデータや画像データを表示する。

【0018】

CPU 15 は、外部記憶装置 12 が記憶するソフトウェアプログラムを RAM 11 に読み出して、そのソフトウェアプログラムを実行制御することにより、以下の機能構成を実現する。

40

【0019】

符号化装置 100 は、機能的には図 2 に示すように構成される。即ち、符号化装置 100 は、符号化候補データ記憶部 101 と、表示部 102 と、入力部 103 と、文字出現頻度取得部 104 と、文字識別番号関連付部 105 と、文字識別番号記憶部 106 と、変換部 107 と、符号化部 108 と、開始位置記憶部 109 と、ビット列出現頻度取得部 110 と、ビット列識別番号関連付部 111 と、ビット列識別番号記憶部 112 と、ビット列識別番号変換部 113 と、参照フラグ付き符号化部 114 と、非参照フラグ付き符号化部 115 と、圧縮データ記憶部 116 と、を備える。符号化候補データ記憶部 101 と、文

50

字識別番号記憶部 106 と、開始位置記憶部 109 と、ビット列識別番号記憶部 112 と、圧縮データ記憶部 116 と、は図 1 に示す外部記憶装置 12 に構築されている。

【0020】

符号化候補データ記憶部 101 は、複数の符号化候補のデータを記憶する。符号化候補のデータは、辞書データ等のテキストデータである。

【0021】

表示部 102 は、符号化候補のデータを記録したファイルの名称（ファイル名）を表示装置 14 に表示する。ユーザは、表示装置 14 に表示されたファイル名を参照して、符号化候補のデータのいずれを符号化するか指定する。

【0022】

入力部 103 は、入力装置 13 が生成した信号を受け付ける。この信号は、ユーザが入力装置 13 を介して指定したファイル名を示す。入力部 103 は、入力装置 13 から受け付けた信号に基づいて、符号化対象のデータを特定する。そして、符号化対象のデータを符号化する旨の指示を文字出現頻度取得部 104 に入力する。

【0023】

文字出現頻度取得部 104 は、入力部 103 から入力された指示に基づいて、符号化候補データ記憶部 101 が記憶する符号化候補のデータの中から符号化対象のデータを取得する。

【0024】

文字出現頻度取得部 104 は、符号化対象のデータに含まれる文字ごとに、符号化対象のデータにおける出現頻度を取得する。例えば、符号化対象のデータが図 4 に示す辞書データ 1 の場合、各文字の辞書データ 1 における出現頻度は図 5 のようになる。

なお、出現頻度の取得は、符号化対象のデータに含まれる全ての文字を対象に行われる。

【0025】

文字識別番号関連付部 105 は、符号化対象のデータに含まれる文字ごとに、文字を識別する文字識別番号を関連付けて文字識別番号記憶部 106 に記憶する。このとき、文字識別番号関連付部 105 は、符号化対象のデータにおいて出現頻度が高い文字から順に、値が小さい文字識別番号を関連付ける。例えば、符号化対象のデータが図 4 に示す辞書データ 1 の場合、図 6 に示すように、出現頻度が 1 番目、2 番目、3 番目・・・に高い文字「t」、「h」、「e」・・・に、それぞれ文字識別番号 0、1、2・・・を関連付ける。

【0026】

以下、出現頻度に基づいて文字に文字識別番号を関連付ける理由と、出現頻度が高い文字に値が小さい文字識別番号を関連付ける理由を説明する。

テキストの圧縮では文字コードの体系が圧縮率低下の原因となることがある。特に、UTF-8 を採用すると、多言語のテキストを圧縮対象とするため、言語によってはコード長が冗長になり、圧縮率が低下する。そこで、本願発明は、文字を文字コードに符号化するのではなく、文字に文字識別番号を関連付け、文字を文字識別番号のバイナリデータに符号化する。これで圧縮率は文字コードの体系に影響を受けずにすむ。また、頻出文字をコード長が長いバイナリデータに符号化すると圧縮データのサイズが大きくなるので、本願発明はそうならないよう工夫をしている。具体的には、上述したように頻出文字に値が小さい文字識別番号を関連付ける。そして、デルタ符号化方法、Variable Byte Code 符号化方法、ハフマン符号化方法といった値が小さいほどコード長が短いバイナリデータに符号化できる符号化方法を用いて、頻出文字をコード長が短いバイナリデータに符号化する。これにより、文字コードに符号化した場合に比べて圧縮データのサイズは大幅に小さくなる。

【0027】

文字識別番号記憶部 106 は、図 6 に示すように、符号化対象のデータに含まれる文字と、文字識別番号と、を関連付けて記憶する。

10

20

30

40

50

【 0 0 2 8 】

変換部 1 0 7 は、文字識別番号記憶部 1 0 6 を参照して、符号化対象のデータに含まれる文字を、文字に関連付けられた文字識別番号に変換する。

【 0 0 2 9 】

符号化部 1 0 8 は、変換部 1 0 7 が変換した文字識別番号をバイナリデータに符号化する。なお、本明細書では符号化部 1 0 8 が符号化したバイナリデータを最終符号化前バイナリデータと言う。図 7 は、図 4 に示す辞書データ 1 が変換部 1 0 7 により符号化された場合の最終符号化前バイナリデータ 2 を示す。なお、最終符号化前バイナリデータ 2 は実際には 0 又は 1 を表すビットの集合である。しかし、説明の都合上、8 ビットごとに 1 6 進数表記していることに留意されたい。例えば、図 7 に示すビット列「1 2 A 5 B 8 C A」は、「0 0 0 1 0 0 1 0」、「1 0 1 0 0 1 0 1」、「1 0 1 1 1 0 0 0」、「1 1 0 0 1 0 1 0」を、それぞれ「1 2」、「A 5」、「B 8」、「C A」というように 1 6 進数表記したものである

10

【 0 0 3 0 】

開始位置記憶部 1 0 9 は、符号化対象のデータに含まれる見出し語の開始位置（本明細書において圧縮境界とも言う。圧縮データを復号する場合の開始地点となる位置を示す。）を記憶する。

【 0 0 3 1 】

ビット列出現頻度取得部 1 1 0 は、最終符号化前バイナリデータに含まれる同一のビット列ごとに、最終符号化前バイナリデータにおける出現頻度を取得する。例えば、図 7 に示す最終符号化前バイナリデータ 2 においては、各ビット列の出現頻度は図 8 のようになる。ただし、上述した圧縮境界を跨ぐビット列の出現数はビット列の出現頻度を含めない。

20

なお、上記ビット列の桁数は符号化方法にもよるが自然言語であれば 3 2 桁程度が適当である。そこで、本実施形態では、上記ビット列の桁数は 3 2 桁とする。ただし、ビット列の桁数は 3 2 桁に限定されず任意の桁数で構わない。

