(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第7699791号
(P7699791)
(45)発行日　令和7年6月30日(2025.6.30)　(24)登録日　令和7年6月20日(2025.6.20)

(51)Int.Cl.　　FI
G06N 3/063 (2023.01)　G06N 3/063
G06F 9/50 (2006.01)　G06F 9/50　150D

請求項の数 5　(全 18 頁)

(21)出願番号　特願2020-194733(P2020-194733)
(22)出願日　令和2年11月24日(2020.11.24)
(65)公開番号　特開2022-83341(P2022-83341A)
(43)公開日　令和4年6月3日(2022.6.3)
審査請求日　令和5年10月31日(2023.10.31)

(73)特許権者　506301140
公立大学法人会津大学
福島県会津若松市一箕町大字鶴賀字上居合
９０番地
(74)代理人　100094525
弁理士　土井　健二
(74)代理人　100094514
弁理士　林　恒徳
(72)発明者　ベン　アブダラ　アブデラゼク
福島県会津若松市一箕町大字鶴賀字上居合
９０番地　公立大学法人会津大学内
(72)発明者　フアンクン　ホアン
福島県会津若松市一箕町大字鶴賀字上居合
９０番地　公立大学法人会津大学内

最終頁に続く

(54)【発明の名称】ＡＩプロセッサ

(57)【特許請求の範囲】
【請求項１】
複数の演算コアを有し、
前記複数の演算コアの少なくともいずれかが、畳み込み層と全結合層とを有するＣＮＮ（Ｃｏｎｖｏｌｕｔｉｏｎａｌ　Ｎｅｕｒａｌ　Ｎｅｔｗｏｒｋ）の機械学習モデルに含まれる複数のニューロンのそれぞれに対応付けられた計算プログラムを分割して前記複数の演算コアのそれぞれに割り当てるマッピング処理を実行し、
前記複数の演算コアのそれぞれが、前記マッピング処理によって割り当てられた前記計算プログラムを実行し、
前記マッピング処理では、前記複数の演算コア間における通信コストが所定の閾値以下になるように、遺伝的アルゴリズムによって前記計算プログラムを前記複数の演算コアに割り当て、さらに、
前記機械学習モデルにおけるパラメータと他の機械学習モデルにおけるパラメータとから生成されたグローバルパラメータを用いることによって、前記機械学習モデルにおけるパラメータを更新し、
前記機械学習モデルは、個人情報を含む画像データの入力に伴って前記個人情報に対応する人物が所定の状態にあるか否かについての情報を出力する機械学習モデルであり、
前記マッピング処理では、前記複数の演算コアに含まれる第１演算コアが、前記マッピング処理の結果を示すマッピングテーブルを生成し、
前記機械学習モデルにおけるパラメータを更新する処理では、前記第１演算コアが、前

記マッピングテーブルを参照し、前記複数の演算コアのうちの複数の他の演算コアに対して前記機械学習モデルにおけるパラメータを送信することによって、前記機械学習モデルにおけるパラメータを更新する、
ことを特徴とするＡＩプロセッサ。
【請求項２】
請求項１において、
前記画像データは、患者が映る画像データであり、
前記機械学習モデルは、前記画像データの入力に伴って前記患者が前記所定の状態にあるか否かについての情報を出力する機械学習モデルである、
ことを特徴とするＡＩプロセッサ。
【請求項３】
請求項１において、
前記ＣＮＮは、プーリング層をさらに有する、
ことを特徴とするＡＩプロセッサ。
【請求項４】
請求項１において、
前記マッピング処理では、
前記計算プログラムについての前記複数の演算コアに対するＮ個のマッピング解をランダムに生成し、
前記Ｎ個のマッピング解のそれぞれを採用した場合における前記通信コストを算出し、
前記Ｎ個のマッピング解から、算出した前記通信コストが小さい順にＭ個のマッピング解を特定し、
前記Ｍ個のマッピング解を交差させることによってＮ－Ｍ個の新たなマッピング解を生成し、
前記Ｍ個のマッピング解と前記Ｎ－Ｍ個の新たなマッピング解とを含むＮ個の新たなマッピング解において突然変異を発生させ、
前記突然変異を発生させた前記Ｎ個の新たなマッピング解のそれぞれを採用した場合における前記通信コストを再度算出し、
前記Ｎ個の新たなマッピング解のうち、再度算出した前記通信コストが最小である特定のマッピング解を特定し、
前記特定のマッピング解の前記通信コストが前記所定の閾値以下であるか否かを判定し、
前記特定のマッピング解の前記通信コストが前記所定の閾値以下であると判定した場合、前記計算プログラムを前記特定のマッピング解に従って前記複数の演算コアに割り当てる、
ことを特徴とするＡＩプロセッサ。
【請求項５】
請求項４において、
前記マッピング処理では、
前記特定のマッピング解の前記通信コストが前記所定の閾値以下でないと判定した場合、前記Ｎ個の新たなマッピング解について、前記通信コストを算出する処理と、前記Ｍ個のマッピング解を特定する処理と、前記Ｎ－Ｍ個の新たなマッピング解を生成する処理と、前記突然変異を発生させる処理と、前記通信コストを再度算出する処理と、前記特定のマッピング解を特定する処理と、前記通信コストが所定の閾値であるか否かを判定する処理とを再度行う、
ことを特徴とするＡＩプロセッサ。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、ＡＩプロセッサに関する。

