

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4586163号
(P4586163)

(45) 発行日 平成22年11月24日(2010.11.24)

(24) 登録日 平成22年9月17日(2010.9.17)

(51) Int. Cl. F I
G06F 7/58 (2006.01) G06F 7/58 B
G09C 1/00 (2006.01) G09C 1/00 650B

請求項の数 4 (全 10 頁)

(21) 出願番号	特願2005-262581 (P2005-262581)	(73) 特許権者	504165591 国立大学法人岩手大学 岩手県盛岡市上田三丁目18番8号
(22) 出願日	平成17年9月9日(2005.9.9)	(74) 代理人	100105371 弁理士 加古 進
(65) 公開番号	特開2007-73012 (P2007-73012A)	(72) 発明者	吉田 等明 岩手県盛岡市上田三丁目18番8号 国立 大学法人岩手大学内
(43) 公開日	平成19年3月22日(2007.3.22)	(72) 発明者	中西 貴裕 岩手県盛岡市上田三丁目18番8号 国立 大学法人岩手大学内
審査請求日	平成20年5月8日(2008.5.8)	審査官	田中 友章

最終頁に続く

(54) 【発明の名称】 暗号化システム

(57) 【特許請求の範囲】

【請求項1】

入力した平文に暗号化を行う暗号化システムであって、
 擬似乱数を出力する2個の擬似乱数発生手段と、
 一方の前記擬似乱数発生手段から出力した擬似乱数を用いてマスクパターンを生成する
 マスクパターン生成手段と、
 他方の前記擬似乱数発生手段から出力した擬似乱数に前記マスクパターンをマスクした
 結果を出力する乱数出力手段とを有する乱数生成システムと、
 入力した平文と前記乱数生成システムからの乱数とを演算して暗号化文を出力する暗号
 化手段と

を備えることを特徴とする暗号化システム。

【請求項2】

請求項1に記載の暗号化システムにおいて、
 前記マスクパターンは、0と1とが同数であることを
 を特徴とする暗号化システム。

【請求項3】

請求項1または2に記載の暗号化システムにおいて、
 前記擬似乱数発生手段はカオス・ニューラルネットワークであり、前記擬似乱数は前記
 カオス・ニューラルネットワークからの出力の下位ビットであること
 を特徴とする暗号化システム。

【請求項 4】

請求項 1 ~ 3 に記載の暗号化システムの機能をコンピュータ・システムに構築させるプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、暗号学的に安全な擬似乱数を発生する技術に関するものである。

【背景技術】

【0002】

デジタルデータの暗号化および復号を行なうシステムにおいては、従来、ストリーム暗号方式という手法が広く用いられている。 10

ストリーム暗号方式とは、平文を 1 ビット、または数ビットごとに暗号化・復号を行なう方法であり、暗号化鍵から鍵系列とよばれる擬似乱数を生成し、その擬似乱数により平文を逐次暗号化する暗号方式である。擬似乱数は、ソフトウェアでも比較的高速に生成することができるため、ストリーム暗号方式による暗号化及び復号システムは、ソフトウェアで実現する場合でも、高速な処理を行なうことができる。この方式では、擬似乱数の生成方法が非常に重要であり、暗号強度に大きく影響する。

【0003】

発明者らは、ホップフィールドネットワークや誤差逆伝搬学習則等で用いられている通常の人工ニューロンからなり、カオス出力が得られるニューラルネットワークを発明し、この独自のニューラルネットワークのことをカオス・ニューラルネットワーク（以下 CNN）と呼んでいる（例えば非特許文献 1，特許文献 1，特許文献 2）。そして CNN を用いることにより、従来技術よりも乱数性に優れた擬似乱数を生成できることを明らかにしている（特許文献 2）。 20