【 0 0 3 2 】

ビット列識別番号関連付部 1 1 1 は、最終符号化前バイナリデータに出現する同一のビット列ごとに、ビット列を識別するビット列識別番号を関連付けてビット列識別番号記憶部 1 1 2 に記憶する。このとき、ビット列識別番号関連付部 1 1 1 は、出現頻度が 1 から 5 番目に高いビット列にビット列識別番号を関連付ける。また、出現頻度が高いビット列から順に、値が小さいビット列識別番号を関連付ける。例えば、図 7 に示す最終符号化前バイナリデータ 2 であれば、図 9 に示すように、出現頻度が 1 番目、2 番目、3 番目、4 番目、5 番目に高いビット列「1 2 A 5 B 8 C A」、ビット列「D F 4 3 A 6 8 C」、ビット列「3 8 5 C 6 5 F 9」、ビット列「9 3 5 A D 6 C D」、ビット列「B 5 8 C E E A 5」に、それぞれビット列識別番号 0、1、2、3、4 を関連付ける。

30

【 0 0 3 3 】

ビット列識別番号記憶部 1 1 2 は、図 9 に示すように、最終符号化前バイナリデータに出現する同一のビット列ごとに、ビット列を識別するビット列識別番号を関連付けて記憶する。

40

【 0 0 3 4 】

ビット列識別番号変換部 1 1 3 は、ビット列識別番号記憶部 1 1 2 を参照して、最終符号化前バイナリデータに出現する同一のビット列を、ビット列に関連付けられたビット列識別番号に変換する。ただし、ビット列識別番号変換部 1 1 3 は、ビット列が上述した圧縮境界を跨ぐ場合は、ビット列をビット列識別番号に変換しない。これにより、見出し語単位で復号可能なように符号化されたデータ（圧縮データ）を生成することができる。

【 0 0 3 5 】

参照フラグ付き符号化部 1 1 4 は、デルタ符号化方法、Variable Byte Code 符号化方法、ハフマン符号化方法といった符号化方法を用いて、ビット列識別番号変換部 1 1 3 が変換したビット列識別番号をバイナリデータに符号化する。そして、ビット列識別番号のバ

50

イナリデータに参照フラグ（復号時にビット列識別番号記憶部 1 1 2 を参照することを示すフラグ）を関連付ける。

このように、頻出するビット列をコード長の短いビット列識別番号のバイナリデータに符号化することで、符号化されたデータ（圧縮データ）のサイズを小さくすることができる。

【 0 0 3 6 】

非参照フラグ付き符号化部 1 1 5 は、最終符号化前バイナリデータに出現するビット列のうちビット列識別番号が関連付けられていないビット列について、ビット列の先頭からビット列識別番号が関連付けられたビット列が出現するまでのビットの桁数をバイナリデータに符号化する。この際、ビットの桁数のバイナリデータに、非参照フラグ（復号時にビット列識別番号記憶部 1 1 2 を参照しないことを示すフラグ）を関連付ける。

また、上記ビット列が圧縮境界を跨ぐ場合は、ビット列の先頭から圧縮境界までのビットの桁数をバイナリデータに符号化し、このバイナリデータに上記非参照フラグを関連付ける。

さらに、ビット列識別番号が関連付けられたビット列のうち圧縮境界を跨ぐためビット列識別番号に変換されなかったビット列についても、圧縮境界を跨ぐ場合と跨がない場合に応じて同様の処理を行う。

【 0 0 3 7 】

圧縮データ記憶部 1 1 6 は、符号化対象のデータが符号化されたバイナリデータ（圧縮データ）を記憶する。

【 0 0 3 8 】

以上のような符号化装置 1 0 0 が実行する符号化処理の流れについて、図 3 に示すフローチャートを参照して説明する。

【 0 0 3 9 】

[符号化処理]

表示部 1 0 2 が、符号化候補のデータを記録したファイルの名称（ファイル名）を表示装置 1 4 に表示しているとする。ユーザは、表示装置 1 4 に表示されたファイル名を参照して、これら符号化候補の中から符号化対象とするもののファイル名を入力装置 1 3 に入力する。入力装置 1 3 は、入力されたファイル名を示す信号を生成し、入力部 1 0 3 に供給する。入力部 1 0 3 は、入力装置 1 3 から供給された信号を受け付け、ユーザが符号化対象としたファイルを特定する。そして、ユーザが符号化対象としたファイルのデータを符号化する旨の指示を文字出現頻度取得部 1 0 4 に供給する。文字出現頻度取得部 1 0 4 は、この指示を受け付け、図 3 に示す符号化処理を開始する。

【 0 0 4 0 】

まず、文字出現頻度取得部 1 0 4 は、指示に基づいて、符号化候補データ記憶部 1 0 1 が記憶する複数の符号化候補のデータの中から符号化対象のデータを取得する（ステップ S 1 0）。以下、理解を容易にするために、符号化対象のデータとして図 4 に示す辞書データ 1 が取得されたものとする。

【 0 0 4 1 】

文字出現頻度取得部 1 0 4 は、辞書データ 1 に含まれる文字ごとに、辞書データ 1 における出現頻度を取得する（ステップ S 1 1）。なお、図 5 に示すように、辞書データ 1 において文字「t」の出現頻度が 9 2 0 4 1 回で最も高い。続いて、文字「h」、文字「e」、文字「s」、文字「r」・・・の出現頻度がそれぞれ 8 3 8 9 0 回、8 0 9 8 4 回、7 6 1 8 9 回、6 8 6 0 7 回・・・で 2 番目、3 番目、4 番目、5 番目・・・に高い。

【 0 0 4 2 】

文字識別番号関連付部 1 0 5 は、辞書データ 1 において出現頻度が高い文字から順に、値が小さい文字識別番号を関連付けて文字識別番号記憶部 1 0 6 に記憶する（ステップ S 1 2）。具体的には、図 6 に示すように、出現頻度が 1 番目、2 番目、3 番目、4 番目、5 番目・・・に高い文字「t」、文字「h」、文字「e」、文字「s」、文字「r」・・・に、それぞれ文字識別番号 0、1、2、3、4・・・を関連付ける。

【 0 0 4 3 】

変換部 1 0 7 は、文字識別番号記憶部 1 0 6 を参照して、辞書データ 1 に含まれる文字を、文字に関連付けられた文字識別番号に変換する（ステップ S 1 3）。具体的には、辞書データ 1 に含まれる文字「t」、文字「h」、文字「e」・・・をそれぞれ文字識別番号 0、1、2・・・に変換する。

【 0 0 4 4 】

符号化部 1 0 8 は、変換部 1 0 7 が変換した文字識別番号をバイナリデータに符号化する（ステップ S 1 4）。具体的には、辞書データ 1 が図 7 に示す最終符号化前バイナリデータ 2 に符号化される。なお、上述したように、説明の都合上、図 7 に示す最終符号化前バイナリデータ 2 を 1 6 進数表記の複数の数値で表していることに留意されたい。実際には、最終符号化前バイナリデータ 2 は、0 又は 1 を表すビットの集合である。

10

【 0 0 4 5 】

ビット列出現頻度取得部 1 1 0 は、最終符号化前バイナリデータ 2 に出現する同一のビット列ごとに、最終符号化前バイナリデータ 2 における出現頻度を取得する（ステップ S 1 5）。ただし、辞書データ 1 に含まれる見出し語の開始位置（圧縮境界）を跨ぐビット列の出現数はビット列の出現頻度を含めない。