【背景技術】
【０００２】
昨今、ＳＡＲＳ－ＣｏＶ２ウイルス（以下、新型コロナウイルス）によって引き起こされるＣＯＶＩＤ－１９（以下、新型コロナウイルス感染症とも呼ぶ）が流行している。この新型コロナウイルスに感染しているか否かの検査を行うための標準的な方法は、例えば、患者由来の採取サンプルを使用する逆転写ポリメラーゼ連鎖反応（ＲＴ－ＰＣＲ：Ｒｅｖｅｒｓｅ Ｔｒａｎｓｃｒｉｐｔｉｏｎ Ｐｏｌｙｍｅｒａｓｅ Ｃｈａｉｎ Ｒｅａｃｔｉｏｎ）であり、６０（％）から９７（％）程度の感度を有している。また、新型コロナウイルスに感染しているか否かの検査を行うための別の方法としては、例えば、患者の肺を撮影したＸ線画像の解析があり、８０（％）から９０（％）程度の精度を有している。
【０００３】
ここで、上記のようなＸ線画像の解析が行われる場合、医師は、患者のＸ線画像を手動で１枚ずつ診断する必要があり、非効率的な診断処置の原因となっている。
【０００４】
そこで、各病院では、例えば、患者が新型コロナウイルスに感染しているか否かの診断や患者が肺炎になっているか否かの診断を機械学習モデル（以下、診断モデルとも呼ぶ）に行わせることによって、多くの患者についての効率的な診断処置を行う場合がある。
【先行技術文献】
【特許文献】
【０００５】
【特許文献１】米国特許出願公開２０２０００２６９９２Ａ１号明細書
【非特許文献】
【０００６】
【非特許文献１】Kun-Chih (Jimmy) Chen, Ting-Yi Wang, “NN-Noxim: High-Level Cycle-Accurate NoC-based Neural Networks Simulator”, 2018 11th International Workshop on Network on Chip Architectures (NoCArc).
【非特許文献２】Xiaoxiao Liu, Wei Wen, Xuehai Qian, Hai Li, Yiran Chen, “Neu-NoC: A High-efficient Interconnection Network for Accelerated Neuromorphic Systems”, 2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC)
【非特許文献３】Kun-Chih (Jimmy) Chen, Masoumeh Ebrahimi, Ting-Yi Wang, Yuch-Chi Yang, “NoC-based DNN Accelerator: A Future Design Paradigm”, NOCS ’19, October 17-18, 2019, New York, NY, USA.
【発明の概要】
【発明が解決しようとする課題】
【０００７】
しかしながら、上記のような診断モデルを用いた診断は、診断モデルを実行するコンピュータの性能によって効率性が大きく異なる。そのため、病院等の医療現場では、例えば、上記のような診断モデルを用いた診断をより効率的に行うことが可能なＡＩ（Ａｒｔｉｆｉｃｉａｌ Ｉｎｔｅｌｌｉｇｅｎｃｅ）プロセッサの開発が求められている。
【０００８】
そこで、本発明の目的は、診断モデルを用いた診断をより効率的に行うＡＩプロセッサを提供することにある。
【課題を解決するための手段】
【０００９】
本発明の一態様におけるＡＩプロセッサは、複数の演算コアを有し、前記複数の演算コアの少なくともいずれかが、畳み込み層と全結合層とを有するＣＮＮ（Ｃｏｎｖｏｌｕｔｉｏｎａｌ Ｎｅｕｒａｌ Ｎｅｔｗｏｒｋ）の機械学習モデルに含まれる複数のニューロンのそれぞれに対応付けられた計算プログラムを分割して前記複数の演算コアのそれぞれに割り当てるマッピング処理を実行し、前記複数の演算コアのそれぞれが、前記マッピ

ング処理によって割り当てられた前記計算プログラムを実行し、前記マッピング処理では、前記複数の演算コア間における通信コストが所定の閾値以下になるように、遺伝的アルゴリズムによって前記計算プログラムを前記複数の演算コアに割り当てる。