発明者らの CNN のニューロンモデルを図 1 (a) および式 (1)，式 (2) に示す。

【0004】

【数 1】

$$x_j(t) = f_s(u_j) = \frac{1}{1 + \exp(-u_j(t))} \quad \dots (1)$$

30

【0005】

【数 2】

$$u_j(t) = \sum_{i=1}^n w_{ij} x_i(t-1) - \theta_j + I_j \quad \dots (2)$$

【0006】

ここで、式 (1) で表される非線形出力関数 f_s は特異な関数ではなく、一般的に用いられている sigmoid 関数である。図 1 (a) および式 (1)，式 (2) におけるその他の各係数は、以下のように定義する。

$x_j(t)$: 時刻 t におけるニューロン j の出力 ($t = 0, 1, 2, \dots$) 40

u_j : ニューロン j の内部状態

$x_i(t-1)$: 時刻 $t-1$ におけるニューロン i からの入力

w_{ij} : ニューロン i からニューロン j への結合荷重

θ_j : ニューロン j の閾値

I_j : ニューロン j への外部入力値

【0007】

自然界では単一で振動する出力を発生するニューロンもあるが、CNN に用いた上述のニューロンの場合には、1 個のニューロンのみでは振動しない。ニューロン・ネットワークの例として、ニューロン 4 個から構成されるカオス・ニューラルネットワーク (CNN) の構成を図 1 (b) に示す。図 1 (b) は、ニューロン 1 (N_1) 1 2 1，ニューロン 50

2 (N_2) 1 2 2 , ニューロン 3 (N_3) 1 2 3 , ニューロン 4 (N_4) 1 2 4 の 4 個のニューロンから構成されている。構造の特徴としてニューロンのサイクルが 1 つあるので、*cyclic-4nn* (以下 *C-4nn*) と呼んでいる。*C-4nn* のパラメータ (結合荷重 W 、閾値 および外部入力 I_j) を表 1 に示す。

【 0 0 0 8 】

【表 1】

W_{12}	-40
W_{13}	20
W_{24}	16.8
W_{31}	-20
W_{43}	-8.6
θ_1	0
θ_2	0
θ_3	0
θ_4	0
I_1	-0.497
I_3	-0.497

10

【 0 0 0 9 】

さらに、発明者らは、CNNより生成された出力の下位 *bit* がよい乱数性を持っていることを明らかにし、従来の乱数発生方法よりも一様性に優れ、容易に真性乱数と区別できないような乱数を生成することができる乱数生成システムを発明している (特許文献 2)。

20

【 0 0 1 0 】

【非特許文献 1】S.Kawamura, H.Yoshida, M.Miura, and M.Abe: Implementation of Uniform Pseudo Random Number Generator and Application to Stream Cipher based on Chaos Neural Network, Proceedings of Papers, the International Conference on Fundamentals of Electronics, Communications and Computer Sciences, R-18, 2002.

【特許文献 1】特開 2 0 0 1 - 1 4 4 7 4 6 号公報

【特許文献 2】特開 2 0 0 3 - 7 6 2 7 2 号公報

30

【発明の開示】

【発明が解決しようとする課題】

【 0 0 1 1 】

ストリーム暗号方式への攻撃を防御する観点から、より堅牢性の高い暗号化を実現できる擬似乱数の生成方法が望まれている。本発明の課題は、擬似乱数の生成において、ストリーム暗号方式などに利用可能な暗号系の暗号化鍵空間を広げる方法 (以降、マスクパターン法と呼ぶ) を提案し、さらに堅牢性の高い暗号化を可能にすることである。

【課題を解決するための手段】

【 0 0 1 2 】

上記の課題を解決するために、本発明は、入力した平文に暗号化を行う暗号化システムであって、擬似乱数を出力する 2 個の擬似乱数発生手段と、一方の前記擬似乱数発生手段から出力した擬似乱数を用いてマスクパターンを生成するマスクパターン生成手段と、

40

他方の前記擬似乱数発生手段から出力した擬似乱数に前記マスクパターンをマスクした結果を出力する乱数出力手段とを有する乱数生成システムと、入力した平文と前記乱数生成システムからの乱数とを演算して暗号化文を出力する暗号化手段とを備えることを特徴とする。

