

(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2008-262248

(P2008-262248A)

(43) 公開日 平成20年10月30日(2008.10.30)

(51) Int.Cl. F 1 テーマコード (参考)
G 0 6 F 17/21 (2006.01) G 0 6 F 17/21 5 9 0 E 5 B 0 0 9
 5 B 1 0 9

審査請求 有 請求項の数 3 O L (全 23 頁)

(21) 出願番号 特願2007-97568 (P2007-97568)
 (22) 出願日 平成19年4月3日(2007.4.3)
 (11) 特許番号 特許第4004060号 (P4004060)
 (45) 特許公報発行日 平成19年11月7日(2007.11.7)
 (31) 優先権主張番号 特願2007-71097 (P2007-71097)
 (32) 優先日 平成19年3月19日(2007.3.19)
 (33) 優先権主張国 日本国(JP)

(71) 出願人 398057868
 加賀美 徹也
 静岡県三島市松が丘1番地の8 シャリエ
 三島松が丘902
 (74) 代理人 100083507
 弁理士 田中 二郎
 (72) 発明者 加賀美 徹也
 静岡県三島市松が丘1番地の8 シャリ
 エ三島松が丘902
 Fターム(参考) 5B009 ME14 MH07 NG02 VA02 VB01
 VB11 VB17
 5B109 ME14 MH07 NG02 VA02 VB01
 VB11 VB17

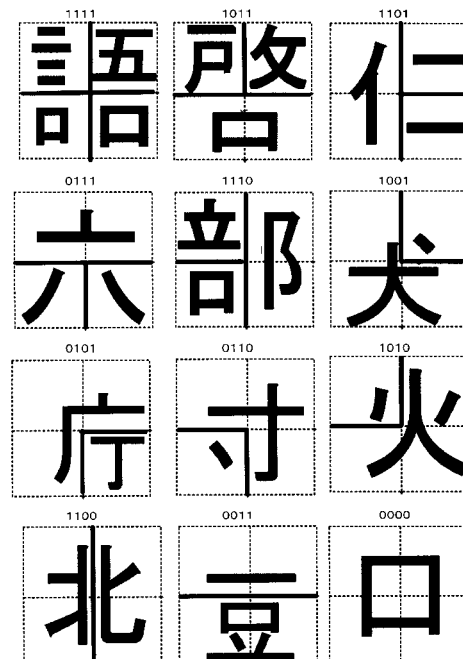
(54) 【発明の名称】 文字検索方法

(57) 【要約】

【課題】 漢字やハングルとの文字の文字検索が、文字知識や特別な装置を前提とせず、簡易な方法で正確、迅速に検索できる文字検索方法を提供する。

【解決手段】 検索文字の構成要素の間隙を、縦方向及び横方向に分割して、この分割の可否をコードに置き換えることで文字をコード化して分類し、前記分類コードを入力することにより文字の検索を可能とし、その後得られた検索文字の意味を多言語や動画等で表示することで文字理解を可能とした文字検索方法。

【選択図】 図3



【特許請求の範囲】

【請求項 1】

検索文字の構成要素の間隙を、縦方向及び横方向に分割して、この分割の可否をコードに置き換えることで文字をコード化し、前記コードとそれに対応する文字を分類して記憶手段に記憶せしめ、前記コードを入力することにより前記記憶手段より文字を検索し、その後得られた文字の意味を多言語や動画等で表示することで文字理解を可能とした文字検索方法。

【請求項 2】

前記コードの入力と一緒に検索文字の発音情報を入力して検索することを特徴とする請求項 1 記載の文字検索方法。

10

【請求項 3】

字形から得た前記コードと共に当該文字に関するコード化した文法情報と意味情報を付加入力することにより、文字情報に加え文法情報と意味情報の組み合わせから検索することを特徴とする請求項 1 又は請求項 2 記載の文字検索方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、漢字やハングル等の文字を簡易な方法で正確、迅速に検索する検索方法に関するものである。

【背景技術】

20

【0002】

従来より、文字、特に漢字を検索する方法としていくつかの特許が提案されている。特許文献 1 に示すものは文字を構成する 1 つ以上の構成要素によって文字を検索するものである。

この発明によれば、文字論理式入力手段 10 から入力された文字論理式に含まれる文字の部品を文字部品特定手段 11 において特定し、これを文字論理式に代入して部品論理式を作成する。作成した部品論理式を部品論理式演算手段 12 において演算し、演算結果として得られた部品の集合を検索条件として該当文字特定手段 13 が文字部品データベース 15 を参照し、該当する文字を特定するものである。

【特許文献 1】特開 2003 - 30183

30

【0003】

前記特許文献 1 に記載の発明は、漢字の文字知識を前提とし、文字をいくつかの部品に分割し、この分割された文字を部品単位で加算、減算、乗算等するように構成しなくてはならず、そのために演算文字検索装置を用いている。

【0004】

前記演算文字検索装置の操作のために、漢字の文字知識を前提とし、文字論理式入力 10、文字部品特定手段 11 等の各手段が必要となるため検索装置の構成が複雑となり、検索方法も複雑なものとなっていた。

【発明の開示】

【発明が解決しようとする課題】

40

【0005】

本発明は上記の点に鑑みなされたものであり、漢字やハングルの文字知識や特別な装置を前提とせず、簡易な方法で行え、かつ正確、迅速に検索できる文字検索方法を提供するものである。

【課題を解決するための手段】

【0006】

本発明の要旨とするところは、検索文字の構成要素の間隙を、縦方向及び横方向に分割して、この分割の可否をコードに置き換えることで文字をコード化し、前記コードとそれに対応する文字を分類して記憶手段に記憶せしめ、前記コードを入力することにより前記記憶手段より文字を検索し、その後得られた文字の意味を多言語や動画等で表示すること

50

で文字理解を可能とした文字検索方法である。

【0007】

また本発明の要旨とするところは、前記コードの入力と一緒に検索文字の発音情報を入力して検索する文字検索方法である。

また本発明の要旨とするところは、字形から得た前記コードと共に当該文字に関するコード化した文法情報と意味情報を付加入力することにより、文字情報に加え文法情報と意味情報の組み合わせから検索することを特徴とする請求項1または請求項2記載の文字検索方法。

【発明の効果】

【0008】

本発明はたとえばインターネットを利用したホームページ上において、漢字あるいはハングル等の検索文字に対する文字知識を持たない欧米人や、文字知識が不十分な学習者、あるいは文字知識を習得済みでフロントエンドプロセッサなどを利用し文字変換処理に習熟した人など幅広い利用者が想定される場面で有効である。

特に、漢字等の文字知識や文字処理システムを持たない欧米人などが漢字等を検索する場合にASCIIコードなどの1バイト系の文字処理装置でも本発明の文字を分割したコードを数字等で入力し、文字を検索することができる。さらに検索した文字の意味も多言語や動画で容易に理解することができる。

【0009】

また、漢字やハングル等の検索文字の文字知識や文字処理システムを持つ日本人や中国人などがフロントエンドプロセッサなどで文字検索の一種である文字変換処理をシフトJISコードやGBコードなどの2バイト系の文字処理装置で、従来の発音情報に加えて本発明のコードと一緒に入力することで変換効率を向上させることができる。

【発明を実施するための最良の形態】

【0010】

本発明の最良の実施形態は、検索文字の構成要素の間隙を、縦方向及び横方向に分割して、この分割の可否をコードに置き換えることで文字をコード化し、前記コードとそれに対応する文字を分類して記憶手段に記憶せしめ、入力手段から前記コードを入力して、演算手段を用いて前記記憶手段より文字を検索することで、漢字の部首や書き順などの文字知識を持たない人でも文字の検索を可能とし、その後、得られた検索文字の意味を多言語や動画等で表示することで文字理解を可能とする。

また、前記検索文字の発音情報を前記コードと一緒に入力して分類することにより、漢字の発音などの文字知識を持つ人には従来の発音情報のみの検索方法よりも文字検索効率を向上させることができる。

【0011】

以下、本発明の実施形態を図に基づいて説明する。

本発明は、図1の記憶手段にコードに対応する文字データベースと処理プログラムを記憶しておくだけで、図2に示すプログラムを実行し検索を行うことができ、発音情報と分類コードを仮名漢字変換ソフト(フロントエンドプロセッサ)のユーザー辞書に追加登録するだけで、文字検索効率を向上させることができる。

【0012】

図1は本発明の一実施形態を示す機能構成ブロック図である。

たとえばインターネット上に公開されている日本語の漢字辞書ホームページをダウンロードしてパソコンや携帯電話などの情報機器上で辞書検索をする場合を想定する。図4に示すような文字4分割コードと漢字を分類し表組み形式で閲覧できる漢字辞書ホームページをZIP方式などで圧縮したファイルとしてダウンロードして解凍し、パソコンの記憶手段30などに予め記憶しておく。

【0013】

図1の記憶手段30には、図4に示すような文字4分割コードとそれに対応する漢字を1つのレコードとしてコードごとに分類した配列のデータベース形式で記憶しておく。利

10

20

30

40

50

ユーザーが閲覧する表示も図4のごとき形式だが、ホームページはシフトJISで作成されていることが多いので、実際は図5に示すような16進数のシフトJISで表記されるコード形式で図1の記憶手段30内に記憶されている。

【0014】

図1の文字コード等入力手段10により、たとえばパソコンや携帯電話の数字キーなどを利用して入力した文字4分割コードを、記憶手段30に予め記憶された文字データベースの文字4分割コードと順次図1の演算手段20（CPUなど）において照合する。実際の照合は図5に示す4分割コードのシフトJIS表記形式で行う。

【0015】

照合処理に必要なプログラムは記憶手段30に事前にインストールされたホームページ閲覧ソフト（ブラウザ）やワープロソフトの検索機能呼び出して用いる。

10

【0016】

入力した4分割コードとデータベースの4分割コードが一致した場合には、その結果として閲覧ページの表組みの中からカーソルキーの直前で一致した4分割コードが図1のたとえば液晶画面などの表示手段40において該当文字列を背景とは異なる色などで強調表示（ハイライト）される。

