(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2007-228582
(P2007-228582A)
(43)公開日　平成19年9月6日(2007.9.6)

(51)Int.Cl.			F I			テーマコード（参考）
H03M	7/40	(2006.01)	H03M	7/40		5C059
H04N	7/26	(2006.01)	H04N	7/13	Z	5J064

審査請求 未請求　請求項の数 14　ＯＬ　（全 17 頁）

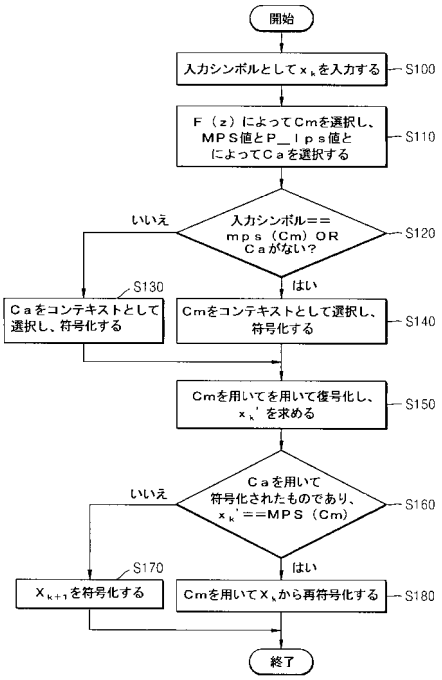
(21)出願番号　特願2007-40925(P2007-40925)
(22)出願日　平成19年2月21日(2007.2.21)
(31)優先権主張番号　10-2006-0017239
(32)優先日　平成18年2月22日(2006.2.22)
(33)優先権主張国　韓国(KR)

(71)出願人　390019839
三星電子株式会社
Ｓａｍｓｕｎｇ　Ｅｌｅｃｔｒｏｎｉｃｓ　Ｃｏ．，Ｌｔｄ．
大韓民国京畿道水原市八達区梅灘洞４１６番地
(74)代理人　100070150
弁理士　伊東　忠彦
(74)代理人　100091214
弁理士　大貫　進介
(74)代理人　100107766
弁理士　伊東　忠重
(72)発明者　▲ちょう▼　榮　佑
大韓民国ソウル特別市冠岳区奉天７洞１５１０－１６番地

最終頁に続く

(54)【発明の名称】圧縮率の向上のために改善されたコンテキストモデル選択を使用するＣＡＢＡＣ符号化方法及び装置、並びにＣＡＢＡＣ復号化方法及び装置

(57)【要約】　（修正有）
【課題】圧縮率の向上のために改善されたコンテキストモデル選択を使用するＣＡＢＡＣ符号・復号化方法及び装置の提供。
【解決手段】以前の入力シンボルに対する統計に基づく第１コンテキストモデルと、第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する第２コンテキストモデルとを選択する段階と、現在の入力シンボルが第１コンテキストモデルのＭＰＳと同じ値を有するか、第２コンテキストモデルを選択できない場合、第１コンテキストモデルを用いて入力シンボルを符号化する段階と、現在の入力シンボルが第１コンテキストモデルのＭＰＳ値と異なる値を有し、第２コンテキストモデルを選択できる場合、第２コンテキストモデルを用いて入力シンボルを符号化する段階とを含むＣＡＢＡＣ符号化方法である。
【選択図】図１



【特許請求の範囲】
【請求項１】
以前の入力シンボルに対する統計に基づく第１コンテキストモデルと、前記第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する第２コンテキストモデルとを選択する段階と、
現在の入力シンボルが前記第１コンテキストモデルのＭＰＳと同じ値を有するか、第２コンテキストモデルを選択できない場合、前記第１コンテキストモデルを用いて前記入力シンボルを符号化する段階と、
前記現在の入力シンボルが前記第１コンテキストモデルのＭＰＳ値と異なる値を有し、第２コンテキストモデルを選択できる場合、前記第２コンテキストモデルを用いて前記入力シンボルを符号化する段階と、を含むことを特徴とするコンテキスト基盤の適応二進算術コーディング（ＣＡＢＡＣ）符号化方法。
【請求項２】
前記第１コンテキストモデルまたは第２コンテキストモデルを用いて符号化された入力シンボルを前記第１コンテキストモデルを用いて復号化する段階と、
前記符号化された入力シンボルが前記第２コンテキストモデルを用いて符号化されたものであり、前記第１コンテキストモデルを用いて復号化された値が前記第１コンテキストモデルのＭＰＳ値と同じ場合、前記現在の入力シンボルを前記第１コンテキストモデルを用いて符号化する段階をさらに含むことを特徴とする請求項１に記載のコンテキスト基盤の適応二進算術コーディング（ＣＡＢＡＣ）符号化方法。
【請求項３】
前記第１コンテキストモデル及び第２コンテキストモデルを選択する段階は、
前記第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する一つ以上のコンテキストモデルのうち、ＬＰＳの発生確率値が最も小さいものを前記第２コンテキストモデルとして選択する段階を含むことを特徴とする請求項１に記載のコンテキスト基盤の適応二進算術コーディング（ＣＡＢＡＣ）符号化方法。
【請求項４】
以前の入力シンボルに対する統計に基づく第１コンテキストモデルと、前記第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する第２コンテキストモデルとを選択するコンテキストモデル選択部と、
現在の入力シンボルが前記第１コンテキストモデルのＭＰＳと同じ値を有するか、第２コンテキストモデルを選択できない場合、前記第１コンテキストモデルを用いて前記入力シンボルを符号化する第１符号化部と、
前記現在の入力シンボルが前記第１コンテキストモデルのＭＰＳ値と異なる値を有し、第２コンテキストモデルを選択できる場合、前記第２コンテキストモデルを用いて前記入力シンボルを符号化する第２符号化部と、を備えることを特徴とするコンテキスト基盤の適応二進算術コーディング（ＣＡＢＡＣ）符号化装置。
【請求項５】
前記第１コンテキストモデルまたは第２コンテキストモデルを用いて符号化された入力シンボルを、前記第１コンテキストモデルを用いて復号化するシンボル復号化部と、
前記符号化された入力シンボルが前記第２コンテキストモデルを用いて符号化されたものであり、前記第１コンテキストモデルを用いて復号化された値が前記第１コンテキストモデルのＭＰＳ値と同じ場合、前記現在の入力シンボルを前記第１コンテキストモデルを用いて符号化する第３符号化部をさらに備えることを特徴とする請求項４に記載のコンテキスト基盤の適応二進算術コーディング（ＣＡＢＡＣ）符号化装置。
【請求項６】
コンテキストモデル選択部は、
前記第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する一つ以上のコンテキストモデルのうち、ＬＰＳの発生確率値が最も小さいものを前記第２コンテキストモデルとして選択することを特徴とする請求項１に記載のコンテキスト基盤の適応二進算術コーデ

ィング（ＣＡＢＡＣ）符号化装置。

【請求項７】

以前の復号化されたシンボルに対する統計に基づく第１コンテキストモデルと、前記第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する第２コンテキストモデルとを選択する段階と、