なお、図 8 に示すように、最終符号化前バイナリデータ 2 において、ビット列「1 2 A 5 B 8 C A」の出現頻度が 1 5 0 回で最も高い。また、ビット列「D F 4 3 A 6 8 C」、ビット列「3 8 5 C 6 5 F 9」、ビット列「9 3 5 A D 6 C D」、ビット列「D 5 8 C E E A 5」、ビット列「1 B 3 C 2 A 0 9」・・・の出現頻度がそれぞれ 1 3 0 回、1 0 0 回、8 0 回、7 0 回、4 0 回・・・で 2 番目、3 番目、4 番目、5 番目、6 番目・・・に高い。

20

【 0 0 4 6 】

ビット列識別番号関連付け部 1 1 1 は、出現頻度が 1 から 5 番目に高いビット列に対して、出現頻度が高いビット列から順に、値が小さいビット列識別番号を関連付けてビット列識別番号記憶部 1 1 2 に記憶する（ステップ S 1 6）。具体的には、図 9 に示すように、出現頻度が 1 番目、2 番目、3 番目、4 番目、5 番目に高いビット列「1 2 A 5 B 8 C A」、ビット列「D F 4 3 A 6 8 C」、ビット列「3 8 5 C 6 5 F 9」、ビット列「9 3 5 A D 6 C D」、ビット列「B 5 8 C E E A 5」に、それぞれビット列識別番号 0、1、2、3、4 を関連付ける。

30

【 0 0 4 7 】

ビット列識別番号変換部 1 1 3 は、ビット列識別番号記憶部 1 1 2 を参照して、最終符号化前バイナリデータ 2 に出現する同一のビット列を、ビット列に関連付けられたビット列識別番号に変換する（ステップ S 1 7）。具体的には、ビット列「1 2 A 5 B 8 C A」、ビット列「D F 4 3 A 6 8 C」、ビット列「3 8 5 C 6 5 F 9」・・・を、それぞれビット列識別番号 0、1、2・・・に変換する。ただし、ビット列識別番号変換部 1 1 3 は、圧縮境界を跨ぐビット列については、ビット列識別番号への変換を行わない。

【 0 0 4 8 】

参照フラグ付き符号化部 1 1 4 は、ビット列識別番号変換部 1 1 3 が変換したビット列識別番号をバイナリデータに符号化する。そして、ビット列識別番号のバイナリデータに参照フラグ（復号時にビット列識別番号記憶部 1 1 2 を参照することを示すフラグ）を関連付ける（ステップ S 1 8）。

40

【 0 0 4 9 】

次に、非参照フラグ付き符号化部 1 1 5 は、最終符号化前バイナリデータ 2 に出現するビット列のうちビット列識別番号が関連付けられていないビット列について、ビット列の先頭からビット列識別番号が関連付けられたビット列が出現するまでのビットの桁数をバイナリデータに符号化する。この際、ビットの桁数のバイナリデータに、非参照フラグ（復号時にビット列識別番号記憶部 1 1 2 を参照しないことを示すフラグ）を関連付ける。また、上記ビット列が圧縮境界を跨ぐ場合は、ビット列の先頭から圧縮境界までのビットの桁数をバイナリデータに符号化し、このバイナリデータに上記非参照フラグを関連付け

50

る。(ステップS 19)。

さらに、非参照フラグ付き符号化部 115 は、ビット列識別番号が関連付けられたビット列のうち圧縮境界を跨ぐためビット列識別番号に変換されなかったビット列についても、圧縮境界を跨ぐ場合と跨がない場合に依じて同様の処理を行う(ステップS 20)。

【0050】

そして、非参照フラグ付き符号化部 115 は、ステップS 10 からS 20 までの処理によって辞書データ1 が符号化されたバイナリデータ(圧縮データ)を圧縮データ記憶部 116 に記憶する(ステップS 21)。

【0051】

次に上記符号化処理によって符号化された圧縮データを部分的に復号する復号装置 200 の物理構成及び機能構成を説明する。

【0052】

本実施形態に係る復号装置 200 は、物理的には図 10 に示すように構成される。即ち、復号装置 200 は、ROM 20 と、RAM 21 と、外部記憶装置 22 と、入力装置 23 と、表示装置 24 と、CPU 25 と、を備える。

【0053】

ROM 20 は、各種初期設定、ハードウェアの検査、プログラムのロード等を行うための初期プログラムを記憶する。RAM 21 は、CPU 25 が実行する各種ソフトウェアプログラム、これらのソフトウェアプログラムの実行に必要なデータ等を一時的に記憶する。

【0054】

外部記憶装置 22 は、例えば、ハードディスクであって、各種ソフトウェアプログラム、データ等を記憶する。これらソフトウェアプログラムの中には、アプリケーションソフトウェアプログラム、OS のような基本ソフトウェアプログラム等が含まれている。

【0055】

入力装置 23 は、キーボード、マウス、トラックパッド等を備え、ユーザからの入力を受け付ける。入力装置 23 は、キーボード、マウス、トラックパッド等からの入力に基づいて信号を生成し、CPU 25 に供給する。

【0056】

表示装置 24 は、液晶ディスプレイ等の画面を備え、CPU 25 から供給されたテキストデータや画像データを画面に表示する。

【0057】

CPU 25 は、外部記憶装置 22 に記憶されたソフトウェアプログラムを RAM 21 に読み出して、そのソフトウェアプログラムを実行制御することにより、以下の機能構成を実現する。

【0058】

復号装置 200 は、機能的には図 11 に示すように構成される。即ち、復号装置 200 は、ビット列識別番号記憶部 201 と、圧縮データ記憶部 202 と、開始位置記憶部 203 と、ビット列識別番号復号方法記憶部 204 と、入力部 205 と、ビット列識別番号復号部 206 と、ビット列識別番号変換部 207 と、文字列識別番号復号方法記憶部 208 と、文字識別番号記憶部 209 と、復号部 210 と、変換部 211 と、表示部 212 と、を備える。なお、ビット列識別番号記憶部 201、圧縮データ記憶部 202、開始位置記憶部 203、文字識別番号記憶部 209 は、それぞれ符号化装置 100 が備えるビット列識別番号記憶部 112、圧縮データ記憶部 116、開始位置記憶部 109、文字識別番号記憶部 106 が記憶するデータと同じデータを記憶している。ビット列識別番号記憶部 201 と、圧縮データ記憶部 202 と、開始位置記憶部 203 と、ビット列識別番号復号方法記憶部 204 と、文字識別番号復号方法記憶部 208 と、文字識別番号記憶部 209 と、は図 10 に示す外部記憶装置 22 に構築されている。