たとえば図1の入力手段10から1111という4分割文字コードを入力し、予め記憶手段30に記憶した図4の文字データベースを図1の演算手段20で照合した結果、図4の「語」の直前の行にカーソルキーが置かれていた場合には、「語」の左側の「1111」が強調表示されるので、引き続き検索を続けたい場合は、ホームページ閲覧ソフトやワープロソフトの検索機能の「次を検索」ボタンを押すと次の行の「競」の左側の「1111」が強調表示される。このようにして順次目的とする文字を検索することができる。

20

【0017】

また、同様のソフトのオプション機能ボタンを使い、入力した1111という4分割コードに一致した行だけをまとめて一覧表示することもできる。

【0018】

図4で使われる数字の1と0はシフトJISでは図5の31と30という表記で表されるが、もし、パソコンがシフトJISなどの漢字処理機能を持たない場合には、欧米で一般的なASCIIコードでも同一の31と30という表記なので、「語」や「競」などといった文字部分のみをホームページ作成時に予めGIF形式やJPG形式の画像ファイル形式で保存しておけば、検索結果は画像である「語」の左側に表示された「1111」や画像である「競」の左側に表示された「1111」などで強調表示することが可能であり、利用者は文字化けせずに文字を表示することができる。

30

【0019】

上で述べた圧縮したホームページをダウンロード後解凍して検索する方法は、インターネットに接続しなくても閲覧できる利点があるが、インターネットに接続したままホームページ閲覧ソフトの検索機能などを利用してオンライン検索することも同様に可能である。

オンライン検索の場合は、図1の記憶手段30等に一時的に閲覧しているHTML形式のファイルが記憶されている状態にあるので、パソコン等の電源を切りキャッシュメモリが消去されるとダウンロード閲覧のように継続的な利用はできないが、ダウンロードをする手間がかからず、常時記憶手段30などの容量を確保する必要がないという利点がある。

40

【0020】

また、膨大な辞書をオンライン検索する場合は、Perl言語などで予め作成したホームページサーバー側のCGI検索プログラムを利用して文字4分割コードを入力欄に入力すれば、該当する文字のみを一覧表示させることもできる。このようなデータベース検索CGIプログラムはフリーソフト等で一般的に入手が容易であり、利用者のパソコン等にインストールされたホームページ閲覧ソフトの検索機能を使わなくても高速にオンライン検索できる利点がある。

【0021】

図2は本発明の検索処理フローチャートである。

50

S 1 0 0 はたとえばホームページ形式の漢字辞書などを検索するための作業の開始を表す。S 2 0 0 は図 1 の入力手段 1 0 から文字 4 分割コードを入力することを表す。S 3 0 0 は後述する文字 4 分割コードの書式を照合用に書式変換するか否かを判断することを表す。もし、変換する必要がある場合には S 4 0 0 においてたとえばホームページに予め記述された JAVA (登録商標) Script などのスクリプトを利用するなどして書式変換処理を行った後、S 5 0 0 において図 1 の演算手段 2 0 を用いて入力した文字 4 分割コードとデータベースの文字 4 分割コードを照合処理することを表す。S 3 0 0 において書式の変換が必要ないと判断する場合には、入力した文字 4 分割コードの書式のまま S 5 0 0 の照合処理を行う。

書式変換とは、たとえば図 4 の文字 4 分割コードは 4 桁の数字が全て 1 もしくは 0 で表す書式だが、これを 1 2 3 4 と全ての桁を異なる数字で表す書式で入力した場合、1 以外の数字は全て 1 に置換するという簡単なスクリプトをホームページ上で処理させることなどをいう。

【 0 0 2 2 】

ただし、S 3 0 0 と S 4 0 0 は、入力書式とデータベースの書式が異なる場合のみに必要なステップなので、それ以外の利用方法の場合には省略してもよい。

【 0 0 2 3 】

S 6 0 0 は図 1 の表示手段 4 0 においてたとえばホームページ上で照合合致した文字 4 分割コード部分を強調表示することなどを表す。

【 0 0 2 4 】

S 7 0 0 は検索した文字の意味をさらに調べたい場合に、その文字ないしは文字の画像にリンクを予め設定しておき、その文字の上をクリックするなどしてホームページの別の場所にジャンプして文字の意味を説明する画面を表示するか否かを判断する。

【 0 0 2 5 】

仮に利用者が文字をクリックして文字の意味を表示させる場合には、S 8 0 0 においてたとえば多言語 (対訳) の言語情報を表示してもよいし、動画などの非言語情報を表示してもよいことを表す。多言語 (対訳) 情報とはたとえば日本語の漢字「語」と一緒に中国語の「詞(Ci)」や英語の「Word」などを表示することをいう。もし、利用者がこれらの言語を理解できる場合には、日本語の「語」という文字の意味を言語的に類推理解できる利点がある。

【 0 0 2 6 】

仮に言語情報では理解できない利用者の場合には、たとえば「競」という文字をクリックすると、動画 (アニメ等) により人が競技をしている画面を表示するような処理を非言語情報による意味情報の表示という。

【 0 0 2 7 】

もし、利用者が検索した文字の意味情報の表示が必要ないと判断した場合には、S 9 0 0 の再入力のステップに進む。引き続き利用者が異なる文字 4 分割コードを入力する場合には、再び S 2 0 0 から処理を継続し、検索を終了する場合には S 1 0 0 0 の終了ステップとなる。たとえばパソコンのウィンドウを閉じるなどの操作を利用者がした場合に終了となる。

【 0 0 2 8 】

図 3 は、文字の分割方法を示す説明図である。

【 0 0 2 9 】

本実施形態の文字検索方法を例えて言うと、文字を乗せたケーキをナイフで上から 4 分割するものであり、ナイフは文字を構成する線と線の間隙に切り込むことができるが、線に触れてはならないものとする。ここで「線」とは、文字の構成要素である直線、曲線、点などの図形の総称を指すものとする。

【 0 0 3 0 】

前記ケーキを時計に例えて、ナイフを切り込む方向を時計の中心から見て 1 2 時方向を「縦方向の上半分 (略称「上」)」と、6 時方向を「縦方向の下半分 (略称「下」)」と、9

10

20

30

40

50

時方向を「横方向の左半分（略称「左」）」と、3時方向を「横方向の右半分（略称「右」）」と呼ぶ。

【0031】

文字を4分割する順序は任意に設定できるが、本実施形態では、まず上、ついで下、3番目に左、最後に右の順序とする。

そして、文字を分割できる場所を1、分割できない場所を0という数字で表し、上 下 左 右の順序に、1または0の組み合わせから成る4桁の数字で検索対象の文字を表し分類し、これを「文字4分割コード」または略称で「コード」と呼ぶ。また愛称として「ケーキカット法」などの名称を用いることにより、コードの適用規則を比喻により理解しやすくできる。

10

【0032】

4分割線を上下左右という一般的な名称で呼び習わす方法に加え、赤緑青黄などの色彩名称を対照させ着色した分割線で図示してもよい。

【0033】

前記4分割コードは、0000から1111までの16通りが考えられる。この16通りのコードの内、1つの文字に複数の分割方法がある場合、「できるだけ多く、かつできるだけ平等に文字を分割できるコードを優先する」という条件の適合度に応じた優先度規則を使う。ケーキを平等に分け合うという比喻で理解がしやすくなる。

そして、検索や表示などの処理は必要に応じて優先度の高いコードを優先度の低いコードよりも先に適用できる。

20

【0034】

図3の「語」は、最も優先度の高い第1番目の優先度コードである1111を表すものである。このコードは、上 下 左 右の順番に文字を4分割したことを意味し、分割可能な箇所を実線で示している。

【0035】

図3の「啓」と「仁」と「六」と「部」は、第2番目の優先度コードを表すものである。この2番目のコードはケーキを3分割するように文字を分類したコードであり、これらのコード間は同一優先度である。

前記コードのうち1011は、図3に示すように、上 左 右の順に分割したことを意味し、例えば「啓」という文字が相当する

30

前記コードのうち1101は、図3に示すように、上 下 右の順に分割したことを意味し、例えば「仁」という文字が相当する

前記コードのうち0111は、図3に示すように、下 左 右の順に分割したことを意味し、例えば「六」という文字が相当する

前記コードのうち1110は、図3に示すように、上 下 左の順に分割したことを意味し、例えば「部」という文字が相当する。

【0036】

図3の「北」と「豆」は、第3番目の優先度コードを表すものである。

この3番目の優先度コードはケーキを2分割するように文字を分類したコードである。これらのコード間は同一優先度である。

40

前記コードのうち1100は、図3に示すように、上 下の順に分割したことを意味し、例えば「北」という文字が相当する。

前記コードのうち0011は、左 右の順に分割したことを意味し、例えば「豆」という文字が相当する。

【0037】

図3の「犬」と「庁」と「寸」と「火」は、第4番目の優先度コードを表すものである。

この4番目の優先度コードはケーキを2分割するように文字を分類したコードである。これらのコード間は同一優先度である。

この場合、2分割のコードという条件は前記3番目の優先度コードと同様であるが、「

50

できるだけ平等に分割する」という条件が適用できないので4番目の規則よりも3番目の規則を優先するのである。

前記コードのうち1001は、上右の順に分割したことを意味し、例えば「犬」という文字が相当する。

前記コードのうち0101は、下右の順に分割したことを意味し、例えば「庁」という文字が相当する。

前記コードのうち0110は、下左の順に分割したことを意味し、例えば「寸」という文字が相当する。

前記コードのうち1010は、上左の順に分割したことを意味し、例えば「火」という文字が相当する。

「火」は1001とも分割できるが、本発明では重複してデータベースを作成することにより、どちらのコードを入力しても目的の文字が検索できるよう冗長性を許してもよい。

【0038】

5番目の優先度コードは理論上は1000、0001、0100、0010の4個のコードが該当するが、ケーキの一部にナイフを切り込めても分割することができないため分割規則から除外する。