符号化されたシンボルを前記第１コンテキストモデルを用いて復号化することで、第１復号シンボルを求める段階と、

前記第１復号シンボルが前記第１コンテキストモデルのＭＰＳと同じ値を有する場合、前記第１復号シンボルをＣＡＢＡＣ復号化の結果値として選択する段階と、

前記第１復号シンボルが前記第１コンテキストモデルのＭＰＳと異なる値を有する場合、前記符号化されたシンボルを前記第２コンテキストモデルを用いて復号化することで、第２復号シンボルを求める段階とを含むことを特徴とするコンテキスト基盤の適応二進算術コーディング（ＣＡＢＡＣ）復号化方法。

【請求項８】

前記第２復号シンボルが前記第２コンテキストモデルのＭＰＳと異なる値を有する場合、前記第１復号シンボルをＣＡＢＡＣ復号化の結果値として選択する段階と、

前記第２復号シンボルが前記第２コンテキストモデルのＭＰＳと同じ値を有する場合、前記第２復号シンボルを前記第２コンテキストモデルを用いて符号化し、その結果を前記第１コンテキストモデルを用いて復号化することで、第３復号シンボルを求める段階と、

前記第３復号シンボルが前記第２コンテキストモデルのＭＰＳと同じ値を有する場合、前記第２復号シンボルを前記ＣＡＢＡＣ復号化の結果値として選択する段階と、

前記第３復号シンボルが前記第２コンテキストモデルのＭＰＳと異なる値を有する場合、前記第１復号シンボルを前記ＣＡＢＡＣ復号化の結果値として選択する段階と、をさらに含むことを特徴とする請求項７に記載のコンテキスト基盤の適応二進算術コーディング（ＣＡＢＡＣ）復号化方法。

【請求項９】

前記第１コンテキストモデル及び第２コンテキストモデルを選択する段階は、

前記第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する一つ以上のコンテキストモデルのうち、ＬＰＳの発生確率値が最も小さいものを前記第２コンテキストモデルとして選択する段階を含むことを特徴とする請求項７に記載のコンテキスト基盤の適応二進算術コーディング（ＣＡＢＡＣ）復号化方法。

【請求項１０】

以前の復号化されたシンボルに対する統計に基づく第１コンテキストモデルと、前記第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する第２コンテキストモデルとを選択するコンテキストモデル選択部と、

符号化されたシンボルを前記第１コンテキストモデルを用いて復号化することで、第１復号シンボルを求める第１復号化部と、

前記第１復号シンボルが前記第１コンテキストモデルのＭＰＳと同じ値を有する場合、前記第１復号シンボルをＣＡＢＡＣ復号化の結果値として選択する第１結果値選択部と、

前記第１復号シンボルが前記第１コンテキストモデルのＭＰＳと異なる値を有する場合、前記符号化されたシンボルを前記第２コンテキストモデルを用いて復号化することで、第２復号シンボルを求める第２復号化部と、を備えることを特徴とするコンテキスト基盤の適応二進算術コーディング（ＣＡＢＡＣ）復号化装置。

【請求項１１】

前記第２復号シンボルが前記第２コンテキストモデルのＭＰＳと異なる値を有する場合、前記第１復号シンボルをＣＡＢＡＣ復号化の結果値として選択する第２結果値選択部と、

前記第２復号シンボルが前記第２コンテキストモデルのＭＰＳと同じ値を有する場合、前記第２復号シンボルを前記第２コンテキストモデルを用いて符号化し、その結果を前記第１コンテキストモデルを用いて復号化することで、第３復号シンボルを求める第３復号

化部と、

前記第３復号シンボルが前記第２コンテキストモデルのＭＰＳと同じ値を有する場合、前記第２復号シンボルを前記ＣＡＢＡＣ復号化の結果値として選択する第３結果値選択部と、

前記第３復号シンボルが前記第２コンテキストモデルのＭＰＳと異なる値を有する場合、前記第１復号シンボルを前記ＣＡＢＡＣ復号化の結果値として選択する第４結果値選択部をさらに含むことを特徴とする請求項１０に記載のコンテキスト基盤の適応二進算術コーディング（ＣＡＢＡＣ）復号化装置。

【請求項１２】

コンテキストモデル選択部は、

前記第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する一つ以上のコンテキストモデルのうち、ＬＰＳの発生確率値が最も小さいものを前記第２コンテキストモデルとして選択することを特徴とする請求項１０に記載のコンテキスト基盤の適応二進算術コーディング（ＣＡＢＡＣ）復号化装置。

【請求項１３】

以前の入力シンボルに対する統計に基づく第１コンテキストモデルと、前記第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する第２コンテキストモデルとを選択する段階と、

現在の入力シンボルが前記第１コンテキストモデルのＭＰＳと同じ値を有するか、第２コンテキストモデルを選択できない場合、前記第１コンテキストモデルを用いて前記入力シンボルを符号化する段階と、

前記現在の入力シンボルが前記第１コンテキストモデルのＭＰＳ値と異なる値を有し、第２コンテキストモデルを選択できる場合、前記第２コンテキストモデルを用いて前記入力シンボルを符号化する段階と、をコンピュータで実行させるためのプログラムを記録したコンピュータで読み取り可能な記録媒体。

【請求項１４】

以前の復号化されたシンボルに対する統計に基づく第１コンテキストモデルと、前記第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する第２コンテキストモデルとを選択する段階と、

符号化されたシンボルを前記第１コンテキストモデルを用いて復号化することで、第１復号シンボルを求める段階と、

前記第１復号シンボルが前記第１コンテキストモデルのＭＰＳと同じ値を有する場合、前記第１復号シンボルをＣＡＢＡＣ復号化の結果値として選択する段階と、

前記第１復号シンボルが前記第１コンテキストモデルのＭＰＳと異なる値を有する場合、前記符号化されたシンボルを前記第２コンテキストモデルを用いて復号化することで、第２復号シンボルを求める段階と、をコンピュータで実行させるためのプログラムを記録したコンピュータで読み取り可能な記録媒体。

【発明の詳細な説明】

【技術分野】

【０００１】

本発明は、コンテキスト基盤の適応二進算術コーディング（Ｃｏｎｔｅｘｔ－ｂａｓｅｄ Ａｄａｐｔｉｖｅ Ｂｉｎａｒｙ Ａｒｉｔｈｍｅｔｉｃ Ｃｏｄｉｎｇ：ＣＡＢＡＣ）に係り、特に、圧縮率の向上のために改善されたコンテキストモデル選択を使用するＣＡＢＡＣ符号化及び復号化に関する。

【背景技術】

【０００２】

ＣＡＢＡＣは、Ｈ．２６４のような動映像圧縮に使われるアルゴリズムであって、以前に入力されたデータによってコンテキストを選択する部分、二進算術コーディングを行う部分、及び統計情報を更新する部分で構成される。コンテキストは、シンボルの確率をさらに正確に予測するために、以前に生じたシンボルによって定義される統計情報収集環境

である。

【０００３】

ＣＡＢＡＣの圧縮率は、コンテキストモデル方法によって決定される。コンテキストモデルをＬＰＳ（Ｌｅａｓｔ　Ｐｒｏｂａｂｌｅ　Ｓｙｍｂｏｌ）値が低くなるように設定することが有利である。しかし、従来の技術では、コンテキストモデルで予想した値が入力シンボルとなれば、コーディングの効率、すなわち圧縮率が高くなるが、コンテキストモデルで予想しない値が入力シンボルとなれば、コーディングの効率が低くなるという問題点があった。