前記マスクパターンは、0 と 1 とが同数であるとするとよい。

前記擬似乱数発生手段はカオス・ニューラルネットワークであり、前記擬似乱数は前記カオス・ニューラルネットワークからの出力の下位ビットとすることもできる。

50

上記の暗号化システムの機能をコンピュータ・システムに構築させるプログラムも本発明である。

【発明の効果】

【0013】

本発明で提案するマスクパターン法を用いた乱数生成システムでは、ある擬似乱数発生手段から生成された擬似乱数を、異なった系の擬似乱数発生手段からの擬似乱数を基に作り出したマスクパターンでマスクして、その結果の乱数列を暗号化の擬似乱数として用いる。そのため、発生可能な暗号化鍵の数を必要に応じて増やすことが可能となり、暗号化鍵空間を広げることができた。これにより、従来よりも暗号学的に安全な擬似乱数を生成することができ、暗号化システムを堅牢化できる。

10

特に、発明者らの技術であるCNNより生成された出力の下位bitを擬似乱数として本実施形態のマスクパターン法を用いることにより、CNNの出力の下位bitの優れた乱数性を落とすことなく暗号化鍵空間を広げることができる。具体的には、発明者らの既存技術（非特許文献1，特許文献1，特許文献2を参照）に対して、0と1とが同数のマスクパターンを用いて上述のマスクパターン法を適用すると、暗号鍵空間を標準で 10^4 倍に改良することができた。これにより解読に要する時間を 10^{29} 年以上に延ばすことができる。もともと、発明者らの既存技術であるCNNによる暗号化方法は、米国の次世代標準暗号であるRijndaelより高速かつ堅牢であるが、本発明の手法によれば、これをさらに堅牢化できる。

【発明を実施するための最良の形態】

20

【0014】

以降、本発明の乱数生成システムの実施形態の一例を詳細に説明する。まず、ストリーム暗号方式の概要と考えられる主な攻撃方法について説明し、次に、これらの攻撃に対処するための方法として、本実施形態の手法であるマスクパターン法を提案する。

< 1. 暗号化 >

本実施形態では、CNNより生成した擬似乱数から、本実施形態で提案する新しい手法であるマスクパターン法により暗号化に用いる乱数を生成し、上述のストリーム暗号方式による暗号化を行うシステムを実装する場合を例として説明するが、他の擬似乱数発生手段を用いてもよい。

以降、この暗号化システムの実装に向け、ストリーム暗号方式の概要と、考えられる主な攻撃方法について述べる。

30

【0015】

(1-1. ストリーム暗号方式)

用いる暗号として、秘密鍵暗号方式（共通鍵暗号方式）の一種である、上述のストリーム暗号方式を使用する。その特徴を以下に示す。

- ・ 1bit毎に暗号化および復号を行う。
- ・ 一般的に堅牢性よりも速度を重視する。

【0016】

(1-2. 暗号化の概要)

本実施形態で用いるストリーム暗号方式の暗号化と復号について概要を示す。

40

(1) 暗号化

平文 XOR 乱数列 = 暗文 ... (3.1)

(2) 復号

暗文 XOR 乱数列 = 平文 ... (3.2)

ここで、平文とは暗号化されていないデータであり、暗文は、暗号化され、そのままでは読めない状態になっているデータである。

【0017】

(1-3. 主な攻撃方法)

一方、ストリーム暗号を攻撃する主な方法として、(1) 既知平文攻撃、(2) 選択平文攻撃、(3) 差分攻撃、等の方法が挙げられる。

50

(1) 既知平文攻撃

攻撃者が、平文と暗文の両方を手に入れることができる立場にある場合の攻撃方法。

平文 XOR 暗文 = 乱数列 ... (3 . 3)

式 (3 . 3) により、乱数列を取得されてしまう。これを繰り返し行うことによって、次の乱数列を予測される恐れがある。

【 0 0 1 8 】

(2) 選択平文攻撃

攻撃者が平文の内容を自由に選択できる立場にある場合の攻撃方法。式 (3 . 1) において、平文の内容を全て 0 にすることにより、

0 XOR 乱数列 = 暗文 ... (3 . 4)

乱数列 = 暗文 ... (3 . 5)