【発明の効果】

【００１０】

本発明の一態様によれば、診断モデルを用いた診断をより効率的に行うことが可能になる。

【図面の簡単な説明】

【００１１】

【図１】図１は、従来の情報処理システム２００の構成について説明する図である。

【図２】図２は、本実施の形態における情報処理システム１００の構成について説明する図である。

【図３】図３は、本実施の形態におけるユーザインターフェースの例について説明する図である。

【図４】図４は、本実施の形態におけるユーザインターフェースの例について説明する図である。

【図５】図５は、本実施の形態におけるユーザインターフェースの例について説明する図である。

【図６】図６は、診断モデルＭＤの具体例について説明する図である。

【図７】図７は、診断モデルＭＤにおける更新処理について説明する図である。。

【図８】図８は、本実施の形態におけるＡＩプロセッサＰＲの具体例を説明する図である。

【図９】図９は、本実施の形態におけるマッピング処理について説明する図である。

【図１０】図１０は、本実施の形態におけるマッピング処理について説明する図である。

【図１１】図１１は、本実施の形態におけるマッピング処理について説明する図である。

【図１２】図１２は、本実施の形態におけるマッピング処理について説明する図である。

【図１３】図１３は、本実施の形態におけるマッピング処理について説明する図である。

【図１４】図１４は、本実施の形態におけるマッピング処理について説明する図である。

【図１５】図１５は、本実施の形態におけるマッピング処理について説明する図である。

【図１６】図１６は、本実施の形態におけるマッピング処理について説明する図である。

【図１７】図１７は、本実施の形態におけるマッピング処理について説明する図である。

【図１８】図１８は、本実施の形態におけるマッピング処理について説明する図である。

【図１９】図１９は、本実施の形態におけるマッピング処理について説明する図である。

【図２０】図２０は、本実施の形態におけるマッピング処理について説明する図である。

【図２１】図２１は、本実施の形態におけるマッピング処理について説明する図である。

【発明を実施するための形態】

【００１２】

以下、図面を参照して本発明の実施の形態について説明する。各実施の形態は、本発明のより良い理解のために準備されている。ただし、かかる実施の形態は、本発明の技術的範囲を限定するものではない。また、本発明の範囲は、特許請求の範囲及びこれと同等のものを網羅している。

【００１３】

［従来の情報処理システム］

初めに、従来の情報処理システム２００について説明を行う。図１は、従来の情報処理システム２００の構成について説明する図である。なお、以下、３つの病院（病院１１、病院１２及び病院１ｎ）が存在する場合について説明を行うが、３つ以外の数の病院が存在するものであってもよい。

【００１４】

図１に示す情報処理システム２００において、病院１１では、医師が患者のＸ線画像２１を分析することによって診断を行う。そして、医師は、例えば、病院１１に設定された

情報処理装置（図示しない）を用いることにより、各患者が新型コロナウイルス感染症に感染しているか否かの診断結果（例えば、感染者数、各感染者の重篤度及び死者数等）を示す患者情報３１を定期的に生成し、生成した患者情報３１を政府の情報処理装置４に送信する。

【００１５】

同様に、図１に示す例において、病院１２では、医師が患者のＸ線画像２２についての診断結果を示す患者情報３２を生成し、生成した患者情報３２を政府の情報処理装置４に送信する。また、病院１ｎでは、医師が患者のＸ線画像２ｎについての診断結果を示す患者情報３ｎを生成し、生成した患者情報３ｎを政府の情報処理装置４に送信する。

【００１６】

ここで、情報処理システム２００では、病院１１、病院１２及び病院１ｎのそれぞれが他の病院と協調することができない。この場合、各医師は、Ｘ線画像についての診断を効率的に行うことができない。

【００１７】

［本実施の形態における情報処理システム］

次に、本実施の形態における情報処理システム１００について説明を行う。図２は、本実施の形態における情報処理システム１００の構成について説明する図である。また、図３から図５は、本実施の形態におけるユーザインターフェースの例について説明する図である。

【００１８】

図２に示す情報処理システム１００において、病院１１では、患者の携帯端末５１から送信されたＸ線画像を病院１１に設置された診断システム６１に入力する。検知システム６１は、例えば、携帯端末５１から送信されたＸ線画像が新型コロナウイルスに感染した患者の画像である否かを判定する診断モデルを実行する情報処理装置である。そして、診断システム６１は、携帯端末５１から送信されたＸ線画像についての診断結果を携帯端末５１に送信する。

【００１９】

同様に、図２に示す例において、病院１２に設置された診断システム６２では、携帯端末５２から送信されたＸ線画像についての診断を行い、その診断結果を携帯端末５２に送信する。また、病院１ｎに設置された診断システム６ｎでは、携帯端末５ｎから送信されたＸ線画像についての診断を行い、その診断結果を携帯端末５ｎに送信する。

【００２０】

これにより、病院１１、病院１２及び病院１ｎでは、Ｘ線画像についての診断に伴う医師の負担を軽減させることが可能になる。また、病院１１、病院１２及び病院１ｎでは、Ｘ線画像についての誤った診断の発生を防止することが可能になる。また、各医師は、例えば、図３に示すユーザインターフェースＵ１を閲覧することにより、各患者の診断状況をリアルタイムで確認することが可能になる。

【００２１】

また、病院１１、病院１２及び病院１ｎでは、診断モデルによる診断を行うことでＸ線画像についての診断を迅速に行うことが可能になり、携帯端末５１、携帯端末５２及び携帯端末５ｎに対して診断結果を迅速に通知することが可能になる。そのため、各患者は、例えば、図４に示すユーザインターフェースＵ２を閲覧することにより、Ｘ線画像についての診断結果を迅速に確認することが可能になる。

【００２２】

そして、診断システム６１は、診断モデルによる診断結果（例えば、感染者数、各感染者の重篤度及び死者数等）を示す患者情報をリアルタイムに生成し、生成した患者情報を政府の情報処理装置４に送信する。