従って、0000という分割不可能なコードのみを最も優先度の低いコードとして採用する。このコードに相当する文字は、例えば図3の「口」である。

【0039】

このように前記コードの組み合わせは、理論上は16通りとなるが5番目のコード処理に従い、最終的には12通りの組み合わせを採用する。

【0040】

なお、前記5番目の優先度コードは「1箇所のみ切り込み可能な文字はほとんど存在せず、1箇所といえども切り込みが不可能な文字は少なからず存在する」という主として漢字の字形に即した対応となっているので、漢字以外の文字、例えばハングルでは4つのコードを除外せず16通りのコードを使ってもよい。

【0041】

また、1つの文字に複数のコードが存在する場合、規則利用者がいずれのコードを指定しても処理ができるよう冗長性を持ったデータベースを作成してもよい。例えば「火」という文字は、1010でもよいし、1001でもよい。

【0042】

文字4分割コードの書式例を説明する。

1文字の文字コード書式には5種類の書式がある。

【0043】

「2進数(ビット)書式」(通称「2進数4桁書式」)は、1文字の上下左右各四分の一ずつの4分割線を0(非分割)か1(分割)の2進数(ビット)でそれぞれ表す4桁(4ビット)の書式で図3がこの書式例である。位置情報は上下左右の順に固定で4桁未満の省略表示はしない。

【0044】

「10進数書式」は、「10進数非省略書式」と「10進数省略書式」に分かれる。

【0045】

「10進数非省略書式」(通称「10進数4桁書式」)は、0(非分割)、1(縦方向上半分分割)、2(縦方向下半分分割)、3(横方向左半分分割)、4(横方向右半分分割)の5つの数字で表す。位置情報の順番は非固定だが4桁未満の省略表示はしない。

昇順の例は1234、1204などであり、降順の例は4321、4021などであり、任意順の例は2341、2401などであり、非分割の例は0000である。

【0046】

「10進数省略書式」は、「10進数非省略書式」と同じ規則だが上下左右全ての分割線が分割不可能な場合のみを0で表し、2つ以下の0は省略表示できる。

10

20

30

40

50

「10進数非省略書式」の例を「10進数省略書式」で表すと、昇順の例は1234、124などであり、降順の例は4321、421などであり、任意順の例は2341、241などであり非分割の例は0である。

【0047】

「日常語書式」は「日常語非省略書式」と「日常語省略書式」に分かれる。

【0048】

「日常語非省略書式」は、「10進数非省略書式」の「1234」の代わりに「上下左右」や「UDLR(Up Down Left Rightの頭文字)」を使う。「赤緑青黄」などの色彩名称を使ってもよい。書式の規則は「10進数非省略書式」と同じである。

【0049】

昇順の例「1234」は「上下左右」か「UDLR」、「1204」は「上下0右」か「UD0R」などで、降順の例「4321」は「右左下上」か「RLDU」、「4021」は「右0下上」か「R0DU」などで、任意順の例「2341」は「下左右上」か「DLRU」、「2401」は「下右0上」か「DR0U」などで、非分割の例「0000」は「0000」などである。

【0050】

「日常語省略書式」は、「10進数省略書式」の「1234」の代わりに「上下左右」や「UDLR(Up Down Left Rightの頭文字)」を使う。書式の規則は「10進数省略書式」と同じで、コードを続ける場合には1文字につき4桁ずつという規則性がないため、文字単位に相当する箇所にハイフンなどの区切り記号の挿入を必須とする。

【0051】

昇順の例「1234」は「上下左右」か「UDLR」、「124」は「上下右」か「UDR」などで、降順の例「4321」は「右左下上」か「RLDU」、「421」は「右下上」か「RDU」などで、任意順の例「2341」は「下左右上」か「DLRU」、「241」は「下右上」か「DRU」などで、非分割の例「0」は「0」で表す。

【0052】

「16進数圧縮書式」(通称「16進数1桁書式」)は、「2進数(ビット)書式」を16進数に変換して1文字で表す。

たとえば、次のような1桁の表示が可能となる。2進数の0000は16進数で0と表し、2進数の0101(10進数の5)は16進数では5と表し、2進数の1010(10進数の10)は16進数ではAと表し、2進数の1111(10進数の15)は16進数ではFと表すので、習熟すると入力大幅に効率化できる。

【0053】

「連想文字圧縮書式」(通称「連想1桁書式」)は図6に示すように「2進数(ビット)書式」を連想しやすいアルファベット等に置き換えて1文字で表す。図6は10進数省略書式と連想文字圧縮書式を対照してある。同じ1桁でも、16進数圧縮書式は論理的だが記憶しにくいいため、初心者には連想文字圧縮書式のほうが記憶しやすく効率がよいという利点がある。

【0054】

次に2文字以上の文字列の4分割コード書式を説明する。

たとえば図4の「北」と「山」という2文字からなる「北山」という苗字を4分割文字コードで表す場合、2進数書式では、図4の4分割コード「1100」と「0000」をつなげて「11000000」と8桁の数字で表すことができるので、もし名簿などを作成する場合は、図4の4分割コードに「11000000」を加え、その右側に「北山」という文字を併記すればよい。

しかし、数字の羅列が見分けにくいとか、数字の0をたくさん入力するのに手間がかかるなどというさまざまな理由から、文字列にも書式の規定が必要となる。

【0055】

「2進数(ビット)書式」の文字列の書式は隣り合う文字と文字の区切り記号のハイフン(-)等を挿入してもよいし、しなくてもよい。理由は4桁ずつコード列が一定に連続

10

20

30

40

50

しているので識別しやすいからである。

【 0 0 5 6 】

区切り記号を挿入しない書式は、内部データはハイフン (-) を挿入せず 4 桁 (4 ビット) ずつ文字列に対応するコードを列記する形式で記憶してあるので、区切り記号なしの 2 進数書式は入力書式と記憶データ書式が同一で誤処理が少ないという長所がある。

【 0 0 5 7 】

区切り記号を挿入する書式は、利用者の入力文字数が 0 と 1 以外にハイフン (-) 記号の分だけ増加するが、入力する利用者が目視して文字コードの区切りを識別しやすいという長所がある。

【 0 0 5 8 】

「 1 0 進数非省略書式 」も、「 2 進数 (ビット) 書式 」の文字列の書式と同じ理由から区切り記号の挿入は任意とする。

【 0 0 5 9 】

「 1 0 進数省略書式 」は、隣り合う文字と文字の区切り記号としてハイフン (-) 等を挿入する。理由はコード列が 1 桁から 4 桁まで一定の長さを持たず変化するため、1 文字分のコードを識別できないからである。

例えば 1 2 3 4 は、区切り記号を挿入すれば 1 2 3 4 (1 文字のコード) か 1 2 - 3 4 (2 文字列のコード列) かが識別できる。

【 0 0 6 0 】

「 日常語非省略書式 」は隣り合う文字と文字の区切り記号のハイフン (-) 等を挿入してもよいし、しなくてもよい。理由は 4 桁ずつコード列が一定に連続しているので識別しやすいからである。

【 0 0 6 1 】

「 日常語省略書式 」は隣り合う文字と文字の区切り記号としてハイフン (-) 等を挿入する。理由はコード列が 1 桁から 4 桁まで一定の長さを持たず変化するため、1 文字分のコードを識別できないからである。

【 0 0 6 2 】

例えば、上下左右 (U D L R) は、区切り記号を挿入すれば (1 文字のコード) 上下左右 (U D L R) か上下 - 左右 (U D - L R) (2 文字列のコード列) かが識別できる。

【 0 0 6 3 】

「 1 6 進数圧縮書式 」は区切り記号は不要だが文字コードの先頭と末尾に # 等の記号を挿入する。その理由は 1 6 進数のコード書式は、数字の 0 から 9 までとアルファベットの A から F (数字 1 5 に相当) までを使い、数字と一部のアルファベットが混在するため、その他の文字コード書式や単なる数字とアルファベット文字列の連続と混同しないよう識別するために # 記号等を文字コード先頭と末尾に挿入するのである。明示的に入力や表示をする場合は全角であっても半角であってもよい。

【 0 0 6 4 】

例えば、「大」の 1 6 進数圧縮書式は「 0 」 (ゼロ) だが、1 0 進数省略書式の「 0 」と識別する場合は 1 6 進数圧縮書式は「 # 0 # 」と明示的に表示する。

【 0 0 6 5 】

本発明の実施形態では「 日常語書式 」と「 1 6 進数圧縮書式 」の「 D 」が重複するが、後者は「 # 」で明示的に識別可能であり、仮に「 # 」記号が脱落しても前者の「 D 」は単独で用いられることはないことから識別できる。また、「 D 」以外はアルファベットを用いる書式間で重複することはない。これらの特長を利用してソフトウェアの処理系に誤処理防止の照合ルーチンを付加してもよい。

【 0 0 6 6 】

1 6 進数圧縮書式はできるだけ少ない文字数で迅速かつ効率的に入力を行うことが主な目的なので文字コードの先頭と末尾に # 記号等を挿入することで続くコード列が 1 6 進数 1 字が 1 文字に対応することは識別が可能なので区切り記号のハイフン (-) 等は不要である。

10

20

30

40

50

【 0 0 6 7 】

「連想文字圧縮書式」は数字を使わず全てアルファベット等で表示するため（0はZ）、区切り記号は不要である。仮に16進数圧縮書式と同じアルファベットで表示でも#記号で識別が可能である。

例えば、「大」の16進数圧縮書式は「#0#」、連想文字圧縮書式は「Z」である。

【 0 0 6 8 】

前記書式を用いることで、漢字知識のない人でも文字4分割コードのみで漢字を検索することが可能となるが、漢字知識のある人や漢字学習者にも文字4分割コードは有益である。