【発明の開示】

【発明が解決しようとする課題】

【０００４】

本発明が達成しようとする技術的課題は、入力シンボルによってコンテキストモデルを選択して、コーディングの圧縮率を向上させうるＣＡＢＡＣ符号化方法及び装置、そして前記ＣＡＢＡＣ符号化方法をコンピュータで実行させるためのプログラムを記録したコンピュータで読み取り可能な記録媒体を提供することである。

【０００５】

本発明が達成しようとする他の技術的課題は、前記ＣＡＢＡＣ符号化により符号化されたシンボルを復号化するためのＣＡＢＡＣ復号化方法及び装置、そして前記ＣＡＢＡＣ復号化方法をコンピュータで実行させるためのプログラムを記録したコンピュータで読み取り可能な記録媒体を提供することである。

【課題を解決するための手段】

【０００６】

前記技術的課題を達成するためのＣＡＢＡＣ符号化方法は、以前の入力シンボルに対する統計に基づく第１コンテキストモデルと、前記第１コンテキストモデルのＭＰＳ（Ｍｏｓｔ　Ｐｒｏｂａｂｌｅ　Ｓｙｍｂｏｌ）と異なるＭＰＳ値を有する第２コンテキストモデルとを選択する段階と、現在の入力シンボルが前記第１コンテキストモデルのＭＰＳと同じ値を有するか、第２コンテキストモデルを選択できない場合、前記第１コンテキストモデルを用いて前記入力シンボルを符号化する段階と、前記現在の入力シンボルが前記第１コンテキストモデルのＭＰＳ値と異なる値を有し、第２コンテキストモデルを選択できる場合、前記第２コンテキストモデルを用いて前記入力シンボルを符号化する段階とを含むことが望ましい。

【０００７】

前記技術的課題を達成するためのＣＡＢＡＣ符号化装置は、以前の入力シンボルに対する統計に基づく第１コンテキストモデルと、前記第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する第２コンテキストモデルとを選択するコンテキストモデル選択部と、現在の入力シンボルが前記第１コンテキストモデルのＭＰＳと同じ値を有するか、第２コンテキストモデルを選択できない場合、前記第１コンテキストモデルを用いて前記入力シンボルを符号化する第１符号化部と、前記現在の入力シンボルが前記第１コンテキストモデルのＭＰＳ値と異なる値を有し、第２コンテキストモデルを選択できる場合、前記第２コンテキストモデルを用いて前記入力シンボルを符号化する第２符号化部とを備えることが望ましい。

【０００８】

前記他の技術的課題を達成するためのＣＡＢＡＣ復号化方法は、以前の復号化されたシンボルに対する統計に基づく第１コンテキストモデルと、前記第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する第２コンテキストモデルとを選択する段階と、符号化されたシンボルを前記第１コンテキストモデルを用いて復号化することで、第１復号シンボルを求める段階と、前記第１復号シンボルが前記第１コンテキストモデルのＭＰＳと同じ値を有する場合、前記第１復号シンボルをＣＡＢＡＣ復号化の結果値として選択する段階と、前記第１復号シンボルが前記第１コンテキストモデルのＭＰＳと異なる値を有する場合、前記符号化されたシンボルを前記第２コンテキストモデルを用いて復号化することで

、第２復号シンボルを求める段階とを含むことが望ましい。

【０００９】

前記他の技術的課題を達成するためのＣＡＢＡＣ復号化装置は、以前の復号化されたシンボルに対する統計に基づく第１コンテキストモデルと、前記第１コンテキストモデルのＭＰＳと異なるＭＰＳ値を有する第２コンテキストモデルとを選択するコンテキストモデル選択部と、符号化されたシンボルを前記第１コンテキストモデルを用いて復号化することで、第１復号シンボルを求める第１復号化部と、前記第１復号シンボルが前記第１コンテキストモデルのＭＰＳと同じ値を有する場合、前記第１復号シンボルをＣＡＢＡＣ復号化の結果値として選択する第１結果値選択部と、前記第１復号シンボルが前記第１コンテキストモデルのＭＰＳと異なる値を有する場合、前記符号化されたシンボルを前記第２コンテキストモデルを用いて復号化することで、第２復号シンボルを求める第２復号化部とを備えることが望ましい。

【発明の効果】

【００１０】

本発明によるＣＡＢＡＣ符号化方法及び装置、並びにＣＡＢＡＣ復号化方法及び装置によれば、以前の入力シンボルに対する統計に基づく一般的なコンテキスト選択以外に、現在の入力シンボルを用いてコンテキスト選択を追加で行うことにより、符号化の圧縮率を向上させることができる。また、符号化過程で検査用の復号化過程を予め行ってアンビギュイティな場合を除去するので、復号化過程で元の入力データをアンビギュイティな場合なしに復元できる。

【発明を実施するための最良の形態】

【００１１】

以下、添付された図面を参照しつつ、本発明によるＣＡＢＡＣ符号化方法及び装置、並びにＣＡＢＡＣ復号化方法及び装置について詳細に説明する。

【００１２】

図１は、本発明の一実施形態によるＣＡＢＡＣ符号化方法の各段階を示すフローチャートである。

【００１３】

まず、現在の入力シンボルとしてx_kが入力される（Ｓ１００）。

【００１４】

関数Ｆ（ｚ）によって第１コンテキストモデルＣｍを選択し、ＭＰＳ値とＬＰＳの発生確率であるＰ＿ｌｐｓ値とによって第２コンテキストモデルＣａを選択する（Ｓ１１０）。関数Ｆ（ｚ）は、モデリング関数である。関数Ｆ（ｚ）は、以前の入力シンボルに対する統計に基づいてコンテキストモデルを決定する。

【００１５】

第２コンテキストモデルＣａを選択する方法は、次の通りである。まず、ＣａとＣｍとは互いに異なるＭＰＳ値を有さなければならない。例えば、ＣｍのＭＰＳ値が０である場合、ＭＰＳ値として１を有するコンテキストモデルのうち一つをＣａとして選択する。ＣｍのＭＰＳ値と異なるＭＰＳ値を有するコンテキストモデルが複数個である場合、そのうちＰ＿ｌｐｓ値が最も小さいコンテキストモデルを選択する。もし、Ｐ＿ｌｐｓ値が最も小さいものが複数個である場合、順序が一番になるコンテキストモデルを選択できる。実施例によって、コンテキストモデルの順序は、コンテキストを示すインデックスの値によって決定される。

【００１６】

入力シンボルx_kがＣｍのＭＰＳと同じ値を有するか、第２コンテキストモデルＣａを選択できない場合、従来の方式によってＣｍを用いて入力シンボルを符号化する（Ｓ１４０）。