となり、乱数列の内容が暗文にそのまま表れ、乱数列を取得されてしまう。

【 0 0 1 9 】

(3) 差分攻撃

平文と暗文のセットを複数持つことができる立場にある場合の攻撃方法。

平文 1 XOR 乱数列 1 = 暗文 1 ... (3 . 6)

平文 2 XOR 乱数列 2 = 暗文 2 ... (3 . 7)

(平文 1 XOR 平文 2) XOR (乱数列 1 XOR 乱数列 2) = (暗文 1 XOR 暗文 2) ... (3 . 8)

このとき、乱数列 1 と乱数列 2 が等しくなる可能性がある、

乱数列 1 XOR 乱数列 2 = 0 ... (3 . 9)

となり、式 (3 . 8) と式 (3 . 9) から、

平文 1 XOR 平文 2 = 暗文 1 XOR 暗文 2 ... (3 . 1 0)

となる。式 (3 . 1 0) より、同じ乱数列を使用すると、乱数列の内容に関わらず、解除されてしまう。

【 0 0 2 0 】

< 4 . マスクパターン法 >

(4 - 1 . 目的)

上述したように、カオス出力の下位 b i t は乱数性に優れている (特許文献 2 参照) が、本実施形態では、暗号化システムにおいて上述の攻撃への対処をより強化するために、以降説明する「マスクパターン法」を提案する。なお、本実施形態では、擬似乱数を発生する擬似乱数発生手段として、上述の C N N を用いる場合を例として説明するが、他の擬似乱数発生手段を用いてもよい。

【 0 0 2 1 】

(4 - 2 . 概要)

本実施形態のマスクパターン法とは、ある擬似乱数発生手段 (ここでは C N N を例として説明する) からの擬似乱数を、異なった系の擬似乱数発生手段 (ここでは C N N を例として説明する) からの擬似乱数を基に作り出したマスクパターンでマスクすることにより、暗号学的に安全な擬似乱数を生成する方法である。本実施形態のマスクパターン法の処理の流れを図 2 (a) に示す。

本実施形態の乱数生成システムにおいて、乱数生成に用いるための擬似乱数の生成は、擬似乱数発生手段により行なう。なお、上述したように、本実施形態では擬似乱数発生手段の一例として発明者らのカオス・ニューラルネットワーク (C N N) を用いる例で説明するが、他の擬似乱数発生手段を用いてもよい。マスクパターンの生成は、マスクパターン生成手段により行なう。擬似乱数発生手段からの擬似乱数をマスクパターンでマスクして暗号化に用いる乱数を生成・出力する処理は、乱数出力手段により行なう。

以降、図 2 (a) および図 2 (b) を参照しながら、本実施形態で提案するマスクパターン法について詳しく説明する。

【 0 0 2 2 】

(4 - 3 . 擬似乱数の生成)

10

20

30

40

50

図2(a)に示すように、マスクパターン法は、あるCNN(CNN-2)220から生成された擬似乱数(222a, 222b, 222c, 222d, ...)を、異なった系のCNN(CNN-1)210から生成された擬似乱数212を基に作り出したマスクパターン214によりマスクして、より堅牢性を高める方法である。

これらの擬似乱数の生成は、擬似乱数発生手段により行なう。ここでは、擬似乱数発生手段として発明者らのCNNを用いた例で説明するが、他の擬似乱数発生手段を用いてもよい。

CNNから擬似乱数を出力する例を図2(b)に示す。図2(b)に示すように、例えばdouble型変数を格納できるレジスタ240を用意し、図1(b)に示すようなCNNを構成するニューロンから、時系列で発生する出力を取り出して、レジスタ240に格納する。次に、レジスタ240から、仮数部の下位のxビットを取り出す。取り出したビット列が乱数となっている。この乱数は、CNNを構成しているニューロンであれば、どのニューロン(図1(b)では N_1 121~ N_4 124)から取り出してもよい。また、数表現がIEEE754形式以外の場合でも、浮動小数点または固定小数点表現を用いている場合は、本手法を適応できる。(詳しくは特許文献2を参照)。なお、本実施形態では、レジスタ240から仮数部の下位24bitを取り出すものとしているが、これは、下位24ビットのランダム性がよいからである。