【 0 0 6 9 】

たとえばパソコンや携帯電話などで日本語や中国語の漢字変換ソフト（フロントエンドプロセッサ）を利用して漢字を入力する場合に従来よりも扱いやすく効率を高めることができる。

【 0 0 7 0 】

図7は中国で採用している「五筆字型」と呼ばれる漢字入力法に使う専用キーボードである。図8は一般的なASCII配列のキーボードだが、図7の五筆字型キーボードには、Zを除くアルファベットキーごとに、漢字の部首を簡略化した構成要素が割り当てられている。

【 0 0 7 1 】

この漢字入力法は漢字の発音を使わず、構成要素や書き順といった字形の文字知識を組み合わせて使う。図9は、五筆字型で「程」という漢字を入力する方法を示している。

【 0 0 7 2 】

「程」は、「禾」「口」「王」と書き順に従って構成要素を組み合わせることができるという伝統的な漢字知識をキーボードの位置を表す3 1 2 3 1 1という数字で置き換える。

あるいは、前記3つの構成要素が割り当てられたキーを「T」「K」「G」とアルファベットで置き換える。

【 0 0 7 3 】

しかし、五筆字型の入力方法を習得するには、100以上の構成要素や書き順などの漢字知識のほか、どのキーにどの構成要素が割り当てられているかという配置などの専用装置の知識や訓練も必要であったため、パソコンや携帯電話などの操作には不向きである。

【 0 0 7 4 】

これに対し、漢字の発音情報（読み方）をローマ字やピンインと呼ばれるアルファベットで入力し漢字変換するフロントエンドプロセッサがパソコンや携帯電話などの操作には広く普及している。

【 0 0 7 5 】

しかし、中国語や日本語の漢字の発音には「同音語」と呼ばれる同じ発音を持つ漢字が多数存在するため、漢字変換の際に場合によっては列挙表示される同音語漢字変換候補の中から目的の漢字を選択するのにスペースバーなどを何回もたたいてしらみつぶしに探してゆくという煩雑な操作が必要であった。

【 0 0 7 6 】

たとえば図10に示す中国語の同音漢字は膨大な数になり、次々にスペースバーをたたいて変換候補の中から目的とする文字を探さねばならなかった。

図10の1は『現代漢語詞典』という単語辞典に掲載された「Y I」という発音の単漢字リストであり、全部で109字ある。

図10の2は『新華字典』という漢字字典に掲載された「S H I」という発音の単漢字リストであり、全部で67字ある。

図10の3は日本のJISに相当する中国の国家標準（GB）コードに含まれる「L I」という発音の単漢字リストであり、全部で75字ある。

仮にフロントエンドプロセッサが1回に表示する同音漢字変換候補数を10字とすれ

10

20

30

40

50

ば、たとえば「Y I」の変換操作にスペースバーを最大で11回近くたたいて探す必要があり不便であった。

【0077】

こうした問題を解決するため、本発明は、文字4分割コードという漢字知識を必要としない字形情報と従来の発音情報を組み合わせることで漢字変換効率を向上させることを実現した。

【0078】

図11は、図10の「Y I」、「S H I」、「L I」という同音漢字グループ3つに対し、4分割コードを組み合わせると細分類した字数と比率を示す表とグラフである。

【0079】

たとえば、「Y I」という発音グループの「伊」は「1100」と4分割コードで表せ、「S H I」という発音グループの「使」も「1100」、「L I」という発音グループの「礼」も「1100」と4分割コードで表せる。

【0080】

その結果、「Y I」、「S H I」、「L I」の「1100」グループは27字、21字、26字がそれぞれ所属し、同音語グループ全体の25%、31%、35%とそれぞれ三分の一から四分の一程度にまで減らすことができた。

【0081】

そこで、4分割コードは12種類あるので、単純に計算するとそれぞれの4分割コードは平均すると約8パーセントずつ同音語を分散させることが理論的には可能となるので、これを同音語の分散率とよぶことにする。

【0082】

そして、図11のグラフを見ると、「1100」以外の4分割コードはほとんど10%以下の同音語分散率であり、漢字変換の際にスペースバーをたたく回数は1回から3回で済むことがわかる。

【0083】

このように、4分割コードと発音を組み合わせると漢字変換は大きな至便性が得られる。

【0084】

しかし、図11の「1100」グループはほかのコードグループよりも同音漢字数が相対的に字数が多い。そこで、図12には、4桁の分割コードを5桁に拡張した場合の「1100」グループの字数と分散率を抽出した。

【0085】

5桁に分割コードを拡張する規則は単純で、たとえば「例」という漢字は縦方向に2箇所分割可能な箇所があるので、こういう漢字は1箇所を分割しても、さらに「再分割」が可能な漢字とみなす。この考えに基づき、再分割可能な漢字は分割コードの5桁目に「1」を、再分割不可能な漢字は分割コードの5桁目に「0」を加えることとする。

【0086】

その結果、図12では1100という4桁の分割コードを5桁に拡張することにより、分散率を10%台まで改善できた。

【0087】

この結果を図11の分散率と比べると、必ずしも全ての漢字を5桁の分割コードで分類する必要はなく、1100などの一部のコードのみに用いればよいということも物語っている。

【0088】

図13は、1100という4桁の分割コードを6桁にまで拡張した場合の分散率を示す。

従来の画数という字形情報はかなり厳密な適用を前提としていたので外国人や初学者には習得が難しかった。そこで、本発明は、前記5桁の分割コードに続く6桁目に、漢字を一見して「複雑そうか?」「シンプルか?」という直感的な印象で分類できる程度の字画情報を導入した。

【0089】

10

20

30

40

50

具体的には、図13の表の左端の列は、9画以上と7画以下で複雑かシンプルかという定量的基準にし、9画以上を110011、7画以下を110010といった具合に分類した。8画はどちらのグループにも重複して漢字を所属させ、冗長性を持たせてある。

【0090】

図13の表の左から2番目の列は、重複分の8画の字数を振り分け、8画以上と8画以下というふうに重複分を2分して集計したことを表し、表の中央から右の列にそれぞれの結果を示した。たとえば「YI」の110011(8画以上)は、 $6 + 2 = 8$ 字と集計した結果である。

【0091】

この6桁拡張分割コードを利用することで、図13の下段グラフを見ると、ほとんどのグループが8%以下の分散率を達成したことがわかる。

【0092】

このように、4分割コードを漢字の発音情報と組み合わせて利用することで、従来の漢字変換の課題を解決することができた。

【0093】

本実施形態の、文字4分割コード検索法は単漢字の絞込みも効果的に行えるが、特に単語(2つ以上の漢字の組み合わせ)の絞込みに応用した場合にも実用レベルである。

【0094】

さらに前記5桁コードの組み合わせなら、 $24 \text{ 分類} \times 24 \text{ 分類} = 576$ 通りの組み合わせ、すなわち単語分類が可能となる。

たとえばHSKと呼ばれる外国人向けの中国語認定試験に含まれる常用語彙6892単語を576通りの分類で割れば、約12単語であるから、1分類で約12単語が平均の包含数となる。この程度の数であれば、例えば常用中国語で読み方の分からない単語を検索する際に、5桁コードを2回(2文字分)入力するだけで、検索候補数が12単語前後となり、ワープロの漢字変換候補数1回分と殆どかわらないという結果が得られ、実用に耐えるのである。

【0095】

次に漢字変換という一種の入力時の漢字検索ソフトを利用する際の、入力用書式について説明する。フロントエンドプロセッサ等で文字4分割コードのみを入力し漢字等に変換する場合、漢字に変換する必要のない単なる数字列と識別する目的で例えば全角の@ (アット)などの記号を文字4分割コードの先頭と末尾に挿入する。

発音と文字4分割コードの組み合わせ書式も同様に、例えば発音と4分割コードを全角イコール(=)記号などを挿入して組み合わせ情報であることを明示し、かつ、全角の@ (アット)などの記号を組み合わせ情報の先頭と末尾に挿入する。

【0096】

たとえば、「昭和」という文字列を変換して検索するために、予め図4の4分割コードに相当するレコードの先頭に、発音と文字コードを組み合わせた@しょうわ=1204-1200@などの書式を昭和という文字列とともに、フロントエンドプロセッサのユーザー登録辞書などに予め登録しておき、変換の際は、@しょうわ=1204-1200@とキーボードから入力後、スペースバーなどをたたくことで「昭和」という文字列を呼び出して変換することができる。

【0097】

通常は電子メールアドレスに使う@記号は半角なので、全角@などの記号は用いられることが少ない。こうした記号類を文字コードにかかわる範囲指定に明示的に付加することでフロントエンドプロセッサの誤変換を防止する。

【0098】

フロントエンドプロセッサのユーザー辞書登録の方法は、1単語ごとに言語バーと呼ばれる操作ツールを用いて登録してもよいし、予めテキストファイルにたとえば、以下のような「読み」(タブ挿入)「語句」(タブ挿入)「品詞」(タブ挿入)「ユーザーコメント」の順番で登録用のリストを作成しておき、まとめて登録してもよい。

10

20

30

40

50

@しょうわ = 1 2 0 4 - 1 2 0 0 @ 昭和 名詞 リンク

【0099】

フロントエンドプロセッサによっては、ユーザーコメント欄に解説用の文字列を入力表示できるだけでなく、リンクを設定することで例えば別のホームページで多言語情報や動画等の非言語情報を表示できるものもあるので、表示された語句の意味を理解する助けにもなる。

【0100】

@は漢字変換以外にも4分割文字コードに関する表記であることを明示する目的で用いてもよいし、半角で使うことも許容する。

【0101】

16進数圧縮書式は@等と異なる#等の記号を使うことにより、例えばアルファベットの「D」が日常語書式でなく16進数圧縮書式であることを区別することができる。

【0102】

従来の発音入力のための漢字変換では、たとえば「きしゃ」という入力の同音語候補が多数表示された場合、文脈によって誤変換を修正したりする必要があった。

たとえば、「きしゃのきしゃはいいとおもう。」と入力した場合、いくつかの変換の可能性がある。以下に示す4つの例はいずれも文法的な誤りがない変換候補だが、ほとんどのフロントエンドプロセッサはどれか1つの変換しかできない。