【００１７】

第２コンテキストモデルを選択できない場合の例としては、選択できるコンテキストモデルが全てＣｍと同じＭＰＳ値を有する場合を考えられる。

【００１８】
入力シンボルx_kがＣｍのＭＰＳと異なる値を有し、第２コンテキストモデルＣａを選択できる場合、Ｃｍを用いて入力シンボルを符号化すると、圧縮効率が低くなる。したがって、Ｃａを用いて入力シンボルを符号化する（Ｓ１３０）。
【００１９】
このように、入力シンボル値がＣｍのＭＰＳ値と同じ場合、Ｃｍを用いて符号化し、入力シンボル値がＣｍのＭＰＳ値と異なる場合、Ｃａを用いて符号化する。このような方式によりコーディングの圧縮効率を高めることができる。
【００２０】
しかし、このような方式によれば、復号化過程でどのコンテキストモデルを用いて復号化しなければならないかアンビギュイティ（ａｍｂｉｇｕｉｔｙ）な場合が発生しうる。このような場合を本発明においてアンビギュイティな場合の発生であると呼ぶ。
【００２１】
本発明では、アンビギュイティな場合の発生を防止するために、ＣＡＢＡＣ符号化部で検査用復号化を行うことでアンビギュイティな場合があるか検査し、前記Ｓ１３０段階によって符号化された値にアンビギュイティな場合があると判断される場合、入力シンボルを第１コンテキストモデルＣｍを用いて符号化してアンビギュイティな場合を除去する。
【００２２】
このために、まず符号化された入力シンボルを第１コンテキストモデルＣｍを用いて復号化してx_k'を求める（Ｓ１５０）。
【００２３】
もし、符号化された入力シンボルがＣａを用いて符号化されたものであり、x_k'がＣｍのＭＰＳ値と同じ値を有するならば、これはアンビギュイティな場合を有することである。すなわち、復号化過程で符号化された入力シンボルをＣｍを用いて復号化する場合、ＣｍのＭＰＳ値が得られるので、復号化部では、入力シンボルがＣｍを用いて符号化されたと認識する可能性が存在する。したがって、このような場合には、圧縮効率が少し低くなっても入力シンボルを再びＣｍを用いて符号化しなければならない（Ｓ１８０）。
【００２４】
Ｃｍを用いて復号化された値がＣｍのＭＰＳと異なる値を有するならば、アンビギュイティな場合が発生しない。したがって、次の入力シンボルX_{k+1}に対して符号化を行う（Ｓ１７０）。
【００２５】
図２は、本発明の一実施形態によるＣＡＢＡＣ復号化方法の各段階を示すフローチャートである。
【００２６】
まず、モデリング関数Ｆ（ｚ）により以前の復号化されたシンボルに対する統計に基づいて第１コンテキストモデルＣｍを選択し、ＣｍのＭＰＳ値と異なるＭＰＳ値を有する第２コンテキストモデルＣａを選択する（Ｓ２００）。このときにも、ＣｍのＭＰＳ値と異なるＭＰＳ値を有するコンテキストモデルが複数個である場合、そのうち最も小さなＰ＿ｌｐｓ値を有するコンテキストモデルを選択する。
【００２７】
第１コンテキストモデルＣｍを用いて暗号化された入力シンボルを復号化する（Ｓ２１０）。以下、Ｓ２１０段階で復号化された値を第１復号シンボルと呼ぶ。
【００２８】
第１復号シンボルがＣｍのＭＰＳ値と同一であるか比較する（Ｓ２２０）。もし、同じ値を有するならば、入力シンボルは、第１コンテキストモデルＣｍを用いて符号化されたものである。したがって、Ｃｍを用いて復号化された値である第１復号シンボルを復号化の結果値として選択し、次の入力シンボルに対して復号化を行う（Ｓ２７０）。
【００２９】
第１復号シンボルがＣｍのＭＰＳ値と異なる値を有する場合、第２コンテキストモデル

Ｃａを用いて暗号化された入力シンボルを復号化する（Ｓ２３０）。以下、Ｓ２３０段階で復号化された値を第２復号シンボルと呼ぶ。

【００３０】

第２復号シンボルがＣａのＭＰＳ値と同一であるか比較する（Ｓ２４０）。もし、異なる値を有するならば、入力シンボルは、第１コンテキストモデルＣｍを用いて符号化されたものである。したがって、Ｃｍを用いて復号化された値である第１復号シンボルを復号化の結果値として選択し、次の入力シンボルに対して復号化を行う（Ｓ２７０）。

【００３１】

第２復号シンボルがＣａのＭＰＳ値と同じ値を有する場合、第２復号シンボルをＣａを用いて再符号化し、その結果をＣｍを用いて復号化する（Ｓ２５０）。以下、Ｓ２５０段階で復号化された値を第３復号シンボルと呼ぶ。

【００３２】

第３復号シンボルがＣａのＭＰＳ値と同一であるか比較する（Ｓ２６０）。もし、同じ値を有するならば、入力シンボルは、第２コンテキストモデルＣａを用いて符号化されたものである。したがって、Ｃａを用いて復号化された値である第２復号シンボルを復号化の結果値として選択し、次の入力シンボルに対して復号化を行う（Ｓ２８０）。

【００３３】

もし、第３復号シンボルがＣａのＭＰＳ値と異なる値を有するならば、入力シンボルは、第１コンテキストモデルＣｍを用いて符号化されたものである。したがって、Ｃｍを用いて復号化された値である第１復号シンボルを復号化の結果値として選択し、次の入力シンボルに対して復号化を行う（Ｓ２７０）。

【００３４】

図１及び図２にそれぞれ示されているＣＡＢＡＣ符号化方法及び復号化方法によれば、アンビギュイティな場合を有せずとも、圧縮効率を向上させることができる。以下、本発明の原理及びアンビギュイティな場合が防止される理由などを説明する。以下、理解の便宜上、Ｈ．２６４の場合を主な実行例として説明する。

【００３５】

Ｂｉｎａｒｙ Ａｒｉｔｈｍｅｔｉｃ Ｃｏｄｉｎｇ（ＢＡＣ）は、二進シンボルの入力を受けて算術コーディング（Ａｒｉｔｈｍｅｔｉｃ Ｃｏｄｉｎｇ：ＡＣ）を行う。ＡＣは、周知のハフマンコーディングより効率の高いエントロピーコーディング方法である。Ｐ個のビットを利用するｐ－ビットＢＡＣは、数字の初期区間を[０，2^{p}）に設定し、各入力シンボルごとにそのシンボルの確率によって区間を分ける方法でコーディングを行う。区間は、開始値を示すＬ（Ｌｏｗｅｒ ｂｏｕｎｄ）と大きさを示すＲ（Ｒａｎｇｅ）とを用いて表現するが、初期区間の場合、Ｌは０であり、Ｒは2^{p}－１となる。シンボルは、発生確率の高いものをＭＰＳ、低いものをＬＰＳに区分する。ＭＰＳとＬＰＳとの値をそれぞれｖａｌＭＰＳ、ｖａｌＬＰＳという。