【００２３】

同様に、図２に示す例において、診断システム６２及び診断システム６ｎのそれぞれは、診断モデルによる診断結果を示す患者情報をリアルタイムに生成して政府の情報処理装

置４に送信する。

【００２４】

具体的に、診断システム６１、診断システム６２及び診断システム６ｎのそれぞれは、例えば、クラウドサーバ７に対して患者情報を送信する。そして、クラウドサーバ７は、例えば、診断システム６１、診断システム６２及び診断システム６ｎからリアルタイム情報８を生成し、生成したリアルタイム情報８を政府の情報処理装置４に送信する。

【００２５】

これにより、病院１１、病院１２及び病院１ｎは、政府の情報処理装置４に対して、最新の患者情報を迅速に送信することが可能になる。そのため、政府の担当者は、例えば、図５に示すユーザインターフェースＵ３を閲覧することにより、各患者についての診断結果を迅速に確認することが可能になる。

【００２６】

［診断モデルの具体例］

次に、診断システム６１、診断システム６２及び診断システム６ｎにおいて実行される診断モデルＭＤの具体例について説明を行う。図６は、診断モデルＭＤの具体例について説明する図である。

【００２７】

図６に示す診断モデルＭＤでは、入力層からＸ線画像が入力された場合、Ｘ線画像が新型コロナウイルスに感染していないこと（肺炎になっていないこと）を示すカテゴリーである「Ｎｏｒｍａｌ」、または、Ｘ線画像が新型コロナウイルスに感染している疑いがあること（肺炎になっている疑いがあること）を示すカテゴリーである「Ｓｕｓｐｅｃｔ」を出力する。

【００２８】

具体的に、図６に示す診断モデルＭＤは、例えば、ソフトマックス関数を用いることによってＸ線画像の分類を行う。また、図６に示す診断モデルＭＤは、例えば、以下の式（１）で表されるクロスエントロピー損失を損失関数Ｅとして用いる。

【００２９】

【数１】

$$E(\boldsymbol{w}_o, \boldsymbol{b}_o) = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{y}_i \times \log\left(\boldsymbol{y}_i'(\boldsymbol{w}_o, \boldsymbol{b}_o)\right)$$

【００３０】

式（１）においてｗ₀及びｂ₀は、診断モデルＭＤにおけるパラメータであり、ｙ及びｙ′のそれぞれは、実際のラベル（正解のラベル）及び予測されたラベルをそれぞれ示す。図６に示す診断モデルＭＤでは、式（１）を利用することによって、実際のラベルと予測されたラベルとの間における損失関数Ｅを算出する。そして、診断モデルＭＤでは、確率的勾配縮小アルゴリズムや逆伝搬アルゴリズムを用いた損失関数Ｅの最小化が行われ、さらに、パラメータの最適化が行われる。

【００３１】

［診断モデルの更新］

次に、診断モデルＭＤの更新を行う処理（以下、更新処理とも呼ぶ）について説明を行う。図７は、診断モデルＭＤにおける更新処理について説明する図である。具体的に、図７（Ａ）は、更新処理を説明するフローチャート図である。また、図７（Ｂ）は、更新処理の実行時における情報の送受信を説明する図である。なお、以下、図２で説明した診断システム６１とクラウドサーバ７との間において行われる更新処理について説明を行う。

【００３２】

診断システム６１は、病院１１の患者から送信されたＸ線画像を含むデータセットを用いることによって、病院１１における診断モデルＭＤの生成を予め行う。そして、診断システム６１は、生成した診断モデルＭＤについてのパラメータ（例えば、以下の式（２）に示す勾配∇ｇＬ）をクラウドサーバ７に送信する（Ｓ１）。

【００３３】
【数２】

$$\nabla g_L = \frac{\delta E(W)}{\delta W}$$

【００３４】
続いて、クラウドサーバ７は、例えば、診断システム６１、診断システム６２及び診断システム６ｎのそれぞれからパラメータを受信したことに応じて、受信したパラメータのそれぞれからグローバルパラメータ（例えば、以下の式（３）に示すグローバル勾配∇ｇＧ）を算出する（Ｓ２）。
【００３５】
さらに、クラウドサーバ７は、算出したグローバルパラメータを診断システム６１、診断システム６２及び診断システム６ｎのそれぞれに送信する（Ｓ３）。
【００３６】
【数３】

$$\nabla g_G = \frac{1}{n}\sum_{i=1}^{n} \nabla g_L^i$$

【００３７】
その後、診断システム６１は、クラウドサーバ７から送信されたグローバルパラメータを受信したことに応じて、以下の式（４）及び式（５）に示すように、病院１１における診断モデルＭＤのパラメータを更新する（Ｓ４）。
【００３８】
【数４】