貴社の記者はいいと思う。

汽車の記者はいいと思う。

記者の喜捨はいいと思う。

貴社の汽車はいいと思う。

【0103】

ところが、予め「きしゃ」という同音語を本発明の書式でユーザー辞書に登録しておけば、以下のような入力を行うことで希望する漢字の一発変換が可能となる。

@0034 - 1200 = きしゃ@の@1230 - 0000 = きしゃ@はいいとおもう。

@1234 - 0000 = きしゃ@の@1230 - 0000 = きしゃ@はいいとおもう。

@1230 - 0000 = きしゃ@の@0034 - 1204 = きしゃ@はいいとおもう。

@0034 - 1200 = きしゃ@の@1234 - 0000 = きしゃ@はいいとおもう。

【0104】

一発変換できる理由は、発音は同じでも、文字4分割コードがそれぞれ異なるからである。

なお、分割コードと発音の順番は逆でもかまわない。

【0105】

文字コードの詳細書式を説明する。

分野別書式を用いると、さらに精密な漢字検索が可能となる。

たとえば、小規模な専門用語辞書などに限定して検索を行う場合、先頭の@に続けて例えば「かな」で専門用語辞書名を入力し、続けてコロン(：)等を入力して検索範囲を限定する。

【0106】

例えば、苗字専門用語辞書(「みょうじ」と略称)内から「堅田」を入力する場合、

@みょうじ：かただ = 1034 - 0000 @と予め図4の4分割コード欄に登録し、文字欄に堅田と同じ行に登録し、ユーザー辞書登録しておけば、苗字専門用語辞書(「みょうじ」と略称)内からのみ検索変換されるので、専門辞書に登録していない次のような同音語は変換候補として表示されないのが精度が向上する。

@かただ = 1034 - 1204 @ 型だ

【0107】

引用書式が変換効率を向上させる場合がある。たとえば、かなを分割コードで表すと1種類のコードに複数のかなが分類されるので、特定のかな1文字にしぼって検索するのに手間がかかる場合がある。アルファベットや数字も同様である。

10

20

30

40

50

【0108】

この場合は、文字コードではなく、かな、アルファベット、数字、記号などの常用的で単純な文字自体を直接文字コードと混在させて入力する場合、文字自体の前後に引用符号であるダブルクォーテーション(" ")等を付加することにより、文字コードと区別する。

【0109】

例えば「昭和38年」の「38」を引用書式で表すと以下ようになる。

@しょうわ"38"ねん=1204-1200-"38"-0000@

なお、発音部の引用符号は省略してもよい。

【0110】

この場合、引用部分に変数の場合があるので、プログラムで引用部分を除く部分一致ができるようにしておくとうい。

【0111】

部品書式が学習途中の人に有益な場合もある。

検索したい漢字の発音は知らないが、その漢字を構成する部品要素の発音(音や訓)を知っている場合、部品要素の発音をセミコロン(;)等を挿入して列記し、文字コードと組み合わせることができる。ただし、部品の読みの先頭と末尾にもセミコロン(;)等を付加する。

【0112】

例えば、「魏(発音は「ギ」)」の部品書式を次のように表す。

@;い;おに;き;=1230@

【0113】

「魏」は「委」と「鬼」という2つの部品から構成されるので、「委」の発音「い」(音読み)と「鬼」の発音「おに」(訓読み)、「き」(音読み)を列挙したのであるが、部品の一部でもよいので予め辞書登録しておけば、学習者などには難しい漢字を検索する助けとなる。

【0114】

文字コードの簡易書式を説明する。

1バイト系の処理機能しかない情報機器等でフォルダ名やファイル名に4分割文字コードを利用する場合、文字コードの簡易書式を用いる。CD-Rにデータファイルを保存する場合のISO-9660規格を基準にするとファイル名は半角大文字アルファベット8字以内、拡張子は3文字以内で記号はアンダースコア(_)が利用できるため、ファイル名の末尾にアンダースコア(_)に続き次の略称を付ける。アンダースコアとアルファベット略称を合わせて「識別子」と呼ぶ。

【0115】

以下が簡易書式の例である。

2進数書式_B(Binary Numberの略称)

10進数書式_D(Decimal Numberの略称)

日常語書式_C(a Commonly used Wordの略称)

16進数書式_H(Hexadecimal Numberの略称)

連想書式_A(Association of Ideaの略称)

【0116】

例えば、「語」の発音を訓令式ローマ字表記で「GO」と入力し、続けて「1234」と文字コードを入力し、最後に10進数書式の識別子を入力する簡易書式例は以下ようになる。

(全て半角) G01234_D.HTM

【0117】

簡易書式は携帯型音楽再生プレーヤーなどでファイルやフォルダ名称を統一的に検索しやすくする場合などにも有益である。

【0118】

10

20

30

40

50

文字 4 分割コードの入力表示用に使用する文字等の種類と特長を一般的な装置との関係で説明する。

【 0 1 1 9 】

数字のみの書式は最も実用範囲が広い。

「 2 進数 (ビット) 書式」は 0 と 1 のみを使うため、例えば携帯電話の数字ボタンやパソコン数字キーやマウス左右ボタン、ゲーム機コントローラの左右ボタン、入力機能を備えたテレビリモコンなどほとんどの既存装置類の必要最小限の入力手段で入力や表示が可能である。

【 0 1 2 0 】

「 1 0 進数書式」は 0 から 4 までの 5 種類の数字を使うため、例えば携帯電話の数字ボタンやパソコン数字キー、入力機能を備えたテレビリモコンなど一般的な既存の情報機器類等で入力や表示が可能である。

【 0 1 2 1 】

アルファベット、数字、ハイフン (-)、@、#、ダブルクォーテーション (")、コロンの (:)、セミコロン (;) アンダースコア (_) などは通常のパソコンや携帯電話の文字入力手段として利用されている。

【 0 1 2 2 】

前記のごとく全ての文字コード書式に必要な文字が上記数種類に限定されているので、例えば (通話専用電話機を除く) 携帯電話の数字ボタンやパソコン数字キー、入力機能を備えたテレビリモコンなど一般的な既存の情報機器類等で入力や表示が可能である。

【 0 1 2 3 】

「日常語書式」の場合は、「上下左右」のように入力には変換の手間がかかったり、装置によっては制約があるが、伝達の際に日常語彙なので 4 分割文字コードの知識がない人でも理解しやすい。アイコンなどでボタンスイッチを日常語書式で表示すれば、高齢者などの機器の操作に不慣れな人にも操作がしやすい長所もある。

【 0 1 2 4 】

アイコン (図形) で前記表示を代用できれば、幅広い利用が可能である。12 通りから最多で 16 通りの 4 分割文字コードのアイコン等を例えばパソコンや券売機などの入力装置の画面等に表示し、マウスポインターやタッチパネル用のペンや指等で選択するだけで、一般的な既存の装置類等で入力や表示が可能である。この手段の長所は文字コードの書式を学習しなくても、その場で直観的に利用者が入力や表示が可能なことである。

【 0 1 2 5 】

図 2 の S 4 0 0 で示した書式変換方法を説明する。この処理はホームページの HTML 形式で JAVA (登録商標) SCRIPT などの簡易スクリプトにより処理してもよいし、ワープロソフトの置換機能を用いて、利用者が簡単な書式変換を行ってもよい。

【 0 1 2 6 】

「 2 進数 (ビット) 書式」ハイフン (-) なしの場合は、入力書式が図 1 の記憶手段のデータと同一形式なのでそのまま照合する。

【 0 1 2 7 】

「 2 進数 (ビット) 書式」ハイフン (-) 挿入の場合は、ハイフンを照合前に削除処理してから内部データと照合する。

【 0 1 2 8 】

「 1 0 進数非省略書式」ハイフン (-) なしの場合は、「 1 2 3 4 」の順番で基準となる 4 桁の数字列テンプレートを予めメモリに記憶しておき、これと入力文字コードを 4 桁ずつ先頭から照合し、一致する数字を昇順に並び替えるが、入力文字コードに 0 が含まれている場合には照合一致しないので、一致しない数字を 0 に置き換える。この処理を 4 桁ずつ繰り返す。

例えば、@ 4 1 2 0 1 0 4 3 @ 1 2 0 4 1 0 3 4 のように処理する。

【 0 1 2 9 】

「 1 0 進数非省略書式」ハイフン (-) 挿入の場合は、ハイフンを照合前に削除処理し

10

20

30

40

50

てから前記の処理を行う。

【0130】

「10進数省略書式」ハイフン(-)挿入の場合は、「1234」の順番で基準となる4桁の数字列テンプレートを予めメモリに記憶しておき、これとハイフンで区切られた入力文字コード列を照合し、一致する数字を昇順に並び替えるが、入力文字コード列中に省略されて4桁の文字列テンプレートに一致しない数字がある場合には0を挿入して補う。その後ハイフンを削除する。この処理をハイフンで区切られた入力文字コード列ごとに繰り返す。

例えば、@412-143@ 124-134 12041034のように処理する。

【0131】

「16進数圧縮書式」の場合は、入力された文字コード列を16進数と2進数の対応表に照合して変換するか、処理装置組み込みの関数で変換する。

例えば、(16進数)DB (2進数)11011011のように処理する。

【0132】

「連想文字圧縮書式」の場合は、入力された文字コード列を連想文字コードと2進数の対応表に照合して変換する。例えば、(連想文字コード)KW (2進数)11011011のように処理する。

【0133】

文字データベースを表す図4のコード欄に品詞などの文法情報と語意分類などの意味情報をコード化して記憶することにより、同一レコードに同一のコードで記憶されている文字を、文法と意味の組み合わせ情報からも検索できる。