【００３６】

ＢＡＣは、区間選択に使用するために、各入力シンボルの確率を管理する。ＭＰＳが発生する確率Ｐ＿ｍｐｓとＬＰＳが発生する確率Ｐ＿ｌｐｓとは、以前の入力シンボルの統計を使用するが、例えば、１００個の入力シンボルのうち０と１とがそれぞれ２０回と８０回ずつ入力されたら、ＭＰＳ値は１であり、Ｐ＿ｌｐｓは０．２となる。新たな入力シンボルをコーディングした後、次のコーディングにこのシンボルの出現を反映できるように確率値を調節する。最初の入力シンボルの場合、以前のシンボルの統計値がないので、特定初期値を使用するが、一般的にＰ＿ｌｐｓは０．５に設定される。Ｈ．２６４の場合、いろいろな動映像データを予め分析して統計値を得て、この値を初期値として使用する。確率値は、Ｐ＿ｌｐｓのみ知っていればＰ＿ｍｐｓ（＝１－Ｐ＿ｌｐｓ）は求められるので、Ｐ＿ｌｐｓのみ管理する。

【００３７】

図７は、理想的な場合のＬＰＳ値の発生確率による圧縮効率を示すグラフである。理想的な場合、１ビットを表現するために必要なビット量は、下記の式（１）の通りであり、

図７は、Ｐ＿ｌｐｓ値による最小ビットの数を示すグラフである。

【００３８】

Ｐ＿ｌｐｓ＊ｌｏｇ２（１／Ｐ＿ｌｐｓ）＋Ｐ＿ｍｐｓ＊ｌｏｇ２（１／Ｐ＿ｍｐｓ）　（１）

前記式（１）によれば、同じ長さのデータをエントロピーコーディングするとき、同じシンボルの比率が高いほど圧縮効率が高くなるが、ＢＡＣの場合にＰ＿ｌｐｓ値が低いほど圧縮効率が高くなるのである。例えば、同じ１０ビットストリングであっても、１１１１０１１１１０が１０１０１００１１０より多く圧縮することが可能である。ＣＡＢＡＣは、圧縮効率を高めるために、二進入力ストリームを同じ値を有するシンボル同士グループ化する。現在の入力シンボルは、統計的に以前に出現したシンボル値と関連性が多いので、以前に現れたシンボル値（これをｚとする）によってグループを分ける。このとき、それぞれのグループをコンテキストとし、Ｐ＿ｌｐｓとｖａｌＭＰＳとをコンテキストごとに管理する。

【００３９】

コンテキストを決定する因子ｚをコンテキストパラメータという。コンテキストの例として、１０１０１０１０１０を圧縮するために二つのコンテキストを作るならば、奇数番目のビットと偶数番目のビットとに対するコンテキストを作りうる。この場合、奇数番目のビットは１のみで構成されており、偶数番目のビットは０のみで構成されているので、高い圧縮効率を示す。

【００４０】

図３は、本発明の一実施形態によるＣＡＢＡＣ符号化装置の構成を示すブロック図である。ＣＡＢＡＣは、各二進入力値に対して下記の過程を繰り返して圧縮を行う。

【００４１】

入力されるデータ１０が二進値でなければ、適当な方法で二進コードに変換される（３００）。次に、各ビットに対するコンテキストを選択する（３１０）。

【００４２】

選択されたコンテキストから取り出したＰ＿ｌｐｓとｖａｌＭＰＳ、入力ビットを用いてＢＡＣを行う（３２０）。

【００４３】

最後に、現在コーディングされたビットを、選択されたコンテキストの確率値に反映する（３３０）。

【００４４】

図４は、二進算術コーディングアルゴリズムが適用される一例を示す図である。

【００４５】

ＢＡＣの入力は、現在区間[Ｌ，Ｌ＋Ｒ）、１ビット入力値、入力値のコンテキスト（ｖａｌＭＰＳとＰ＿ｌｐｓ値とを有する）が使われる。符号化は、入力ビット値がＭＰＳであるかＬＰＳであるかによって、区間をそれぞれ[Ｌ，Ｌ＋Ｒ＊Ｐ＿ｍｐｓ）、[Ｌ＋Ｒ＊Ｐ＿ｍｐｓ、Ｌ＋Ｒ）に変更する。

【００４６】

区間を繰り返して変更すると、それ以上分けることができない状況となるので、これを防止するために再規格化（ｒｅｎｏｒｍａｌｉｚａｔｉｏｎ）を行う。再規格化は、現在の区間を拡大し、このために必要な情報をメモリに保存する過程である。

【００４７】

コーディング以後には、選択されたコンテキストのＰ＿ｌｐｓとｖａｌＭＰＳとに現在コーディングされたビットを反映するように更新する。この値は、直前まで現れたシンボルの頻度を用いて設定される。例えば、１００個の入力シンボルのうち、０と１とがそれぞれ２０回と８０回ずつ入力されたら、ｖａｌＭＰＳは１であり、Ｐ＿ｌｐｓは０．２となる。

【００４８】

最初の入力シンボルの場合、シンボルに対する統計値がないため、仮定された初期値を

使用するが、一般的にＰ＿ｌｐｓは０．５に設定される。Ｈ．２６４の場合、いろいろな動映像データを予め分析して統計値を得て、この値を初期値として使用する。確率は、入力シンボルの回数ではない他の方法で管理されてもよい。

【００４９】

ＢＡＣでは、同じ長さのデータを圧縮するときに同じシンボルが多く発生すれば、圧縮効率が高くなるので、入力シンボルを同じ値を有するグループ（すなわちコンテキスト）に分ける方法を提供する。

【００５０】

図５Ａないし図５Ｃは、Ｈ．２６４でのマクロブロックの動作ベクトル差（ｍｖｄ：ｍｏｔｉｏｎ　ｖｅｃｔｏｒ　ｄｉｆｆｅｒｅｎｃｅ）を示す図であり、図６Ａないし図６Ｃは、Ｈ．２６４でのマクロブロックのｍｖｄを二進化した値の一番目のビットを符号化する場合のコンテキストモデリングの一例を示す図である。図５Ａないし図５Ｃ、図６Ａないし図６Ｃを参照して、Ｈ．２６４のｍｖｄを二進化した一番目のビットをコーディングするときに用いられるグループ化方法を説明する。

【００５１】

図５Ａを参照するに、マクロブロックＢのｍｖｄが現在コーディングされるビットとすれば、左側マクロブロックＡと上側マクロブロックＣとのｍｖｄによってコンテキストが決定される。コンテキストパラメータｚは、ブロックＡとブロックＢとのｍｖｄ値によって決定される。コンテキストパラメータｚからコンテキスト集合Ｃ（本例では０，１，２）へのマップをＦ（ｚ）とすれば、ｍｖｄの場合にＦ（ｚ）は、次の式（２）の通りである。

【００５２】

Ｆ（０，０）＝０，Ｆ（０，１）＝０，Ｆ（０，２）＝０，Ｆ（０，３）＝１

Ｆ（１，０）＝０，Ｆ（１，１）＝０，Ｆ（１，２）＝１，Ｆ（１，３）＝１　（２）

デコーダは、エンコーダと初期Ｒ、コンテキストを決定するマッピングＦ、コンテキストパラメータ値のないときに使用するコンテキスト、及び各コンテキストごとの初期Ｐ＿ｌｐｓとｖａｌＭＰＳとを共有する。符号化されたデータが入力されれば、Ｐビットほど読み取ってＬ値に設定し、Ｐ＿ｌｐｓ、ｖａｌＭＰＳ、Ｌ、Ｒを用いてＡＣ復号化を行う。ＡＣ復号化過程でＬおよびＲは再調整される。復号化の後、復号化されたシンボルを反映するように、選択されたコンテキストのＰ＿ｌｐｓとｖａｌＭＰＳとを更新する。