【0059】

ビット列識別番号記憶部 201 は、最終符号化前バイナリデータに出現する同一のビッ

10

20

30

40

50

ト列ごとに、ビット列を識別するビット列識別番号を関連付けて記憶する。

【 0 0 6 0 】

圧縮データ記憶部 2 0 2 は、符号化対象のデータが上記符号化処理によって符号化されたバイナリデータ（圧縮データ）を記憶する。

【 0 0 6 1 】

開始位置記憶部 2 0 3 は、符号化対象のデータに含まれる見出し語の開始位置（圧縮境界）を記憶する。

【 0 0 6 2 】

ビット列識別番号復号方法記憶部 2 0 4 は、圧縮データに含まれるビット列識別番号のバイナリデータをビット列識別番号に復号する方法を記憶する。具体例を挙げると、ビット列識別番号のバイナリデータがデルタ符号化方法を用いて符号化されている場合には、ビット列識別番号復号方法記憶部 2 0 4 は、デルタ符号化方法で符号化されたバイナリデータを符号化前の元データに戻すロジックをビット列識別番号復号方法として記憶する。

10

【 0 0 6 3 】

入力部 2 0 5 は、入力装置 2 3 が生成した信号を受け付ける。この信号は、ユーザが入力装置 2 3 を介して指定した見出し語を示す。入力部 2 0 5 は、ユーザが指定した見出し語及びその見出し語の例文であることを復号対象のデータを特定するための条件に設定する。そして、ビット列識別番号復号部 2 0 6 に設定した条件を入力する。

【 0 0 6 4 】

ビット列識別番号復号部 2 0 6 は、圧縮データのうち入力部 2 0 5 から入力された条件を満たすデータを復号対象のデータとして特定する。例えば、見出し語「the」とその例文であることが条件であれば、圧縮データに含まれる見出し語「the」とその例文を復号対象のデータとして特定する。なお、圧縮データにおける見出し語及び例文の位置は、開始位置記憶部 2 0 3 が記憶する各見出し語の開始位置に基づいて特定される。

20

【 0 0 6 5 】

また、ビット列識別番号復号部 2 0 6 は、見出し語の開始位置に参照フラグと非参照フラグのどちらが存在するか判別する。参照フラグが存在すると判別した場合、参照フラグに関連付けられているビット列識別番号のバイナリデータをビット列識別番号に復号する。復号方法は、ビット列識別番号復号方法記憶部 2 0 4 が記憶する復号方法が採用される。

30

一方、非参照フラグが存在すると判別した場合、復号部 2 1 0 に復号処理の制御を移す。

【 0 0 6 6 】

ビット列識別番号変換部 2 0 7 は、ビット列識別番号記憶部 2 0 1 を参照し、ビット列識別番号復号部 2 0 6 が復号したビット列識別番号を、ビット列識別番号に関連付けられたビット列に変換する。そして、変換したビット列をキューにコピーする。

【 0 0 6 7 】

文字識別番号復号方法記憶部 2 0 8 は、圧縮データに含まれる文字識別番号のバイナリデータを文字識別番号に復号する方法を記憶している。具体例を挙げると、文字識別番号のバイナリデータがデルタ符号化方法を用いて符号化されている場合には、文字識別番号復号方法記憶部 2 0 8 は、デルタ符号化方法で符号化されたバイナリデータを符号化前の元データに戻すロジックを文字列識別番号復号方法として記憶する。

40

【 0 0 6 8 】

文字識別番号記憶部 2 0 9 は、圧縮データに含まれる文字と、文字を識別する文字識別番号と、を関連付けて記憶している。

【 0 0 6 9 】

復号部 2 1 0 は、ビット列識別番号変換部 2 0 7 がキューにコピーしたビット列を文字識別番号に復号する。

一方、復号部 2 1 0 は、ビット列識別番号復号部 2 0 6 から復号処理の制御を移された場合、非参照フラグに関連付けられているビットの桁数を示すデータを読み込む。そして

50

、上記桁数分のバイナリデータをさらに読み込み、キューにコピーする。そして、キューにコピーしたバイナリデータを文字識別番号に復号する。

なお、復号方法は、文字識別番号復号方法記憶部 208 が記憶する復号方法が採用される。

【0070】

変換部 211 は、文字識別番号記憶部 209 を参照し、復号部 210 が復号した文字識別番号を文字に変換する。

【0071】

表示部 212 は、圧縮データが復号されたデータ（符号化前の元データ）を表示装置 24 に表示する。

【0072】

以上のような復号装置 200 が実行する復号処理の流れについて、図 12 に示すフローチャートを参照して説明する。

【0073】

[復号処理]

ここで、圧縮データ記憶部 202 が記憶する圧縮データは、図 4 に示す辞書データ 1 のバイナリデータであるとする。そして、ユーザが、「the」の定義や「the」の使用例を調べるために、入力装置 23 に見出し語「the」を入力したとする。この場合、入力装置 23 は、入力された見出し語「the」を示す信号を生成し、入力部 205 に供給する。入力部 205 は、入力装置 23 から供給された信号を受け付け、入力された見出し語が「the」であることを特定する。そして、見出し語「the」とその例文であることを復号対象の条件に設定し、ビット列識別番号復号部 206 に設定した条件を入力する。ビット列識別番号復号部 206 は、入力部 205 から上記条件を受け付け、図 12 に示す復号処理を開始する。

【0074】

ビット列識別番号復号部 206 は、圧縮データのうち上記条件を満たすデータを復号対象のデータとして特定する（ステップ S30）。ここで上記条件を満たすのは見出し語「the」とその例文である。従って、見出し語「the」とその例文が復号対象のデータとして特定される。

【0075】

次に、ビット列識別番号復号部 206 は、見出し語「the」の開始位置に参照フラグと非参照フラグのどちらが存在するか判別する（ステップ S31）。参照フラグが存在すると判別した場合、参照フラグに関連付けられているビット列識別番号のバイナリデータを読み込む。そして、ビット列識別番号復号方法記憶部 204 が記憶する復号方法を用いて、ビット列識別番号のバイナリデータをビット列識別番号に復号する（ステップ S32）。なお、ここではステップ S32 において復号されたビット列識別番号を「0」とする。

【0076】

ビット列識別番号変換部 207 は、ビット列識別番号記憶部 201 を参照して、ビット列識別番号復号部 206 が復号したビット列識別番号を、ビット列識別番号に関連付けられたビット列に変換する（ステップ S33）。そして、変換したビット列をキューにコピーする（ステップ S34）。具体的には、ステップ S32 において復号されたビット列識別番号「0」を、ビット列識別番号「0」に関連付けられたビット列「12A5B8CA」（図 9 参照）に変換し、ビット列「12A5B8CA」をキューにコピーする。

【0077】

復号部 210 は、キューに存在するビット列を、文字識別番号復号方法記憶部 208 が記憶する復号方法を用いて文字識別番号に復号する（ステップ S35）。ここでは、ビット列「12A5B8CA」が文字識別番号「0」、「1」、「2」、「3」に復号されたとする。

【0078】

10

20

30

40

50

変換部 2 1 1 は、文字識別番号記憶部 2 0 9 を参照し、復号部 2 1 0 が復号した文字識別番号を文字に変換する（ステップ S 3 6）。具体的には、ステップ S 3 5 で復号された文字識別番号「0」、「1」、「2」、「3」を、それぞれの文字識別番号に関連付けられた文字「t」、「h」、「e」、「s」（図 6 参照）に変換する。