【0134】

たとえば前記6桁の分割コードに続けて7桁目に文法情報、8桁目に意味情報のコードを付加することとする。

【0135】

文法情報とは、名詞、動詞、形容詞といった品詞情報などを指す。

次に説明する実施形態では、名詞とそれ以外の品詞という情報を文法情報とするが、これに限定されず、たとえば主語と述語という構文情報を文法情報として使ってもよい。

【0136】

上記のように名詞とそれ以外に分類した場合は、7桁目の文法情報を10進数書式で表し、名詞を7、動詞や形容詞などの名詞以外の品詞を8、品詞を指定しない場合を0で表すこととする。

【0137】

文字データベース図4のたとえば「北」の6桁目までの字形情報を2進数非省略書式で表すと110000となる。「北」の品詞は名詞なので、7桁目の文法情報を付加したコードは1100007と表せる。

【0138】

文字データベース図4のたとえば「打」の6桁目までの字形情報を2進数非省略書式で表すと110000となる。「打」の品詞は動詞であるため、名詞以外の品詞なので、7桁目の文法情報を付加したコードは1100008と表せる。

【0139】

意味情報とは、人間に関する語、自然に関する語といった語意の分類情報などを指す。

次に説明する実施形態では、人間に関するか人間以外かという情報を意味情報とするが、そのほかたとえば動物と植物とそれ以外のもののように分類方法は任意である。

【0140】

たとえば、検索文字が人間に関する場合は8桁目に9、人間以外に関する場合は8桁目に0という情報を10進数書式で付加することにより、文字データベース図4の「北」は人間以外の自然に関する語なので11000070、「打」は人間に関する語なので11000089と表せ、検索時により詳細な識別が可能となる。

【0141】

10

20

30

40

50

このように、6桁目までの字形情報のみでコードを入力して検索すると前記「北」と「打」の2つの漢字は同一のコードなので検索時に識別できないが、7桁目の文法情報と8桁目の意味情報を付加するとコードが異なるので検索時に識別できる。

【図面の簡単な説明】

【0142】

【図1】本発明の機能構成ブロック図である。

【図2】本発明の検索処理フローチャートである。

【図3】本発明の文字の分割方法を示す説明図である。

【図4】本発明の文字データベース表である。

【図5】本発明の文字データベースのシフトJIS表記である。

10

【図6】本発明の10進数省略書式と連想文字圧縮書式の対照表である。

【図7】五筆字型のキー配列である。

【図8】ASCII配列のキー配列である。

【図9】五筆字型の入力例である。

【図10】中国語の同音漢字の例である。

【図11】中国語の同音漢字の分散率である。

【図12】中国語の同音漢字の4桁コードと5桁コードの分散率である。

【図13】中国語の同音漢字の6桁コードの分散率である。

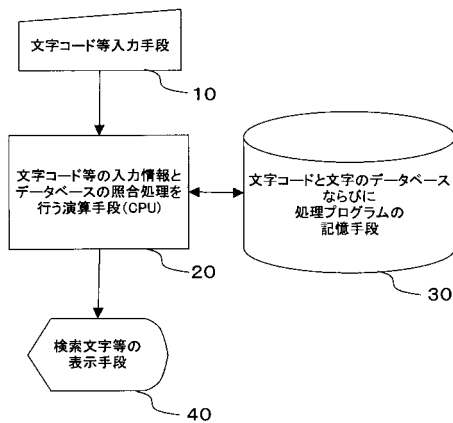
【符号の説明】

【0143】

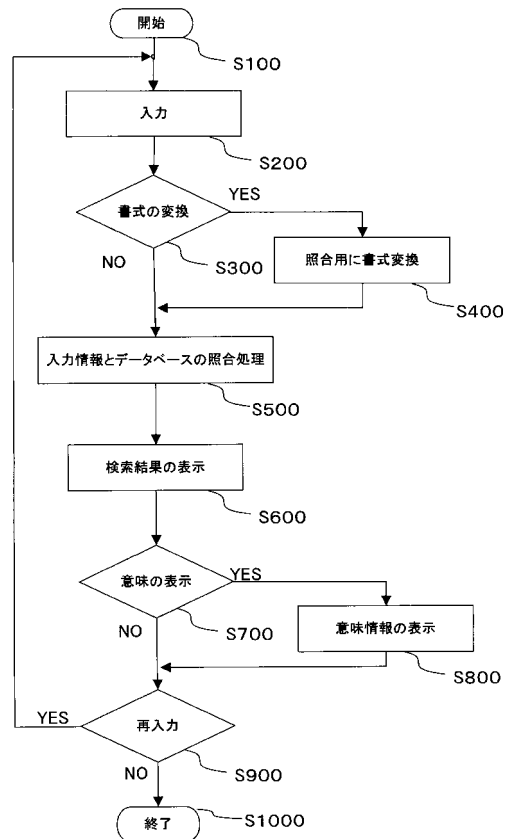
- 10 入力手段
- 20 演算手段
- 30 記憶手段
- 40 表示手段

20

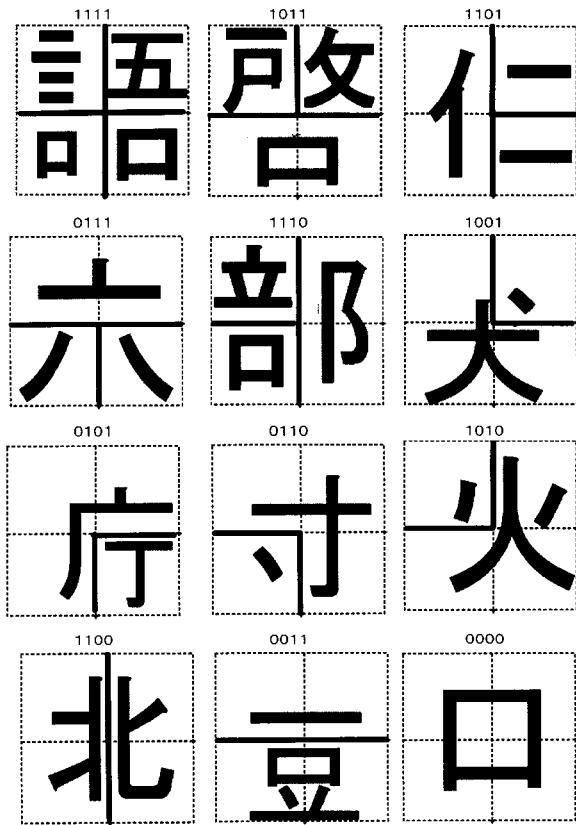
【図1】



【図2】



【 図 3 】



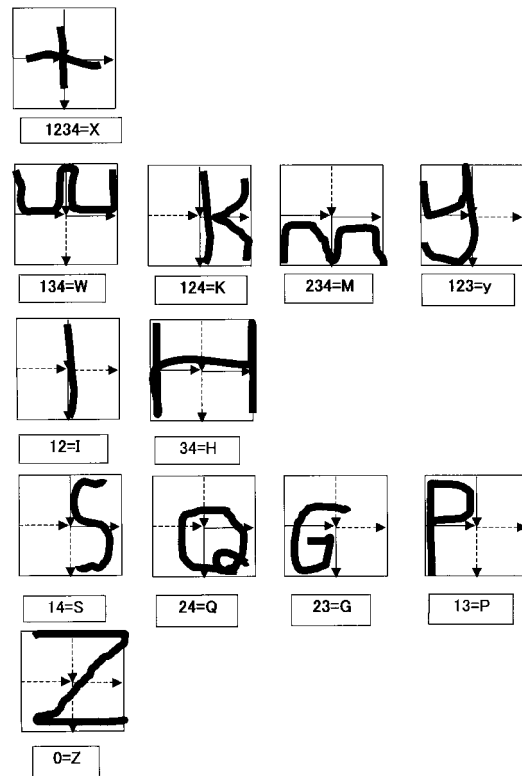
【 図 4 】

4分割コード	文字
...	...
1111	語
1111	競
...	...
1011	啓
1011	整
...	...
1101	仁
1101	暗
...	...
0111	六
0111	品
...	...
1110	部
1110	記
...	...
1001	犬
1001	起
...	...
0101	庁
0101	存
...	...
0110	寸
0110	可
...	...
1010	火
1010	半
...	...
1100	北
1100	打
...	...
0011	豆
0011	示
...	...
0000	口
0000	山
...	...

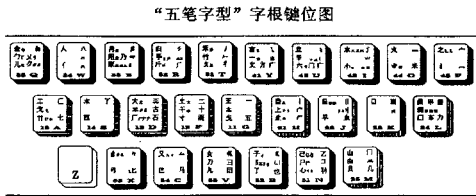
【 図 5 】

4分割コードのシフトJIS表記	文字のシフトJIS表記
...	...
31313131	8CEA
31313131	8BA3
...	...
31303131	8C5B
31303131	90AE
...	...
31313031	906D
31313031	88C3
...	...
30313131	985A
30313131	9569
...	...
31313130	9594
31313130	8B4C
...	...
31303031	8CA2
31303031	8B4E
...	...
30313031	92A1
30313031	91B6
...	...
30313130	90A1
30313130	89C2
...	...
31303130	89CE
31303130	94BC
...	...
31313030	966B
31313030	91C5
...	...
30303131	93A4
30303131	8EA6
...	...
30303030	8CFB
30303030	8E52
...	...