【００５３】

コンテキストモデリングは、入力されるビットストリームのパターンを調べて同じ値を有するビット同士でグループを作るようにモデリングする。このとき、特定状況でのコンテキストは、統計的に以前の値と相関関係があるので、以前値によって決定する。図６Ａないし図６Ｃに示されている例で見れば、現在コーディングしようとするブロックＢのｍｖｄ値の一番目のビットのためのコンテキストは、ブロックＡとＣの値によって決定される。図６Ｂに示すように、ＡとＣのｍｖｄ値が２であれば、図６Ｂのように、Ｂのｍｖｄの一番目のビットのためのコンテキストは、１に固定される。このような状況であれば、コンテキスト１は、シンボル値がほとんど１である可能性が高いため、圧縮効率を高めることができる。同じ値を有するシンボルの頻度が高いほど圧縮効率が高いためである。

【００５４】

しかし、図５Ｃに示すように、Ｂのｍｖｄが０であれば、図６Ｃに示すように、ｍｖｄの一番目のビットの値は１でないため、他のコンテキストを選択することが圧縮に好都合でありうる。

【００５５】

本発明は、状況によってコンテキストを選択できる方法を提示することで、望ましくないパターンを有する入力に対してコーディング効率を高める方法を提案する。

【００５６】

図８は、一般的なコンテキストモデリングと本発明によるコンテキストモデリングとを比較して示す図である。図８を参照するに、コンテキストモデルは、直前に出たシンボル

値をそのまま利用したのであるが、この場合、コンテキスト０ｃｔｘ０とコンテキスト１ ｃｔｘ１とのＰ＿ｌｐｓは、それぞれ１／５、１／４となる。このモデルは、０と１とが交互に生じる場合を予想して作られたものである。しかし、図８の例では、入力ｘ６とｘ８が予想されたパターンとは違って入力された。この場合、ｘ６、ｘ８のコンテキストを異なるものに変更することが、コーディング効率がさらに高くなる。

【００５７】

本発明によるＣＡＢＡＣ復号化方法では、コンテキストを選択するときに、コンテキストパラメータとして各コンテキストが有しているＰ＿ｌｐｓとｖａｌＭＰＳ値とをヒントとして使用して、Ｐ＿ｌｐｓ値を低く維持できるモデリング方法を提案する。Ｃａは、コーディング効率を高めるために選択可能な候補となるコンテキストをいう。

【００５８】

本発明によるＣＡＢＡＣ符号化アルゴリズムは、現在の入力が選択された第１コンテキストのＭＰＳと同じ場合、選択された第１コンテキストを用いてコーディングし、異なる場合に現在入力された値がｖａｌＭＰＳである第２コンテキストを用いてコーディングする。

【００５９】

図８の場合を例として説明すれば、ｘ１ないしｘ５、ｘ７、及びｘ９は、既存に選択された第１コンテキストＣｍを使用し、ｘ６、ｘ８は、復号化がアンビギュイティな場合の問題がなければ、コーディング効率を高める異なるコンテキストを使用する。コーディング効率を高めるコンテキストは、ｖａｌＭＰＳが入力シンボル値と同一であり、Ｐ＿ｌｐｓが最も低いコンテキストとなる。かかる条件を有するコンテキストが存在しなければ、既存に選択されたコンテキストを使用し、複数存在するならば、順序上一番目のものを選択する。新たに選択されたコンテキストＣａを用いてコーディングするというのは、入力シンボルをそのシンボルが発生する可能性の高いコンテキストを用いて符号化するということなので、コーディング効率は高くなる。しかし、この場合にデコーダでアンビギュイティな場合が発生して復号化の不可能な場合が起こる。したがって、アンビギュイティな場合が発生しないようにしなければならない。デコータでアンビギュイティな場合が発生しないようにするために、エンコーダは、検査用復号化を共に行ってアンビギュイティな場合があるか検査し、アンビギュイティな場合があると判断されれば、既存に選択したコンテキストを用いてコーディングする。

【００６０】

図９は、本発明によるＣＡＢＡＣ符号化方法の遂行過程を入力シンボルの進行によって示す図である。図９において、Ｃａｓｅ１は、x_iとＣm_iとのｖａｌＭＰＳが同じ場合であり、Ｃａｓｅ２は、x_iとＣm_iとのｖａｌＭＰＳが異なる場合ある。

【００６１】

まず、x_iがＣm_iのＬＳＰであれば、Ｃa_iを用いてコーディングする（４０２）。エンコーダは、コーディングされるデータを復号化する間にＣa_iを用いて符号化してx_iが発見されれば、唯一の復号化が可能であるか検査する（４０４）。唯一の復号化が可能であれば、次に進み（４０７）、そうでなければ、x_iから再度コーディングする（４０６）。また、コーディング時には、Ｃa_iの代わりにＣm_iを使用しなければならない。すなわち、符号化過程で復号化で発生する状況を予測してアンビギュイティな場合が発生する状況では、従来の方法で選択したコンテキストをそのまま選択するという規則を置いてアンビギュイティな場合を回避することである。

【００６２】

以下、本発明による符号化及び復号化でアンビギュイティな場合が発生しないことを説明する。

【００６３】

エンコーダは、符号化時に復号化のアンビギュイティな場合の検査のために、符号化されたデータαを毎度復号化する。復号化過程でアンビギュイティな場合が発生しない時の

み、新たなコンテキストを用いてコーディングする。

【００６４】

次は、符号化アルゴリズムのコンテキスト選択方法である。

【００６５】

Ｅｎｃｏｄｅ（x_i）

ｅ１．Cm_i、Ca_iを決定

ｅ２．Ｅ（Cm_i，x_i）

ｅ３．ｉｆ（x_i＝＝ｍｐｓ（Ｃｍ）orCa_iが存在しない）

ｅ４．Ｅｎｃｏｄｅ（x_{i+1}）

ｅ５．ｅｌｓｅ（x_iがＣｍのＬＰＳであるが、x_iをＭＰＳとして有する他のコンテキストがある状況）

ｅ６．Ｅ（Ca_i，x_i）

ｅ７．ｉｆ（αを復号化して現在復号化されるx_iがＣａを用いて符号化されたシンボルであれば）

ｅ８．ｉｆ（D_i（Cm_i、α＝＝ｍｐｓ（Cm_i））

ｅ９．x_iから再度符号化。この場合は無条件Ｃｍを利用する。

【００６６】

ｅａ．ｅｌｓｅ

ｅｂ．ｎｅｘｔ　Ｅｎｃｏｄｅ（）（Ｃａを用いて符号化しても、復号化でアンビギュイティな場合が無い）

このような符号化アルゴリズムでｅ８及びｅ９段階は、Ｃａを用いて符号化されたｘは、Ｃｍを用いて復号化すれば、その結果が常にｍｐｓ（Ｃａ）であるということを保証する。