【0023】

(4-4. マスクパターンの生成)

次に、本実施形態で用いるマスクパターン214の生成方法を、一例を挙げて説明する。

マスクパターン214は、本実施形態の乱数生成システムのマスクパターン生成手段により生成する。なお、ここで説明するマスクパターン生成方法は一例であり、他の方法により生成してもよい。

本実施形態で用いるマスクパターン214は24bitの長さとする。ここでは、例として、0と1が同数(0が12bit、1が12bit)でマスクパターンを構成する場合を示す。本実施形態のマスクパターン法では、0と1とが必ずしも同数でなくてもよいが、0と1とが同数の場合にマスクパターンの組合せの数が最大となり、堅牢性も最大になるからである。このマスクパターンを8個生成し、192bitを1セットとして用いる。

この1セットのマスクパターンでCNN-2)220からの擬似乱数(222a, 222b, 222c, 222d, ...)のマスクを行なった結果の乱数列(230a, 230b, 230c, 230d, ...)を、暗号化の際の乱数列として用いる。

ここで、24bitのマスクパターンの可能な組み合わせの数は、 ${}_{24}C_{12} \cdot 10^6$ となる。1セットのマスクパターン(192bit)を考えると、可能なマスクパターンの数は ${}_{106}P_8 \cdot 10^{48}$ となる。

【0024】

上述したように、本実施形態では、0と1の個数が同数になるように24bitのマスクパターンを作り、それを8個生成して1セット(192bit)のマスクパターンとする。ここで、0と1の個数を同数にするため、例えば、24bit内で順にペアとなるビットを決め、片方に0、もう片方に1を入れていく。以降、マスクパターン生成の例として、CNN-1)210から生成した擬似乱数212を用いて、ペアとなるビットの決定、およびペアとなったビットのどちらが0でどちらが1になるかを決定する方法を説明する。なお、マスクパターンの生成方法はこれに限られず、他の方法を用いてもよい。また、上述したように0と1とが同数でなくてもよい。

【0025】

(1) ペアとなるビットの決定

本実施形態の例では、CNN-1)210から24bitの擬似乱数212を生成し、この擬似乱数の値を用いて、ペアを順次1組ずつ決めていく。24bitのマスクパターンを生成するには、12組のペアを決定することになる。

10

20

30

40

50

24 bitのマスクパターン生成において、CNN - 1 210からの擬似乱数 x_n (24 bitの場合 $n = 1, 2, \dots, 12$)を用いて、 n 組目のペアの相手となるビットの番号 y_n (24 bitの場合 $n = 1, 2, \dots, 12$)を決定する処理は、次の式(4)で表される。

【0026】

【数3】

$$y_n = \left[x_n \div \frac{2^{24}}{23-2(n-1)} \right] + 1 \quad \dots(4)$$

なお、式(4)において $[a]$ はガウス記号であり、実数 a を超えない最大の整数であることを示している。

10

【0027】

以降、式(4)を参照しながら、ペアとなるビットを決定する処理の一例を説明する。

まず、24 bitのマスクパターンの1ビット目とペアになるビットを決定する(1組目のペアの決定)。CNN - 1 210からの24 bitの擬似乱数 x_1 を用いて、1ビット目とペアになるビットを決める。

具体的には、24 bitで表現できる数(すなわち、0から $2^{24} - 1$ までの 2^{24} 個の数)を、マスクパターン内のまだ決定していないビットの数(ここでは1ビット目を除いた数である23)で割って、そのビット数(ここでは23)のグループに分割し、数が小さい方から順にグループ__1, グループ__2, ..., とする。一方、まだ決定していない