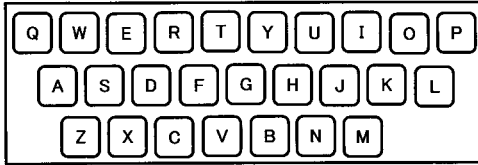
【 図 6 】



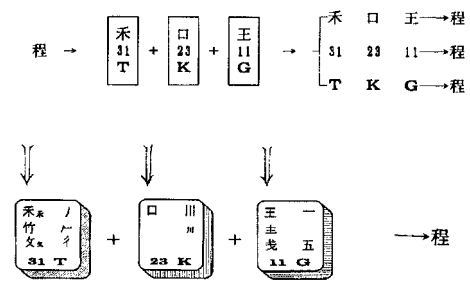
【图 7】



【图 8】



【图 9】



【图 10】

1. 《现代汉语词典》中有109个“YI”：

衣 依 依 一 壹 椅 漪 漪 医 揖 夥 噫 伊 伊 翼
 沂 宜 圮 夷 瘼 奠 奠 腴 赜 赜 赜 蛇 遗 仪 移 彝
 疑 疑 怡 怡 怡 怡 怡 迤 迤 迤 迤 蛟 蛟 蛟 乙
 乞 以 苾 矣 已 尾 醜 愈 愈 愈 愈 愈 愈 亦 奕
 裔 益 溢 溢 溢 溢 溢 溢 肄 肄 弋 抑 易 易 易
 邑 悒 悒 义 议 艾 刘 食 挾 挾 屹 屹 偈 偈 诣
 逸 肄 肄 疫 疫 忆 忆 艺 屹 译 译 译 译 译
 翊 翊 羿 羿

2. 《新华词典》中有 67 个“SHI”：

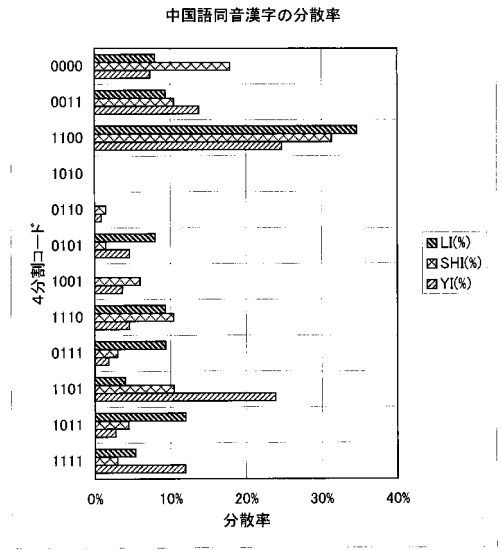
著 韶 师 嘘 失 夔 夔 夔 夔 夔 夔 夔 夔 夔 夔 夔 夔
 石 时 食 制 蚀 蚀 实 实 实 实 实 实 实 实 实 实 实
 式 示 士 世 世 世 世 世 世 世 世 世 世 世 世 世
 嗜 嗜 嗜 嗜 嗜 嗜 嗜 嗜 嗜 嗜 嗜 嗜 嗜 嗜 嗜 嗜 嗜
 侍 室 视 试 试 试 试 试

3. 国家标准 6763 个汉字中有 75 个“LI”：

哩 丽 丽 鹧 鹧 哩 哩 哩 哩 哩 哩 哩 哩 哩 哩 哩 哩
 黎 黎 黎 黎 黎 黎 黎 黎 黎 黎 黎 黎 黎 黎 黎 黎 黎
 漉 漉 力 荔 荔 荔 荔 荔 荔 荔 荔 荔 荔 荔 荔 荔
 妨 妨 立 莅 莅 笠 笠 吏 丽 丽 利 利 利 利 利 利
 庚 庚 栎 栎 砾 砾 砾 砾 栗 栗 栗 栗 栗 栗 栗 栗

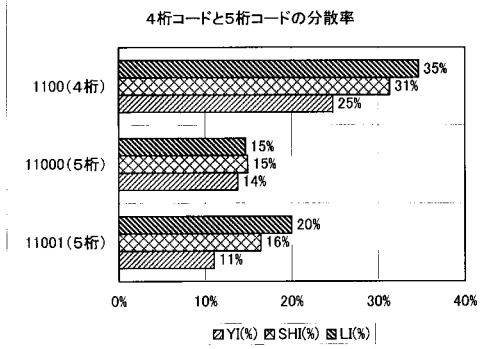
【图 11】

	YI	SHI	LI	YI(%)	SHI(%)	LI(%)
1111	13	2	4	12%	3%	5%
1011	3	3	9	3%	4%	12%
1101	26	7	3	24%	10%	4%
0111	2	2	7	2%	3%	9%
1110	5	7	7	5%	10%	9%
1001	4	4	0	4%	6%	0%
0101	5	1	6	5%	1%	8%
0110	1	1	0	1%	1%	0%
1010	0	0	0	0%	0%	0%
1100	27	21	26	25%	31%	35%
0011	15	7	7	14%	10%	9%
0000	8	12	6	7%	18%	8%
計	109	67	75	100%	100%	100%



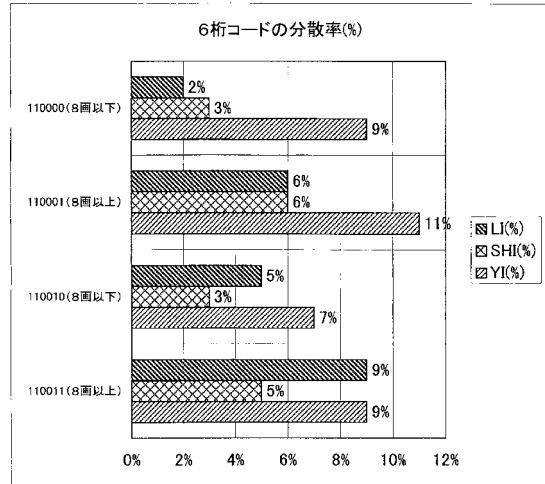
【 図 1 2 】

	YI	SHI	LI	YI(%)	SHI(%)	LI(%)
11001(5桁)	12	11	15	11%	16%	20%
11000(5桁)	15	10	11	14%	15%	15%
1100(4桁)	27	21	26	25%	31%	35%



【 図 1 3 】

		YI字数(%)	SHI字数(%)	LI字数(%)
110011(9画以上)	110011(8画以上)	6(7%)	8(9%)	6(4.0%)
11001*(8画重複)	110010(8画以下)	2(2%)	6(7%)	2(1.3%)
110010(7画以下)	110001(8画以上)	4(4%)	7(8%)	5(3%)
110001(9画以上)	110000(8画以下)	3(3%)	10(11%)	3(2.0%)
11000*(8画重複)		3(3%)	3(2.0%)	8(6%)
110000(7画以下)		5(6%)	8(9%)	4(3%)



	YI(%)	SHI(%)	LI(%)
110011(8画以上)	9%	5%	9%
110010(8画以下)	7%	3%	5%
110001(8画以上)	11%	6%	6%
110000(8画以下)	9%	3%	2%

【 手続補正書 】

【 提出日 】平成19年7月13日(2007.7.13)

【 手続補正 1 】

【 補正対象書類名 】特許請求の範囲

【 補正対象項目名 】全文

【 補正方法 】変更

【 補正の内容 】

【 特許請求の範囲 】

【 請求項 1 】

文字4分割コード等の入力を受け付けるための入力手段10と、検索文字の構成要素の間隙をコード化した文字4分割コードに対応する文字データベースや文字検索プログラムを記憶するための記憶手段30と、入力情報と文字データベースの照合を行うための演算手段20と、検索結果を表示するための表示手段40とを備えた検索装置における文字検索方法であって、

前記記憶手段30に文字データベースを記憶するに際し、文字の構成要素間に間隙がある場合は分割線が引け、間隙がない場合は分割線が引けないという判断基準に基づき、文字に対し縦方向、横方向の順で略十字形に分割線が引けるか否かを、文字の上、下、左、右の4つの部分ごとに順に判断し、分割線が引ける場合は数字の1、引けない場合は数字の0で表し、この数字を前記上、下、左、右の順に、4桁の数字の1桁目、2桁目、3桁目、4桁目に対応するそれぞれの桁に割り当てることで文字をコード化して文字4分割コードとし、該文字4分割コードとそれに対応する文字または文字画像、多言語、動画ファイルをデータベースとして分類して記憶しておき、

前記入力手段10が、前記文字4分割コードの入力を受け付けるステップと、前記演算手段20が、前記入力を受け付けた文字4分割コードと前記記憶手段30に記憶された文字4分割コードとを照合し、これらの文字4分割コードが合致した場合に、該文

字 4 分割コードに対応する文字または文字画像、多言語、動画ファイルを前記表示手段 40 に表示するステップと、により構成され、

文字や文字の構成要素に関する知識のない者が入力した文字 4 分割コードから対応する文字または文字画像を検索表示することを可能とし、得られた文字または文字画像の意味を多言語や動画で理解することも可能とした文字検索方法。

【請求項 2】

文字 4 分割コード及び文字発音情報の入力を受け付けるための入力手段 10 と、検索文字の構成要素の間隙をコード化した文字 4 分割コード及び該文字発音情報の組み合わせに対応する文字データベースや文字検索プログラムを記憶するための記憶手段 30 と、入力情報と文字データベースの照合を行うための演算手段 20 と、検索結果を表示するための表示手段 40 とを備えた検索装置における文字検索方法であって、

前記記憶手段 30 に文字データベースを記憶するに際し、文字の構成要素間に間隙がある場合は分割線が引け、間隙がない場合は分割線が引けないという判断基準に基づき、文字に対し縦方向、横方向の順で略十文字形に分割線が引けるか否かを、文字の上、下、左、右の 4 つの部分ごとに順に判断し、分割線が引ける場合は数字の 1、引けない場合は数字の 0 で表し、この数字を前記上、下、左、右の順に、4 桁の数字の 1 桁目、2 桁目、3 桁目、4 桁目に対応するそれぞれの桁に割り当てることで文字をコード化して文字 4 分割コードとし、該文字 4 分割コードの直後に文字発音情報をアルファベットで併記することにより前記文字 4 分割コード及び該文字発音情報の組み合わせに対応する文字または文字画像、多言語、動画ファイルをデータベースとして分類して記憶しておく、