【００６７】

デコータのコンテキスト選択は、次のように行う。

【００６８】

Ｄｅｃｏｄｅ（α，ｉ）（ｉ番目シンボルx_iを復号化する）

ｄ１．Cm_i、Ca_iを決定

ｄ２．ｉｆ（D_i（Ｃｍ，α）＝＝ｍｐｓ（Ｃｍ））

ｄ３．Ｄｅｃｏｄｅ（α，ｉ＋１）（すなわち、Ｃｍを用いて復号化し、次のシンボルに進む）

ｄ４．ｅｌｓｅ　ｉｆ（（D_i（Ｃａ，α＝＝ｍｐｓ（Ｃａ））（この場合は、Ｃｍを用いて復号化してもＭＰＳ（Ｃａ）となる）

ｄ５．ｉｆ（D_i（Ｃｍ，Ｅ（Ｃａ，ｍｐｓ（Ｃａ））））＝＝ｍｐｓ（Ｃａ））

ｄ６．D_i（Ｃａ，α，Ｄｅｃｏｄｅ（α，ｉ＋１）（すなわち、Ｃａを用いて復号化し、次のシンボルに進む）

ｄ７．D_i（Ｃｍ，α，Ｄｅｃｏｄｅ（α，ｉ＋１）（Ｃｍを用いて復号化し、次のシンボルに進む）

唯一に復号化することが可能であることを見せるためには、x_iを符号化するときに使用したコンテキストと復号化するときに使用したコンテキストとが常に同一であるということを示せばよい。エンコーダとデコータとで同じ方式でＰ＿ｌｐｓ、ｖａｌＭＰＳを維持するため、デコータでx_iを復号化する直前のＰ＿ｌｐｓとｖａｌＭＰＳとは、エンコーダでx_iを符号化する直前の値と同一である。したがって、この状況でCm_iとCa_iとは、エンコーダとデコータとが同様のものを探す。もし、Ｃａがない場合であれば、符号化と復号化とがいずれもＣｍを利用せざるを得ないため問題とならない。したがって、Ｃａがある場合だけ考えればよい。

【００６９】

符号化アルゴリズムによれば、入力シンボルｘがｍｐｓ（Ｃｍ）である場合には、Ｅ（Ｃｍ，ｘ）が行われる。ｘがｍｐｓ（Ｃｍ）でない場合には、Ｅ（Ｃｍ，ｘ）やＥ（Ｃａ，ｘ）が行われる。それぞれの場合に、符号化されたデータをａ１、ａ２、ａ３とすれば

、式（３）ないし式（５）のように表現できる。

【００７０】

ａ１＝Ｅ（Ｃｍ，ｍｐｓ（Ｃｍ））　（３）

ａ２＝Ｅ（Ｃａ，ｍｐｓ（Ｃａ））　（４）

ａ３＝Ｅ（Ｃｍ，ｍｐｓ（Ｃａ））　（５）

式（３）の場合、D_i（Ｃｍ，ａ１）＝＝ｍｐｓ（Ｃｍ）であり、ｄ２段階の結果は真（ｔｒｕｅ）となるので、同じコンテキストを使用して復号化する。

【００７１】

式（４）の場合、ｅ８及びｅ９段階によってｄ２段階の結果が偽（ｆａｌｓｅ）となり、ｄ４段階の結果が真となる。

【００７２】

式（５）の場合、Ｃａを用いて復号化するか、Ｃｍを用いて復号化する場合のいずれもｍｐｓ（Ｃａ）である。したがって、ｄ２段階の結果が偽となり、ｄ４段階の結果が真となる。

【００７３】

式（４）及び式（５）の場合、符号化時にｅ８及びｅ９を適用して選択するコンテキストを区分するので、ｄ５段階でｅ７及びｅ８を適用して見れば、その結果によって式（４）の場合であるか、式（５）の場合であるか区分される。以上で、デコータとエンコーダとが特定の入力シンボルに対して常に同じコンテキストを使用するので、復号化は元の入力データを復元するようになる。

【００７４】

本発明によれば、ＣＡＢＡＣの圧縮効率を高める。ＣＡＢＡＣは、Ｈ．２６４などマルチメディアデータの圧縮に一般に使われるので、本発明を採用すれば、動映像圧縮の性能を高めることができる。本発明による方法を行うために、エンコーダおよびデコータの複雑度が少し高まるが、ハードウェア（Ｈ／Ｗ）の性能が継続的に向上しているので問題とならない。

【００７５】

本発明は、コンピュータで読み取り可能な記録媒体にコンピュータ（情報処理の機能を持つ装置を全て含む）が読み取り可能なコードとして具現可能である。コンピュータで読み取り可能な記録媒体は、コンピュータシステムによって読み取れるデータが保存されるあらゆる種類の記録装置を含む。コンピュータで読み取り可能な記録媒体の例としては、ＲＯＭ、ＲＡＭ、ＣＤ－ＲＯＭ、磁気テープ、フロッピー（登録商標）ディスク、光データ保存装置などがある。

【００７６】

前記説明が多様な実施形態に適用される本発明の新規な特徴を中心に説明したが、当業者であれば、本発明の範囲を逸脱しない範囲内において、前記説明された装置及び方法の形態及び細部事項から多様な削除、代替、変更が可能であるということが理解できるであろう。したがって、本発明の範囲は、前述した説明ではなく、特許請求の範囲によって定義される、それと同等な範囲内にある全ての変更は、本発明の範囲に属する。

【産業上の利用可能性】

【００７７】

本発明は、動映像圧縮関連の技術分野に好適に用いられる。

【図面の簡単な説明】

【００７８】

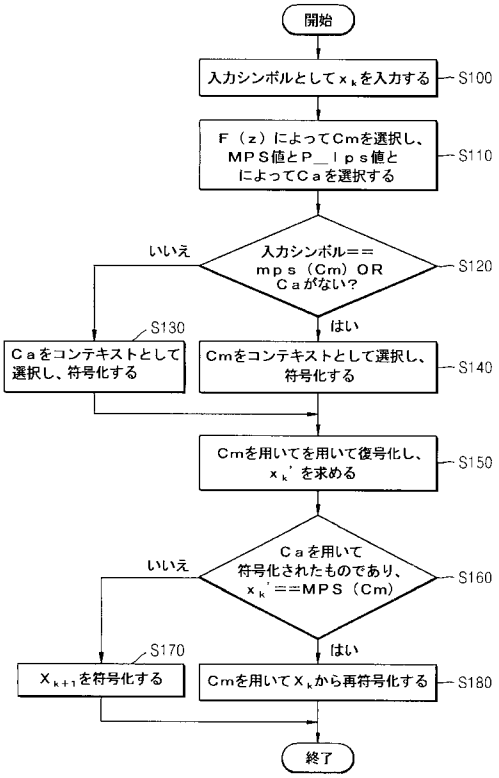
【図１】本発明の一実施形態によるＣＡＢＡＣ符号化方法の各段階を示すフローチャートである。