$$\mathbf{W}^{r+1} = \mathbf{W}^{r} - \eta \nabla g_G$$

【００３９】
【数５】

$$\mathbf{b}^{r+1} = \mathbf{b}^{r} - \eta \nabla g_G$$

【００４０】
式（４）及び式（５）において、Ｗｒ及びｂｒのそれぞれは、r回目に行われたＳ１からＳ４の処理（ｒ番目のトレーニングラウンド）における重み及びバイアスをそれぞれ示している。また、式（４）及び（５）において、ηは学習率を示している。
【００４１】
すなわち、情報処理システム１００では、フェデレーテッドモデルラーニング（ＦＭＬ：Ｆｅｄｅｒａｔｅｄ　Ｍｏｄｅｌ　Ｌａｅｒｎｉｎｇ）によって、各病院の診断モデルＭＤ（診断モデルＭＤのパラメータ）の生成を行う。
【００４２】
これにより、情報処理システム１００は、各病院の患者についての個人情報を含むデータセットを他の病院等の外部に送信することなく、各病院の診断モデルＭＤの精度を高めることが可能になる。そのため、各病院の診断モデルＭＤは、各患者のプライバシーを守りつつ、新型コロナウイルスの診断を精度良く行うことが可能になる。
【００４３】
なお、Ｓ１からＳ４の処理は、各病院の診断モデルＭＤのそれぞれの判定精度が必要な条件を上回るまで繰り返し行われるものであってよい。
【００４４】
また、各病院の診断モデルＭＤは、例えば、診断システム６１、診断システム６２及び診断システム６ｎ以外のコンピュータ（例えば、後述する図８に示すホストコンピュータＨＣ）において生成されるものであってもよい。

【００４５】
［診断モデルのマッピング］
次に、従来のＡＩプロセッサ（以下、ＡＩチップとも呼ぶ）ＰＲに対する診断モデルＭＤのマッピングを行う処理（以下、マッピング処理とも呼ぶ）について説明を行う。
【００４６】
ＡＩプロセッサＰＲは、例えば、診断システム６１、診断システム６２及び診断システム６ｎに搭載されたプロセッサである。そして、ＡＩプロセッサＰＲに対する診断モデルＭＤのマッピングが行われる場合、ＡＩプロセッサＰＲでは、例えば、診断モデルＭＤを構成するニューロンを複数のグループにクラスタリングした後、ＡＩプロセッサＰＲに含まれる複数の演算コアのそれぞれに対する各グループのマッピングを行う。
【００４７】
しかしながら、従来のマッピング処理では、複数の演算コア間における通信コストについて考慮されていない場合がある。そのため、従来のマッピング処理では、マッピング処理が必要な時間内に終了しない場合があった。
【００４８】
また、ニューロンのクラスタリングと複数の演算コアへのマッピングは、一般的に、どちらもＮＰ困難な問題であり、多項式時間において最適に解決することができない場合がある。
【００４９】
そこで、本実施の形態におけるＡＩプロセッサＰＲでは、遺伝的アルゴリズムを用いることにより、複数の演算コア間における通信コストを抑えるように、診断モデルＭＤの各層を構成するニューロンのマッピングを行う。
【００５０】
［本実施の形態におけるＡＩプロセッサの具体例］
次に、本実施の形態におけるＡＩプロセッサＰＲの具体例について説明を行う。図８は、本実施の形態におけるＡＩプロセッサＰＲの具体例を説明する図である。以下、診断モデルＭＤがホストコンピュータＨＣにおいて生成されるものとして説明を行う。
【００５１】
図８に示すＡＩプロセッサＰＲは、１５個の演算コアを有している。具体的に、図８に示すＡＩプロセッサＰＲは、畳み込み層と対応付けられた１０個の演算コアＣと、全結合層と対応付けられた３個の演算コアＦを有している。また、各演算コアは、それぞれルータＲと接続している。なお、以下、ＡＩプロセッサＰＲが１５個の演算コアを有している場合について説明を行うが、ＡＩプロセッサＰＲは、これ以外の数の演算コアを有するものであってもよい。
【００５２】
また、図８に示すＡＩプロセッサＰＲは、プーリング層やアクティベーション機能と対応付けられた１個の演算コアＵと、各演算コアに対して重み係数を送信する１個の演算コアＩ／Ｏを有している。なお、演算コアＩ／Ｏは、例えば、他のＡＩプロセッサＰＲとの間の通信（チップ間通信）や各ＡＩプロセッサＰＲの制御を行うホストコンピュータＨＣとの通信を行うものであってもよい。
【００５３】
また、図８に示すＡＩプロセッサＰＲは、ＡＩプロセッサＰＲに対する入力を記憶するオンチップメモリＭ（以下、単にメモリＭとも呼ぶ）を有する。演算コアＣ等の各演算コアは、メモリＭから入力をロードして処理を実行する。そして、各演算コアは、次の層に対応する処理の実行を可能とするために、処理の実行に伴う各演算コアの出力をメモリＭに記憶する。
【００５４】
さらに、図８に示すＡＩプロセッサＰＲは、演算コア及びメモリＭのそれぞれに対応するルータＲと、Ｅｘｔｅｒｎａｌ　ＤＲＡＭ（Ｄｙｎａｍａｉｃ　Ｒａｍｄｏｍ　Ａｃｃｅｓｓ　Ｍｅｍｏｒｙ）とを有する。