前記入力手段 10 が、前記文字 4 分割コード及び該文字発音情報の組み合わせの入力を受け付けるステップと、

前記演算手段 20 が、前記入力を受け付けた文字 4 分割コード及び該文字発音情報の組み合わせと前記記憶手段 30 に記憶された文字 4 分割コード及び該文字発音情報の組み合わせとを照合し、これらの文字 4 分割コード及び該文字発音情報の組み合わせが合致した場合に、該文字 4 分割コード及び該文字発音情報の組み合わせに対応する文字または文字画像、多言語、動画ファイルを前記表示手段 40 に表示するステップと、により構成され、文字や文字の発音知識を持つ者が入力した文字 4 分割コード及び該文字発音情報の組み合わせから対応する文字または文字画像を検索表示することを可能とし、得られた文字または文字画像の意味を多言語や動画で理解することも可能とした文字検索方法。

【請求項 3】

文字 4 分割コード及び文字文法意味情報の入力を受け付けるための入力手段 10 と、検索文字の構成要素の間隙をコード化した文字 4 分割コード及び該文字文法意味情報に対応する文字データベースや文字検索プログラムを記憶するための記憶手段 30 と、入力情報と文字データベースの照合を行うための演算手段 20 と、検索結果を表示するための表示手段 40 とを備えた検索装置における文字検索方法であって、

前記記憶手段 30 に文字データベースを記憶するに際し、文字の構成要素間に間隙がある場合は分割線が引け、間隙がない場合は分割線が引けないという判断基準に基づき、文字に対し縦方向、横方向の順で略十文字形に分割線が引けるか否かを、文字の上、下、左、右の 4 つの部分ごとに順に判断し、分割線が引ける場合は数字の 1、引けない場合は数字の 0 で表し、この数字を前記上、下、左、右の順に、8 桁の数字の 1 桁目、2 桁目、3 桁目、4 桁目の順に対応するそれぞれの桁に割り当てることで文字をコード化して文字 4 分割コードとし、該文字の分割線と異なる箇所さらに分割可能な構成要素間の間隙があるか否かを 5 桁目に数字で表し、該文字の構成要素の多寡を 6 桁目に数字で表すことによりコード化し、このコードに続けて名詞かそれ以外かという文法情報を 7 桁目に数字で表し、人間かそれ以外かという意味情報を数字で 8 桁目に表すことでコード化し、前記文字 4 分割コード及び該文字に関する該文字文法情報と意味情報のコードを組み合わせるとして 8 桁のコードとし、それに対応する文字または文字画像、多言語、動画ファイルをデータベースとして分類して記憶しておく、

前記入力手段 10 が、前記 8 桁のコードを受け付けるステップと、

前記演算手段 20 が、前記入力を受け付けた前記 8 桁のコードと前記記憶手段 30 に記憶された前記 8 桁のコードとを照合し、これらの前記 8 桁のコードが合致した場合に、該 8 桁のコードに対応する文字または文字画像、多言語、動画ファイルを前記表示手段 40 に表示するステップと、により構成され、

文字に関する文法や意味の知識を持つ者が入力した前記 8 桁のコードから対応する文字または文字画像を検索表示することを可能とし、得られた文字または文字画像の意味を多言語や動画で理解することも可能とした文字検索方法。

【手続補正 2】

【補正対象書類名】明細書

【補正対象項目名】0006

【補正方法】変更

【補正の内容】

【0006】

本発明の要旨とするところは、文字 4 分割コード等の入力を受け付けるための入力手段 10 と、検索文字の構成要素の間隙をコード化した文字 4 分割コードに対応する文字データベースや文字検索プログラムを記憶するための記憶手段 30 と、入力情報と文字データベースの照合を行うための演算手段 20 と、検索結果を表示するための表示手段 40 とを備えた検索装置における文字検索方法であって、

前記記憶手段 30 に文字データベースを記憶するに際し、文字の構成要素間に間隙がある場合は分割線が引け、間隙がない場合は分割線が引けないという判断基準に基づき、文字に対し縦方向、横方向の順で略十文字形に分割線が引けるか否かを、文字の上、下、左、右の 4 つの部分ごとに順に判断し、分割線が引ける場合は数字の 1、引けない場合は数字の 0 で表し、この数字を前記上、下、左、右の順に、4 桁の数字の 1 桁目、2 桁目、3 桁目、4 桁目に対応するそれぞれの桁に割り当てることで文字をコード化して文字 4 分割コードとし、該文字 4 分割コードとそれに対応する文字または文字画像、多言語、動画ファイルをデータベースとして分類して記憶しておく、

前記入力手段 10 が、前記文字 4 分割コードの入力を受け付けるステップと、前記演算手段 20 が、前記入力を受け付けた文字 4 分割コードと前記記憶手段 30 に記憶された文字 4 分割コードとを照合し、これらの文字 4 分割コードが合致した場合に、該文字 4 分割コードに対応する文字または文字画像、多言語、動画ファイルを前記表示手段 40 に表示するステップと、により構成され、

文字や文字の構成要素に関する知識のない者が入力した文字 4 分割コードから対応する文字または文字画像を検索表示することを可能とし、得られた文字または文字画像の意味を多言語や動画で理解することも可能とした文字検索方法にある。

【手続補正 3】

【補正対象書類名】明細書

【補正対象項目名】0007

【補正方法】変更

【補正の内容】

【0007】

また本発明の要旨とするところは、文字 4 分割コード及び文字発音情報の入力を受け付けるための入力手段 10 と、検索文字の構成要素の間隙をコード化した文字 4 分割コード及び該文字発音情報の組み合わせに対応する文字データベースや文字検索プログラムを記憶するための記憶手段 30 と、入力情報と文字データベースの照合を行うための演算手段 20 と、検索結果を表示するための表示手段 40 とを備えた検索装置における文字検索方法であって、

前記記憶手段 30 に文字データベースを記憶するに際し、文字の構成要素間に間隙がある場合は分割線が引け、間隙がない場合は分割線が引けないという判断基準に基づき、文字に対し縦方向、横方向の順で略十文字形に分割線が引けるか否かを、文字の上、下、左、右の 4 つの部分ごとに順に判断し、分割線が引ける場合は数字の 1、引けない場合は数

字の 0 で表し、この数字を前記上、下、左、右の順に、4 桁の数字の 1 桁目、2 桁目、3 桁目、4 桁目に対応するそれぞれの桁に割り当てることで文字をコード化して文字 4 分割コードとし、該文字 4 分割コードの直後に文字発音情報をアルファベットで併記することにより前記文字 4 分割コード及び該文字発音情報の組み合わせに対応する文字または文字画像、多言語、動画ファイルをデータベースとして分類して記憶しておく、

前記入力手段 10 が、前記文字 4 分割コード及び該文字発音情報の組み合わせの入力を受け付けるステップと、

前記演算手段 20 が、前記入力を受け付けた文字 4 分割コード及び該文字発音情報の組み合わせと前記記憶手段 30 に記憶された文字 4 分割コード及び該文字発音情報の組み合わせとを照合し、これらの文字 4 分割コード及び該文字発音情報の組み合わせが合致した場合に、該文字 4 分割コード及び該文字発音情報の組み合わせに対応する文字または文字画像、多言語、動画ファイルを前記表示手段 40 に表示するステップと、により構成され、

文字や文字の発音知識を持つ者が入力した文字 4 分割コード及び該文字発音情報の組み合わせから対応する文字または文字画像を検索表示することを可能とし、得られた文字または文字画像の意味を多言語や動画で理解することも可能とした文字検索方法にある。

また、本発明の要旨とするところは、文字 4 分割コード及び文字文法意味情報の入力を受け付けるための入力手段 10 と、検索文字の構成要素の間隙をコード化した文字 4 分割コード及び該文字文法意味情報に対応する文字データベースや文字検索プログラムを記憶するための記憶手段 30 と、入力情報と文字データベースの照合を行うための演算手段 20 と、検索結果を表示するための表示手段 40 とを備えた検索装置における文字検索方法であって、

前記記憶手段 30 に文字データベースを記憶するに際し、文字の構成要素間に間隙がある場合は分割線が引け、間隙がない場合は分割線が引けないという判断基準に基づき、文字に対し縦方向、横方向の順で略十字字形に分割線が引けるか否かを、文字の上、下、左、右の 4 つの部分ごとに順に判断し、分割線が引ける場合は数字の 1、引けない場合は数字の 0 で表し、この数字を前記上、下、左、右の順に、8 桁の数字の 1 桁目、2 桁目、3 桁目、4 桁目の順に対応するそれぞれの桁に割り当てることで文字をコード化して文字 4 分割コードとし、該文字の分割線と異なる箇所さらに分割可能な構成要素間の間隙があるか否かを 5 桁目に数字で表し、該文字の構成要素の多寡を 6 桁目に数字で表すことによりコード化し、このコードに続けて名詞かそれ以外かという文法情報を 7 桁目に数字で表し、人間かそれ以外かという意味情報を数字で 8 桁目に表すことでコード化し、前記文字 4 分割コード及び該文字に関する該文字文法情報と意味情報のコードを組み合わせるとして 8 桁のコードとし、それに対応する文字または文字画像、多言語、動画ファイルをデータベースとして分類して記憶しておく、

前記入力手段 10 が、前記 8 桁のコードを受け付けるステップと、

前記演算手段 20 が、前記入力を受け付けた前記 8 桁のコードと前記記憶手段 30 に記憶された前記 8 桁のコードとを照合し、これらの前記 8 桁のコードが合致した場合に、該 8 桁のコードに対応する文字または文字画像、多言語、動画ファイルを前記表示手段 40 に表示するステップと、により構成され、

文字に関する文法や意味の知識を持つ者が入力した前記 8 桁のコードから対応する文字または文字画像を検索表示することを可能とし、得られた文字または文字画像の意味を多言語や動画で理解することも可能とした文字検索方法にある。