20

ビット(ここでは1ビット目を除いた23 bit)を順に、ビット__1, ビット__2, ..., とする。

次に、24 bitの擬似乱数 x_1 の値が含まれているグループ__ y_1 を見つける。そして、 y_1 番目のビットであるビット__ y_1 を、1ビット目とペアになるビットに決定する。

【0028】

同様に、2組目以降も、マスクパターンからまだ決定していないビット(n 組目では $24 - 2(n - 1)$ ビット残っている)のうち最初のビットを選び、これとペアになるビットを決定する。CNN - 1 210から次の擬似乱数 x_n を生成して、上述の1組目と同様に、24 bitで表現できる数を、まだ決定していないビット数(n 組目では $23 - 2$

30

上記の処理を、全てのペアが決定するまで繰り返す。なお、最後の(12組目の)ペアは残りの2 bitであるので、ペアの決定処理を行なう必要はなく、次に説明するビットの値の決定(0か1かの決定)のみ行なえばよい。

【0029】

(2) ペアとなったビットの値の決定

n 組目(24 bitの場合 $n = 1, 2, \dots, 12$)のペアとなるビットが決定したら、擬似乱数 x_n の最下位ビットにより、 n 組目のビットのどちらが0でどちらが1になるかを決定する。本実施形態では、例えば、擬似乱数 x_n の最下位ビットが0の場合には、先

40

の(まだ決定していないビットのうち最初の)ビットを0とし、ペアに決定したビットを1とする。逆に擬似乱数 x_n の最下位ビットが1の場合には、先のビットを1とし、ペアに決定したビットを0とする。のように決めておく。

以上の(1)(2)の処理により、1個のマスクパターンが生成される。

さらに、ここまでの処理を計8回繰り返し、1セット192 bitのマスクパターンとする。なお、マスクパターンのビット数は24 bit以外でもよく、1セットのマスクパターン数は8個以外でもよい。

【0030】

(4-5. 暗号化に用いる乱数列の生成)

次に、上述で生成したマスクパターンセット214を用いて、暗号化に用いる擬似乱数

50

を生成する方法について説明する。この乱数は、本実施形態の乱数生成システムの乱数出力手段により生成し、出力する。

マスクパターン 2 1 4 生成時に使った CNN - 1 2 1 0 とは違う系の CNN (CNN - 2 2 2 0) を使い、2 4 b i t の擬似乱数を 8 個 (2 2 2 a , 2 2 2 b , 2 2 2 c , 2 2 2 d , ...) を取り出す。これを上述で生成した 1 セット (8 個) のマスクパターン 2 1 4 によりマスクする。本実施形態では、例えばマスクが 0 のビットは破棄し、マスクが 1 のビットだけを取り出すものとする。この結果、出力 (2 3 0 a , 2 3 0 b , 2 3 0 c , 2 3 0 d , ...) は 1 セット 9 6 b i t (= 1 2 b i t × 8) となる。これを暗号化の際の乱数列として用いる。

ここで、組み合わせるマスクパターンの数は、上述したように、必ずしも 8 個でなくとも良いので、2 4 ビット i 個使ったとすれば、2 4 i ビットのマスクパターンとなり、可能な組み合わせの数は、1 0 の 6 i 乗個になる。即ち、発生可能な乱数列の数は、1 0 の 6 i 乗個になる。

このように、例えば、 $i = 1 0 0$ なら 1 0 の 6 0 0 乗個、 $i = 1 0 0 0$ なら 1 0 の 6 0 0 0 乗個といった具合に、いくらでも増すことができる。

【0031】

(4 - 6 . マスクパターン法を用いた暗号化システム)

上述の (1 - 2 . 暗号化の概要) で説明したように、ストリーム暗号方式による暗号化および復号の手法は、次のとおりである。

(1) 暗号化

平文 XOR 乱数列 = 暗文 ... (3 . 1)

(2) 復号

暗文 XOR 乱数列 = 平文 ... (3 . 2)