【００５５】
［本実施の形態におけるマッピング処理］
次に、本実施の形態におけるマッピング処理について説明を行う。図９は、本実施の形態におけるマッピング処理について説明するフローチャート図である。図１０から図２１は、本実施の形態におけるマッピング処理について説明する図である。
【００５６】
ＡＩプロセッサＰＲの演算コアＵは、診断モデルＭＤ（ニューラルネットワーク）を構成するニューロンについてのＮ個のマッピング解をランダムに決定する（Ｓ１１）。
【００５７】
具体的に、ＡＩプロセッサＰＲがそれぞれ８個のニューロンと対応付けることが可能な４個の演算コアを有しており、かつ、診断モデルＭＤを構成するニューロンの数が３０個である場合、マッピング解の組合せは、３２個のニューロンにおいて３０個のニューロンを配置する組合せである４９６通りになる。そのため、演算コアＵは、この場合、その４９６通りのうちのＮ通りに対応するＮ個のマッピング解をランダムに決定する。
【００５８】
以下、ＡＩプロセッサＰＲがそれぞれ８個のニューロンと対応付けることが可能な４個の演算コアを有しており、かつ、診断モデルＭＤを構成するニューロンの数が３０個であるものとして説明を行う。さらに、診断モデルＭＤが１０個のニューロンからなる層Ｌ１、層Ｌ２及び層Ｌ３をそれぞれ有するニューラルネットワークであるものとして説明を行う。
【００５９】
［マッピング結果の具体例］
次に、本実施の形態におけるマッピング結果の具体例について説明を行う。図１０は、各演算コアにマッピングされたニューロンの数を示す図である。また、図１１は、各演算コアにマッピングされたニューロンの識別情報を示す図である。
【００６０】
図１０及び図１１に示す例は、１行目１列目の演算コアと対応付けられているニューロンが、層Ｌ１に対応する３つのニューロン（ニューロン２、３及び５）と、層Ｌ２に対応する２つのニューロン（ニューロン１１及び１２）と、層Ｌ３に対応する３つのニューロン（ニューロン２２、２３及び２８）とであることを示している。
【００６１】
また、図１０及び図１１に示す例は、１行目２列目の演算コアと対応付けられているニューロンが、層Ｌ１に対応する３つのニューロン（ニューロン６、７及び８）と、層Ｌ２に対応する１つのニューロン（ニューロン１３）と、層Ｌ３に対応する３つのニューロン（ニューロン２１、２９及び３０）とであることを示している。なお、図１０及び図１１に示す例は、１行目２列目の演算コアにおいてさらに対応可能なニューロンの数（Ｆ）が１であることを示している。
【００６２】
また、図１０及び図１１に示す例は、２行目１列目の演算コアと対応付けられているニューロンが、層Ｌ１に対応する２つのニューロン（ニューロン１及び４）と、層Ｌ２に対応する３つのニューロン（ニューロン１５、１７及び１９）と、層Ｌ３に対応する３つのニューロン（ニューロン２４、２５及び２６）とであることを示している。
【００６３】
また、図１０及び図１１に示す例は、２行目２列目の演算コアと対応付けられているニューロンが、層Ｌ１に対応する２つのニューロン（ニューロン９及び１０）と、層Ｌ２に対応する４つのニューロン（ニューロン１４、１６、１８及び２０）と、層Ｌ３に対応する１つのニューロン（ニューロン２７）とであることを示している。なお、図１０及び図１１に示す例は、２行目２列目の演算コアにおいてさらに対応可能なニューロンの数（Ｆ）が１であることを示している。
【００６４】

図９に戻り、演算コアＵは、Ｓ１１の処理で決定したＮ個のマッピング解から不適切なマッピング解を削除する（Ｓ１２）。

【００６５】

具体的に、演算コアＵは、例えば、図１２に示すように、９個のニューロンと対応付けられた演算コア（１行目１列目の演算コア）が存在するマッピング解を削除する。また、演算コアＵは、例えば、図１３に示すように、同一のニューロン（ニューロン８）が複数の演算コアと対応付けられたマッピング解を削除する。

【００６６】

そして、演算コアＵは、Ｓ１２の処理で削除されなかったマッピング解のそれぞれに対応する通信コストを算出する（Ｓ１３）。

【００６７】

具体的に、各マッピング解の通信コストは、以下の式（６）によって表現される。

【００６８】

【数６】

$$F_{cost} = \sum_{i=0,j=0}^{W} d_{i,j} \times c_{i,j}$$

【００６９】

式（６）において、$d_{i,j}$は、ニューロンｉとニューロンｊとの間の距離を示しており、$c_{i,j}$は、ニューロンｉとニューロンｊとの間の接続状況を示している。

【００７０】

具体的に、$d_{i,j}$は、例えば、ニューロンｉと対応付けられている演算コアとニューロンｊと対応付けられている演算コアとの間に存在するルータＲの数に１を加算した値である。また、$c_{i,j}$は、例えば、ニューロンｉとニューロンｊとが直接接続している場合に１になり、ニューロンｉとニューロンｊとが直接接続していない場合に０になる。

【００７１】

さらに具体的に、図１１で説明した例において、ニューロン１及びニューロン２は、それぞれ層Ｌ１と対応付けられている。そのため、この場合、$d_{1,2}$は１になり、$c_{1,2}$は０になる。また、図１１で説明した例において、ニューロン１は、層Ｌ１と対応付けられており、ニューロン１４は、層Ｌ２と対応付けられている。そのため、この場合、$d_{1,14}$は４になり、$c_{1,2}$は１になる。