【 0 0 7 9 】

文字への変換を終えると、変換部 2 1 1 は、全ての復号対象のデータを文字に変換したか否かを判別する（ステップ S 3 7）。

【 0 0 8 0 】

変換部 2 1 1 は、復号対象のデータに変換されていない部分が存在すると判別した場合（ステップ S 3 7 ; No）、復号処理の制御をビット列識別番号復号部 2 0 6 に移す。この場合、ビット列識別番号復号部 2 0 6 は、復号済みであるビット列識別番号のバイナリデータの後尾に参照フラグと非参照フラグのどちらが存在するか判別する（ステップ S 3 1）。ビット列識別番号復号部 2 0 6 は、非参照フラグが存在すると判別した場合、復号処理の制御を復号部 2 1 0 に移す。この場合、復号部 2 1 0 は、非参照フラグに関連付けられているビットの桁数のバイナリデータを読み込む（ステップ S 3 8）。そして、復号済みであるビット列識別番号のバイナリデータの後尾から上記桁数分のバイナリデータを読み込み（ステップ S 3 9）、キューにコピーする（ステップ S 4 0）。

【 0 0 8 1 】

復号部 2 1 0 は、キューに存在するバイナリデータを、文字識別番号復号方法記憶部 2 0 8 が記憶する復号方法を用いて文字識別番号に復号する（ステップ S 3 5）。

【 0 0 8 2 】

変換部 2 1 1 は、文字識別番号記憶部 2 0 9 を参照し、復号部 2 1 0 が復号した文字識別番号を文字に変換する（ステップ S 3 6）。

【 0 0 8 3 】

文字への変換を終えると、変換部 2 1 1 は、全ての復号対象のデータを文字に変換したか否かを判別する（ステップ S 3 7）。復号対象のデータに変換されていない部分が存在すると判別した場合（ステップ S 3 7 ; No）、復号処理の制御をビット列識別番号復号部 2 0 6 に移す。そして、ステップ S 3 1 から S 3 7 までの処理が、全ての復号対象のデータが文字に変換されるまで繰り返し実行される。

【 0 0 8 4 】

ステップ S 3 7 において、変換部 2 1 1 が全ての復号対象のデータを文字に変換したと判別した場合（ステップ S 3 7 ; Yes）、表示部 2 1 2 は、変換された文字を表示装置 2 4 に表示する（ステップ S 4 1）。具体的には、見出し語「the」とその例文を表示装置 2 4 に表示する。

【 0 0 8 5 】

このように、上記復号処理では、圧縮データの一部がユーザに指定された見出し語の開始位置から逐次的に復号される。その際、参照フラグと非参照フラグのどちらが存在するかが復号前に判別され、復号しようとしているバイナリデータが文字識別番号のバイナリデータかビット列識別番号のバイナリデータかが特定される。ここで、参照フラグが存在すると判別された場合は、ビット列識別番号のバイナリデータであると特定され、非参照フラグが存在すると判別された場合は、文字識別番号のバイナリデータであると特定される。そして、特定されたバイナリデータの種別に応じた復号方法で、文字識別番号のバイナリデータは文字識別番号に復号され、ビット列識別番号のバイナリデータはビット列に復号される。

ここで注目すべき点は、文字識別番号を文字に変換するために必要となる変換前の元データ、ビット列を文字識別番号に変換するために必要となる変換前の元データを文字識別番号記憶部 2 0 9 とビット列識別番号記憶部 2 0 1 から自在に取得できることである。これは、圧縮データの一部である復号対象のデータに変換前の元データが存在しなくても、元データに変換可能ということを意味する。従って、従来 of 符号化方法・復号方法であれば、符号化対象のデータを予め部分的に符号化しておかなければ部分的復号を実行できな

10

20

30

40

50

かったが、本願発明によればそのようなことをしなくても部分的復号を実行できる。

【0086】

以上説明したように、本実施形態に係る符号化装置100は、符号化対象のデータを文字列識別番号及びビット列識別番号のバイナリデータに符号化する。そして、復号装置200は、文字識別番号記憶部209とビット列識別番号記憶部201を参照し、符号化装置100が符号化したバイナリデータ(圧縮データ)を部分的に復号する。これらは、上述したように、符号化対象のデータを部分的に符号化しなくても圧縮データの部分的復号を可能とする。従って、従来の符号化方法・復号方法に比べて符号化に手間がかからず、符号化効率が高い。

【0087】

(変形例)

以上に本発明の実施形態について説明したが、上記実施形態は一例であり、本発明の適用範囲はこれに限られない。すなわち、本発明の実施形態は種々の応用が可能であり、あらゆる実施の形態が本発明の範囲に含まれる。

【0088】

例えば、上記実施形態では、出現頻度が1から5番目に高いビット列にビット列識別番号を関連付けたが、ビット列識別番号を関連付ける対象が、出現頻度が1から5番目に高いビット列に限定されるわけではない。例えば、出現頻度が1から10番目に高いビット列にビット列識別番号を関連付けてもよいし、全てのビット列にビット列識別番号を関連付けてもよい。ただし、出現頻度が低いビット列をコード長が長いビット列識別番号のバイナリデータに符号化するとかえって圧縮率が低下するため、ビット列識別番号を関連付ける対象を出現頻度が高いビット列に限定したほうが好ましい。

【0089】

また、上記実施形態では、見出し語の開始位置を圧縮境界としたが、見出し語の開始位置と見出し語の例文の開始位置を圧縮境界としてもよい。

【0090】

また、上記実施形態では、符号化装置100と復号装置200とを互いに独立した装置として記載した。しかし、符号化装置100が、復号装置200の機能を全て備え、復号装置として動作してもよいし、復号装置200が、符号化装置100の機能を全て備え、符号化装置として動作してもよい。

【0091】

また、復号装置200は、圧縮データから復号対象のデータを特定する特定部を備えていてもよい。そして、図12に示す復号処理のステップS30で、特定部が圧縮データから復号対象のデータを特定してもよい。この場合、ビット列識別番号復号部206は、特定部が特定した復号対象のデータを復号する。

【0092】

なお、本発明に係る機能を実現するための構成を予め備えた符号化装置、復号装置として提供できることはもとより、プログラムの適用により、既存のパーソナルコンピュータや情報端末機器等を、本発明に係る符号化装置、復号装置として機能させることもできる。すなわち、上記実施形態で例示した符号化装置、復号装置による各機能構成を実現させるためのプログラムを、既存のパーソナルコンピュータや情報端末機器等を制御するCPU等が実行できるように適用することで、本発明に係る符号化装置、復号装置として機能させることができる。また、本発明に係る符号化方法、復号方法は、符号化装置、復号装置を用いて実施できる。

【0093】

また、このようなプログラムの適用方法は任意である。プログラムを、例えば、コンピュータが読取可能な記録媒体[CD-ROM(Compact Disc Read Only Memory)、DVD(Digital Versatile Disc)、MO(Magneto Optical disc)等]に格納して適用できる他、インターネット等のネットワーク上のストレージにプログラムを格納しておき、これをダウンロードさせることにより適用することもできる。