【図２】本発明の一実施形態によるＣＡＢＡＣ復号化方法の各段階を示すフローチャートである。

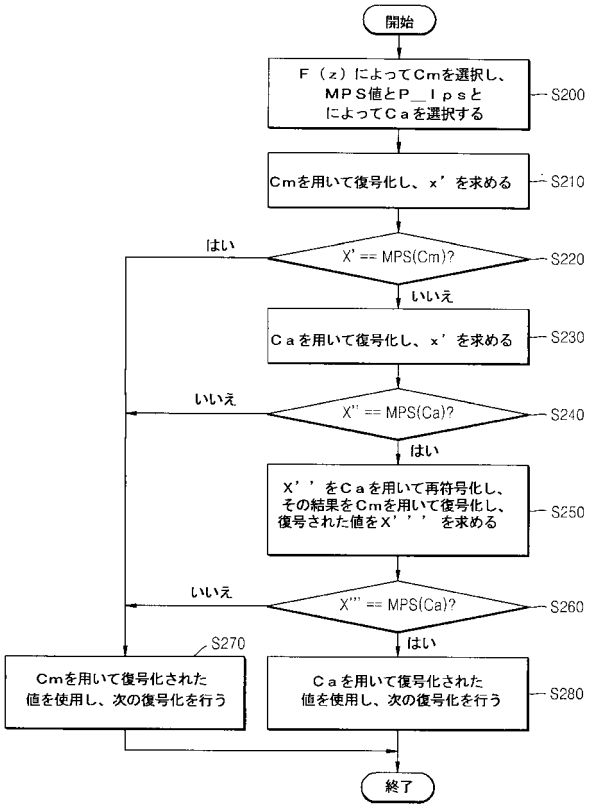
【図３】本発明の一実施形態によるＣＡＢＡＣ符号化装置の構成を示すブロック図である。

【図４】二進算術コーディングアルゴリズムが適用される一例を示す図である。
【図５Ａ】Ｈ．２６４でのマクロブロックのｍｖｄを示す図である。
【図５Ｂ】Ｈ．２６４でのマクロブロックのｍｖｄを示す図である。
【図５Ｃ】Ｈ．２６４でのマクロブロックのｍｖｄを示す図である。
【図６Ａ】Ｈ．２６４でのマクロブロックのｍｖｄを符号化する場合のコンテキストモデリングの一例を示す図である。
【図６Ｂ】Ｈ．２６４でのマクロブロックのｍｖｄを符号化する場合のコンテキストモデリングの一例を示す図である。
【図６Ｃ】Ｈ．２６４でのマクロブロックのｍｖｄを符号化する場合のコンテキストモデリングの一例を示す図である。
【図７】理想的な場合のＬＰＳ値の発生確率による圧縮効率を示すグラフである。
【図８】一般的なコンテキストモデリングと本発明によるコンテキストモデリングとを比較して示す図である。
【図９】本発明によるＣＡＢＡＣ符号化方法の遂行過程を入力シンボルの進行によって示す図である。
【符号の説明】
【００７９】
ｘ　入力シンボル
x_i　ｉ番目入力シンボル
ｍｐｓ（Ｃ）　コンテキストＣのＭＰＳ値
Ｃｍ　一般的なコンテキストモデリングにより選択されたコンテキスト
Cm_i　x_iのためのＣｍ
Ｃａ　ｘとＭＰＳ（Ｃｍ）とが異なるときに使用するコンテキスト
Ca_i　x_iのためのＣａ
α　符号化されたデータ
Ｅ（Ｃ，ｘ）　コンテキストＣと入力ｘとを用いて算術コーディング（符号化）の遂行
Ｄ（Ｃ，α）　コンテキストＣと符号化されたデータαとを用いて算術コーディング（復号化）の遂行
D_i（Ｃ，α）　コンテキストＣと符号化されたデータαとを用いてｉ番目のデータ（x_i）を復号化

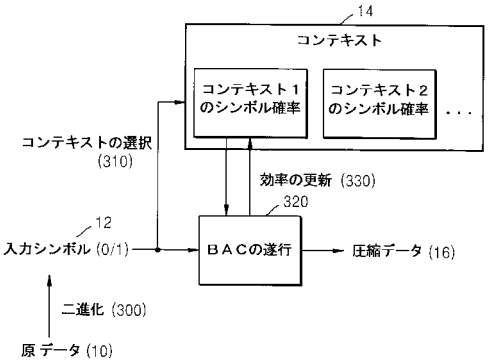
【図１】



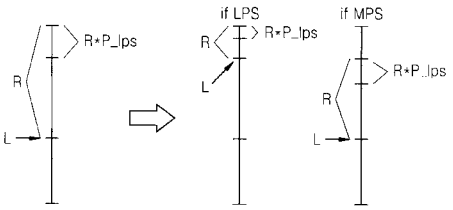
【図２】



【図３】



【図４】



【図５Ａ】

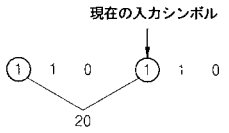
【図５Ｂ】

【図５Ｃ】

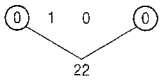
【図６Ａ】

mvd	二進値
0	0
1	1 0
2	1 1 0
3	1 1 1 0
4	1 1 1 1 0
…	…

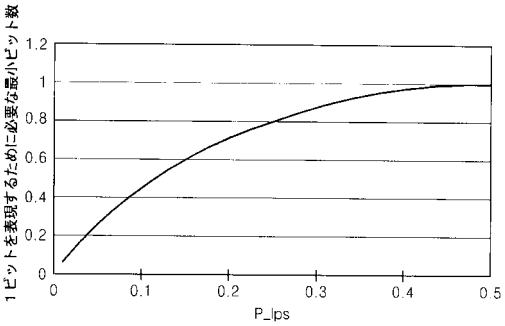
【図６Ｂ】



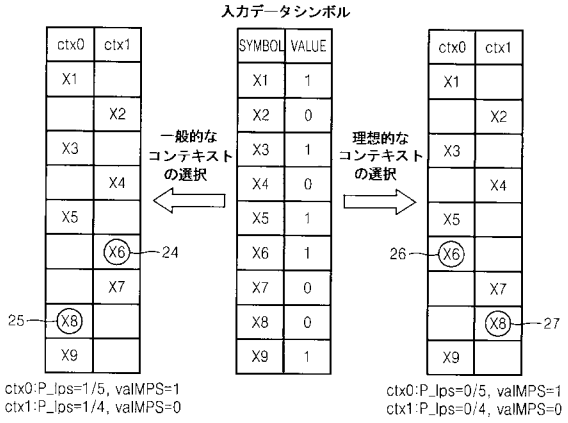
【図６Ｃ】



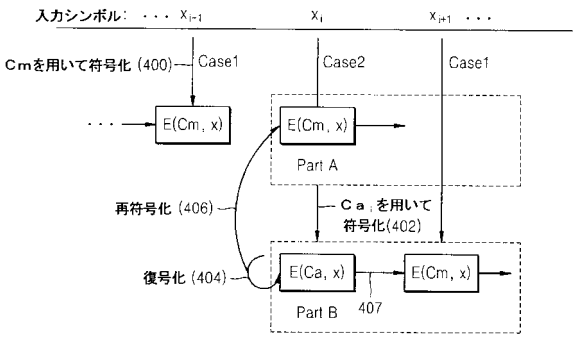
【図７】



【図８】



【図９】



フロントページの続き

Ｆターム(参考) 5C059 MA00 ME11 UA02 UA05
5J064 AA02 BA09 BA10 BC14 BC22 BD02 BD03