【００７２】

続いて、演算コアＵは、Ｓ１２の処理で削除されなかったマッピング解のうち、Ｓ１３の処理で算出した通信コストが条件を満たすＭ個のマッピング解を特定する（Ｓ１４）。

【００７３】

具体的に、演算コアＵは、例えば、Ｓ１２の処理で削除されなかったマッピング解から、Ｓ１３の処理で算出した通信コストが高い順にＭ個のマッピング解を特定する。

【００７４】

その後、演算コアＵは、Ｓ１４の処理で特定したＭ個のマッピング解を交差（クロスオーバー）させることによってＮ－Ｍ個の新たなマッピング解を決定する（Ｓ１５）。

【００７５】

具体的に、演算コアＵは、例えば、図１４及び図１５に示す親１及び親２がＳ１４の処理で特定したＭ個のマッピング解に含まれている場合、図１６に示すように、親１及び親２のそれぞれの割合を５０（％）とした新たな子孫を作成する。

【００７６】

さらに具体的に、図１４に示す例において、親１の場合の１行目１列目の演算コアには、層Ｌ１に対応する３つのニューロンと、層Ｌ２に対応する２つのニューロンと、層Ｌ３に対応する３つのニューロンとが対応付けられている。また、図１５に示す例において、親２の場合の１行目１列目の演算コアには、層Ｌ１に対応する１つのニューロンと、層Ｌ２に対応する４つのニューロンと、層Ｌ３に対応する３つのニューロンとが対応付けられ

ている。そのため、この場合、新たな子孫の場合の１行目１列目の演算コアには、図１６に示すように、層Ｌ１に対応する２（３＊０．５＋１＊０．５）つのニューロンと、層Ｌ２に対応する３（４＊０．５＋２＊０．５）つのニューロンと、層Ｌ３に対応する３（３＊０．５＋３＊０．５）つのニューロンとが対応付けられる。

【００７７】

なお、例えば、図１７に示すように、各演算コアと対応付けられたニューロンの数に小数が含まれている場合、演算コアＵは、図１８に示すように、各演算コアと対応付けられたニューロンの数のそれぞれが整数になるように調整を行うものであってもよい。

【００７８】

図９に戻り、演算コアＵは、Ｎ個のマッピング解（Ｓ１４の処理で特定したＭ個のマッピング解とＳ１５の処理で決定したＮ－Ｍ個の新たなマッピング解との合計）において突然変異を発生させる（Ｓ１６）。

【００７９】

具体的に、演算コアＵは、図１９に示すように、例えば、図１８に示す複数の演算コアから１行目１列目の演算コアと２行目２列目の演算コアとを特定し、さらに、診断モデルＭＤを構成する複数の層から層Ｌ１と層Ｌ２とを特定する。そして、演算コアＵは、例えば、特定した１行目１列目の演算コアにおける層Ｌ１に対応付けられたニューロンの数である２と、２行目２列目の演算コアにおける層Ｌ２に対応付けられたニューロンの数である３とのうちの最小値である２を特定する。その後、演算コアＵは、例えば、１行目１列目の演算コアにおける層Ｌ１に対応付けられたニューロンの数である２から、最小値として特定した値である２を減算し、さらに、１行目１列目の演算コアにおける層Ｌ２に対応付けられたニューロンの数である３に対して、最小値として特定した値である２を加算する。

【００８０】

同様に、演算コアＵは、例えば、２行目２列目の演算コアにおける層Ｌ２に対応付けられたニューロンの数である３から２を減算し、さらに、２行目２列目の演算コアにおける層Ｌ１に対応付けられたニューロンの数である４に対して２を加算する。

【００８１】

そして、演算コアＵは、例えば、Ｓ１２の処理と同様に、Ｓ１６の処理が行われた後のＮ個のマッピング解が制約を満たしているか否かを判定する。その結果、制約を満たしていないマッピング解が存在した場合、演算コアＵは、存在したマッピング解を削除する。さらに、演算コアＵは、Ｓ１３の処理と同様に、削除されなかったマッピング解の通信コストを算出する（Ｓ１７）。

【００８２】

その後、Ｓ１７の処理で算出した通信コストのうちの最適なコスト（最小のコスト）が予め定められた条件を満たしているか否かを判定する（Ｓ１８）。

【００８３】

その結果、Ｓ１７の処理で算出した通信コストのうちの最適なコストが予め定められた条件を満たしていると判定した場合、演算コアＵは、マッピング処理を終了（正常終了）する。

【００８４】

一方、Ｓ１７の処理で算出した通信コストのうちの最適なコストが予め定められた条件を満たしていないと判定した場合、演算コアＵは、例えば、Ｓ１２以降の処理の実行回数（すなわち、世代数）が予め定められた所定回数に到達したか否かを判定する（Ｓ１９）。

【００８５】

その結果、Ｓ１２以降の処理の実行回数が予め定められた所定回数に到達していないと判定した場合、演算コアＵは、例えば、Ｓ１２以降の処理を再度行う。

【００８６】

一方、Ｓ１２以降の処理の実行回数が予め定められた所定回数に到達していると判定し

た場合、演算コアＵは、例えば、ニューロンのマッピングを終了（異常終了）する。

【００８７】

このように、本実施の形態におけるＡＩプロセッサＰＲは、複数の演算コアを有し、複数の演算コアの少なくともいずれかが、畳み込み層と全結合層とを有する診断モデルＭＤに含まれる複数のニューロンのそれぞれに対応付けられた計算プログラムを分割して複数の演算コアのそれぞれに割り当てるマッピング処理を実行する。