10

20

30

40

50

【 0 0 9 4 】

以上、本発明の好ましい実施形態について説明したが、本発明は係る特定の実施形態に限定されるものではなく、本発明には、特許請求の範囲に記載された発明とその均等の範囲とが含まれる。以下に、本願出願の当初の特許請求の範囲に記載された発明を付記する。

【 0 0 9 5 】

(付記 1)

符号化対象のデータに含まれる文字と、該文字を識別する文字識別番号と、を関連付けて記憶する文字識別番号記憶部と、

前記文字識別番号記憶部を参照して、前記符号化対象のデータに含まれる文字を、該文字に関連付けられた文字識別番号に変換する変換部と、

10

前記変換部が変換した文字識別番号をバイナリデータに符号化する符号化部と、

前記符号化部が符号化した文字識別番号のバイナリデータに出現する同一のビット列ごとに、該ビット列を識別するビット列識別番号を関連付けてビット列識別番号記憶部に記憶するビット列識別番号関連付部と、

前記ビット列識別番号記憶部を参照して、前記文字識別番号のバイナリデータに出現する同一のビット列を、該ビット列に関連付けられたビット列識別番号に変換するビット列識別番号変換部と、

前記ビット列識別番号変換部が変換したビット列識別番号に、復号時に前記ビット列識別番号記憶部を参照することを示す参照フラグを関連付けて、該ビット列識別番号をバイナリデータに符号化する参照フラグ付き符号化部と、

20

を備える符号化装置。

【 0 0 9 6 】

(付記 2)

前記文字識別番号は、前記符号化対象のデータにおいて出現頻度の高い文字に関連付けられたものほど小さく、

前記符号化部は、前記文字識別番号が小さいほど該文字識別番号を符号化して得られるバイナリデータのデータ量が小さい符号化方法を用いて、前記文字識別番号をバイナリデータに符号化する、

付記 1 に記載の符号化装置。

30

【 0 0 9 7 】

(付記 3)

前記文字識別番号のバイナリデータに出現するビット列のうち前記ビット列識別番号が関連付けられていないビット列については、該ビット列の先頭から前記ビット列識別番号が関連付けられたビット列が出現するまでのビットの桁数に、復号時に前記ビット列識別番号記憶部を参照しないことを示す非参照フラグを関連付けて、該ビットの桁数をバイナリデータに符号化する非参照フラグ付き符号化部、

を備える付記 1 又は 2 に記載の符号化装置。

【 0 0 9 8 】

(付記 4)

前記ビット列識別番号は、前記文字識別番号のバイナリデータにおいて出現頻度の高い同一のビット列に関連付けられたものほど小さく、

40

前記参照フラグ付き符号化部は、前記ビット列識別番号が小さいほど該ビット列識別番号を符号化して得られるバイナリデータのデータ量が小さい符号化方法を用いて、前記ビット列識別番号をバイナリデータに符号化する、

付記 1 乃至 3 の何れかに記載の符号化装置。

【 0 0 9 9 】

(付記 5)

前記符号化対象のデータは、見出し語を含み、

前記同一のビット列の出現頻度は、前記符号化対象のデータにおける前記見出し語の開

50

始位置を跨ぐ該同一のビット列の出現数を含まない、

付記 4 に記載の符号化装置。

【 0 1 0 0 】

(付記 6)

前記符号化対象のデータは、見出し語を含み、

前記ビット列識別番号変換部は、前記ビット列識別番号が関連付けられたビット列が、前記見出し語の開始位置を跨ぐ場合、該ビット列を前記ビット列識別番号に変換せず、

前記非参照フラグ付き符号化部は、前記ビット列識別番号が関連付けられたビット列のうち前記ビット列識別番号変換部が前記ビット列識別番号に変換しなかったビット列について、該ビット列の先頭から該ビット列が跨ぐ前記見出し語の開始位置までのビットの桁数に、前記非参照フラグを関連付けて、該ビットの桁数をバイナリデータに符号化する、
付記 3 に記載の符号化装置。

10

【 0 1 0 1 】

(付記 7)

前記非参照フラグ付き符号化部は、前記ビット列識別番号が関連付けられていないビット列が、前記見出し語の開始位置を跨ぐ場合、該ビット列の先頭から該開始位置までのビットの桁数に、前記非参照フラグを関連付けて、該ビットの桁数をバイナリデータに符号化する、

付記 6 に記載の符号化装置。

【 0 1 0 2 】

(付記 8)

前記ビット列識別番号記憶部が記憶する同一のビット列の桁数は、3 2 桁である、

付記 1 乃至 7 の何れかに記載の符号化装置。

【 0 1 0 3 】

(付記 9)

符号化対象のデータに含まれる各文字を識別する各文字識別番号のバイナリデータと、前記各文字識別番号のバイナリデータに出現する同一のビット列を識別するビット列識別番号のバイナリデータと、で構成された圧縮データを記憶する圧縮データ記憶部と、

前記符号化対象のデータに含まれる文字と、該文字を識別する前記文字識別番号と、を関連付けて記憶する文字識別番号記憶部と、

30

前記各文字識別番号のバイナリデータに出現する前記同一のビット列ごとに、該ビット列を識別する前記ビット列識別番号を関連付けて記憶するビット列識別番号記憶部と、

復号対象のデータを特定するための条件を入力する入力部と、

前記圧縮データ記憶部が記憶する圧縮データのうち前記条件を満たすデータを復号対象のデータとして特定し、特定した復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号のバイナリデータとのうち、前記ビット列識別番号のバイナリデータを前記ビット列識別番号に復号するビット列識別番号復号部と、

前記ビット列識別番号記憶部を参照して、前記ビット列識別番号復号部が復号した前記ビット列識別番号を、該ビット列識別番号に関連付けられたビット列に変換するビット列識別番号変換部と、

40

前記復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号変換部が変換したビット列とを前記文字識別番号に復号する復号部と、

前記文字識別番号記憶部を参照して、前記復号部が復号した前記文字識別番号を、該文字識別番号に関連付けられた文字に変換する変換部と、

を備える復号装置。

【 0 1 0 4 】

(付記 10)

符号化対象のデータに含まれる文字と、該文字を識別する文字識別番号と、を関連付けて記憶する文字識別番号記憶部を参照して、前記符号化対象のデータに含まれる文字を、該文字に関連付けられた文字識別番号に変換する変換ステップと、

50

前記変換ステップで変換した文字識別番号をバイナリデータに符号化する符号化ステップと、

前記符号化ステップで符号化した文字識別番号のバイナリデータに出現する同一のビット列ごとに、該ビット列を識別するビット列識別番号を関連付けてビット列識別番号記憶部に記憶するビット列識別番号関連付ステップと、

前記ビット列識別番号記憶部を参照して、前記文字識別番号のバイナリデータに出現する同一のビット列を、該ビット列に関連付けられたビット列識別番号に変換するビット列識別番号変換ステップと、

前記ビット列識別番号変換ステップで変換したビット列識別番号に、復号時に前記ビット列識別番号記憶部を参照することを示す参照フラグを関連付けて、該ビット列識別番号をバイナリデータに符号化する参照フラグ付き符号化ステップと、