ここで、本実施形態のマスクパターンを適用して得られた乱数列を、演算 (3 . 1) および (3 . 2) の乱数列として用いることにより、入力した平文の暗号化および復号を行なうことができる。

【0032】

(4 - 7 . マスクパターン法の効果)

本実施形態のマスクパターン法を用いることにより、発生可能な暗号化鍵の数を必要なだけ増やすことができ (暗号化鍵空間の拡張)、安全なストリーム暗号方式を実現することができる。例えば発明者らの既存技術である CNN (非特許文献 1 , 特許文献 1 , 特許文献 2 を参照) に対して、上述のマスクパターン法を適用すると、暗号鍵空間を標準で $1 0 ^ { 4 8 }$ 倍に改良することができた。これにより解読に要する時間を $1 0 ^ { 2 9 }$ 年以上に延ばすことができる。もともと、発明者らの既存技術による暗号化方法は、米国の次世代標準暗号である R i j n d a e l より高速かつ堅牢であるが、本実施形態の手法によれば、これをさらに堅牢化できる。また、現在の米国標準である D E S などの暗号系は、数十時間で解読可能な暗号であるのに対して、本技術を用いると理論的には解読できない暗号系を構成できる点が優位である。

【図面の簡単な説明】

【0033】

【図 1】 (a) CNN のニューロンモデルである。 (b) ニューロン 4 個から構成されるカオス・ニューラルネットワーク (C - 4 n n) のモデル図である。

【図 2】 (a) 本実施形態のマスクパターン法の処理の流れを示す図である。 (b) CNN 出力から擬似乱数を取り出す処理を示す図である。

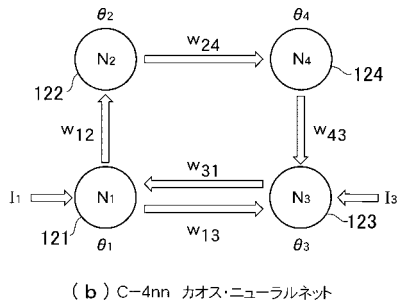
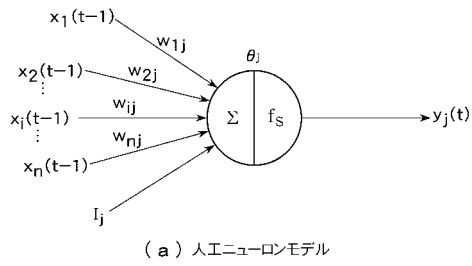
10

20

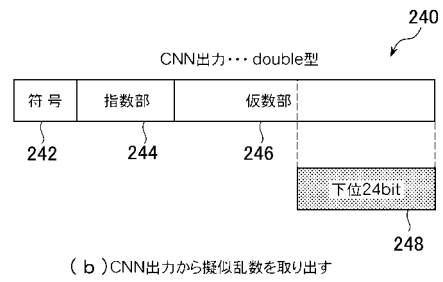
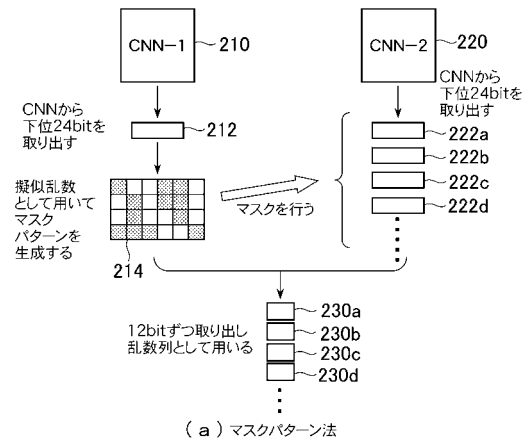
30

40

【 図 1 】



【 図 2 】



フロントページの続き

(56)参考文献 特開2003-076272(JP,A)
特開平10-240500(JP,A)
特開平11-340799(JP,A)
特開2003-032244(JP,A)
特開2002-366029(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06F 7/58
G09C 1/00