を備える符号化方法。

【 0 1 0 5 】

(付記 1 1)

復号対象のデータを特定するための条件を入力する入力ステップと、

符号化対象のデータに含まれる各文字を識別する各文字識別番号のバイナリデータと、前記各文字識別番号のバイナリデータに出現する同一のビット列を識別するビット列識別番号のバイナリデータと、で構成された圧縮データのうち、前記条件を満たすデータを復号対象のデータとして特定する特定ステップと、

前記特定ステップで特定した前記復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号のバイナリデータとのうち、前記ビット列識別番号のバイナリデータを前記ビット列識別番号に復号するビット列識別番号復号ステップと、

前記各文字識別番号のバイナリデータに出現する前記同一のビット列ごとに、該ビット列を識別する前記ビット列識別番号を関連付けて記憶するビット列識別番号記憶部を参照して、前記ビット列識別番号復号ステップで復号した前記ビット列識別番号を、該ビット列識別番号に関連付けられたビット列に変換するビット列識別番号変換ステップと、

前記復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号変換ステップで変換したビット列とを前記文字識別番号に復号する復号ステップと、

前記符号化対象のデータに含まれる文字と、該文字を識別する前記文字識別番号と、を関連付けて記憶する文字識別番号記憶部を参照して、前記復号ステップで復号した前記文字識別番号を、該文字識別番号に関連付けられた文字に変換する変換ステップと、

を備える復号方法。

【 0 1 0 6 】

(付記 1 2)

コンピュータを、

符号化対象のデータに含まれる文字と、該文字を識別する文字識別番号と、を関連付けて記憶する文字識別番号記憶部を参照して、前記符号化対象のデータに含まれる文字を、該文字に関連付けられた文字識別番号に変換する変換部、

前記変換部が文字識別番号をバイナリデータに符号化する符号化部、

前記符号化部が符号化した文字識別番号のバイナリデータに出現する同一のビット列ごとに、該ビット列を識別するビット列識別番号を関連付けてビット列識別番号記憶部に記憶するビット列識別番号関連付部、

前記ビット列識別番号記憶部を参照して、前記文字識別番号のバイナリデータに出現する同一のビット列を、該ビット列に関連付けられたビット列識別番号に変換するビット列識別番号変換部、

前記ビット列識別番号変換部が変換したビット列識別番号に、復号時に前記ビット列識別番号記憶部を参照することを示す参照フラグを関連付けて、該ビット列識別番号をバイナリデータに符号化する参照フラグ付き符号化部、

として機能させるためのプログラム。

【 0 1 0 7 】

(付記 1 3)

コンピュータを、

復号対象のデータを特定するための条件を入力する入力部、

符号化対象のデータに含まれる各文字を識別する各文字識別番号のバイナリデータと、前記各文字識別番号のバイナリデータに出現する同一のビット列を識別するビット列識別番号のバイナリデータと、で構成された圧縮データのうち、前記条件を満たすデータを復号対象のデータとして特定する特定部、

前記特定部が特定した前記復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号のバイナリデータとのうち、前記ビット列識別番号のバイナリデータを前記ビット列識別番号に復号するビット列識別番号復号部、

前記各文字識別番号のバイナリデータに出現する前記同一のビット列ごとに、該ビット列を識別する前記ビット列識別番号を関連付けて記憶するビット列識別番号記憶部を参照して、前記ビット列識別番号復号部が復号した前記ビット列識別番号を、該ビット列識別番号に関連付けられたビット列に変換するビット列識別番号変換部、

前記復号対象のデータを構成する前記文字識別番号のバイナリデータと前記ビット列識別番号変換部が変換したビット列とを前記文字識別番号に復号する復号部、

前記符号化対象のデータに含まれる文字と、該文字を識別する前記文字識別番号と、を関連付けて記憶する文字識別番号記憶部を参照して、前記復号部が復号した前記文字識別番号を、該文字識別番号に関連付けられた文字に変換する変換部、

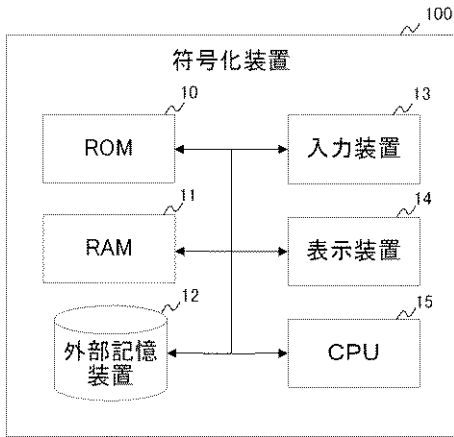
として機能させるためのプログラム。

【符号の説明】

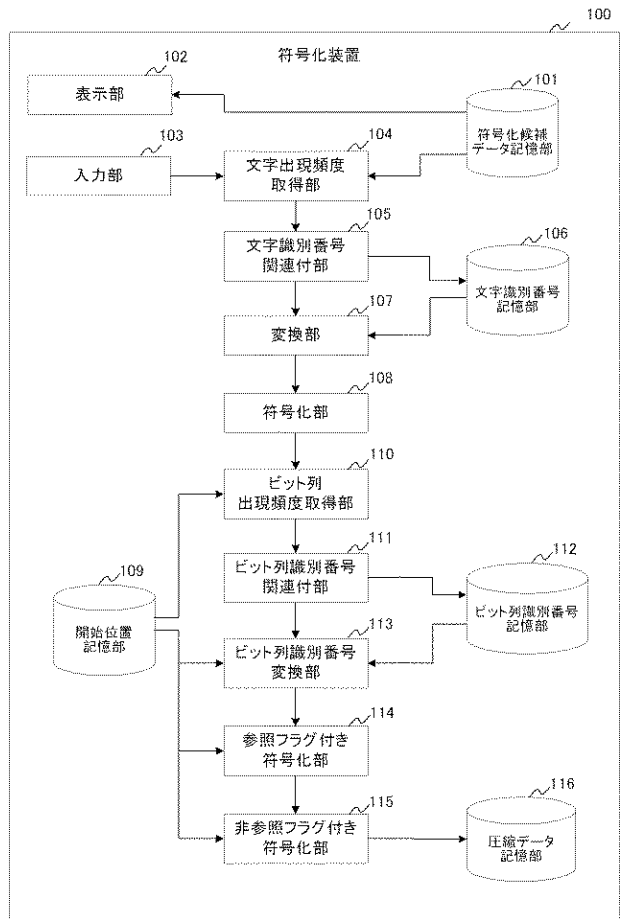
【 0 1 0 8 】

1 辞書データ、 2 最終符号化前バイナリデータ、 1 0 ROM、 1 1 RAM、 1 2 外部記憶装置、 1 3 入力装置、 1 4 表示装置、 1 5 CPU、 1 0 0 符号化装置、 1 0 1 符号化候補データ記憶部、 1 0 2 表示部、 1 0 3 入力部、 1 0 4 文字出現頻度取得部、 1 0 5 文字識別番号関連付部、 1 0 6 文字識別番号記憶部、 1 0 7 変換部、 1 0 8 符号化部、 1 0 9 開始位置記憶部、 1 1 0 ビット列出現頻度取得部、 1 1 1 ビット列識別番号関連付部、 1 1 2 ビット列識別番号記憶部、 1 1 3 ビット列識別番号変換部、 1 1 4 参照フラグ付き符号化部、 1 1 5 非参照フラグ付き符号化部、 1 1 6 圧縮データ記憶部、 2 0 0 復号装置、 2 0 ROM、 2 1 RAM、 2 2 外部記憶装置、 2 3 入力装置、 2 4 表示装置、 2 5 CPU、 2 0 0 復号装置、 2 0 1 ビット列識別番号記憶部、 2 0 2 圧縮データ記憶部、 2 0 3 開始位置記憶部、 2 0 4 ビット列識別番号復号方法記憶部、 2 0 5 入力部、 2 0 6 ビット列識別番号復号部、 2 0 7 ビット列識別番号変換部、 2 0 8 文字識別番号復号方法記憶部、 2 0 9 文字識別番号記憶部、 2 1 0 復号部、 2 1 1 変換部、 2 1 2 表示部

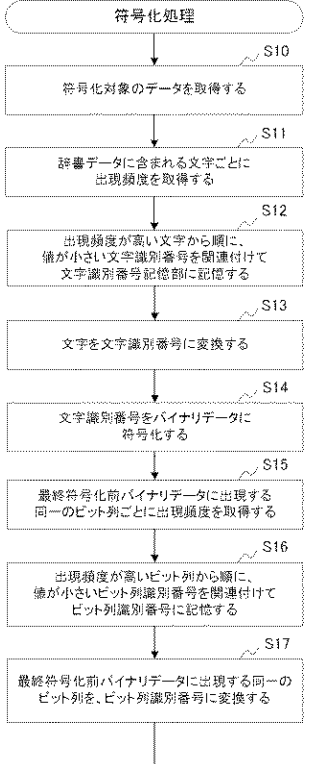
【図1】



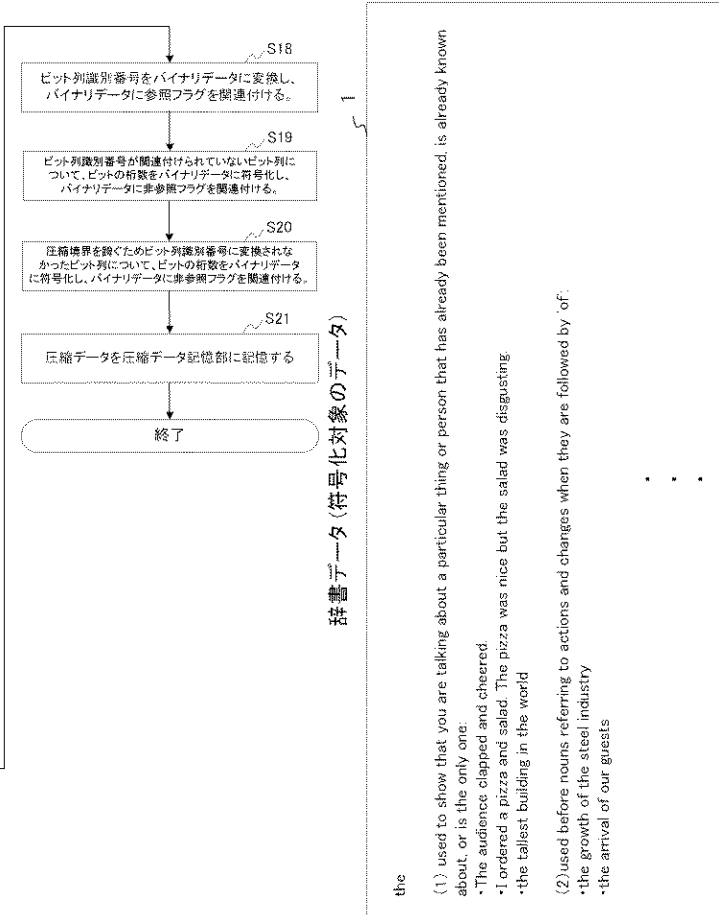
【図2】



【図3】



【図4】



最終符号化前バイナリデータ(16進数表記)

2

12A5B8CA ... 1B3C2A09 ... 385C65F9 ...
--

【図7】

【図5】

文字	出現頻度
t	92041
h	83890
e	80984
s	76189
r	68607
:	:
:	:

【図8】

ビット列(16進数表記)	出現頻度
12A5B8CA	150
DF43A68C	130
385C65F9	100
935AD6CD	80
B58CEEA5	70
1B3C2A09	40
:	:
:	:
:	:

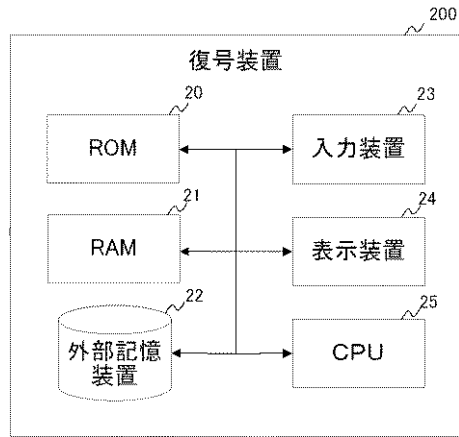
【図6】

文字	文字識別番号
t	0
h	1
e	2
s	3
r	4
:	:
:	:

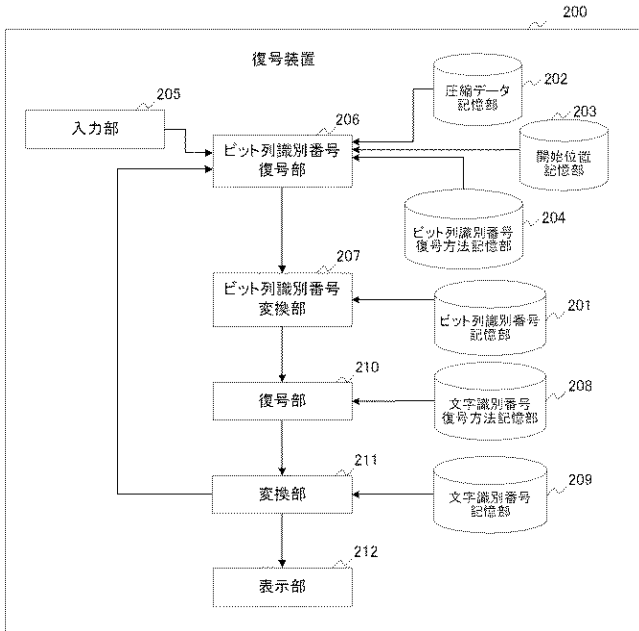
【図9】

ビット列識別番号	
0	12A5B8CA
1	DF43A68C
2	385C65F9
3	935AD6CD
4	B58CEEA5
ビット列(16進数表記)	

【図10】



【図11】



【図12】