【００８８】

そして、本実施の形態におけるＡＩプロセッサは、複数の演算コアのそれぞれが、マッピング処理によって割り当てられた計算プログラムを実行する。

【００８９】

具体的に、本実施の形態におけるマッピング処理では、複数のコア間における通信コストが所定の閾値以下になるように、遺伝的アルゴリズムによって計算プログラムを複数の演算コアに割り当てる。

【００９０】

さらに具体的に、演算コアＵは、マッピング処理が正常終了した場合、マッピング処理の結果を示すマッピングテーブル（図示しない）を生成する。続いて、演算コアＵは、ホストコンピュータＨＣから診断モデルＭＤのパラメータをダウンロードする。さらに、演算コアＵは、マッピングテーブルを参照し、診断モデルＭＤのパラメータを演算コアＣ及び演算コアＦのそれぞれに送信する。また、演算コアＵは、マッピングテーブルを各ルータＲに対しても送信する。

【００９１】

その後、演算コアＵは、例えば、入力データ（例えば、患者のＸ線画像）が入力された場合、マッピングテーブルを参照し、入力データを最初の層に含まれるニューロンに対応する演算コアのそれぞれに送信する。そして、各ルータＲは、最初の層に対応する処理が完了したことに応じて、最初の層からの出力データを次の層に含まれるニューロンに対応する演算コアに送信する。さらに、各ルータＲは、最後の層に対応する処理が完了するまでの間、処理対象の層の次の層に含まれるニューロンに対応する演算コアに対する送信を繰り返し行う。そして、各ルータＲは、最後の層からの出力データ（診断モデルＭＤの出力データ）をＤＲＡＭに格納する。

【００９２】

すなわち、本実施の形態におけるＡＩプロセッサＰＲは、従来の方法よりも簡易なアルゴリズムであって、かつ、予測可能な時間内に結果を得ることが可能なアルゴリズムである遺伝的アルゴリズムを用いることによってマッピング処理を行う。

【００９３】

これにより、本実施の形態におけるＡＩプロセッサＰＲは、各ニューロンの演算コアへのマッピングが行われる際の通信コストを抑えることが可能になる。そのため、本実施の形態におけるＡＩプロセッサＰＲは、診断モデルを用いた診断をより効率的に行うことが可能になる。

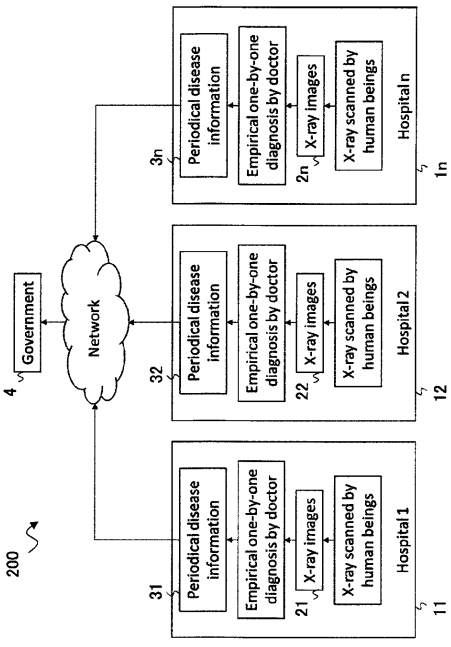
【符号の説明】

【００９４】

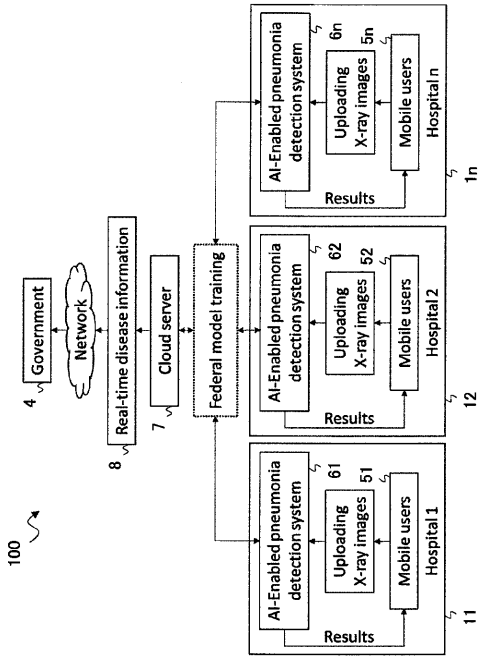
４：情報処理装置
７：クラウドサーバ
８：リアルタイム情報
１１：病院
１２：病院
１ｎ：病院
５１：携帯端末
５２：携帯端末
５ｎ：携帯端末
６１：診断システム

６２：診断システム
６ｎ：診断システム
１００：情報処理システム
ＭＤ：診断モデル
ＰＲ：ＡＩプロセッサ

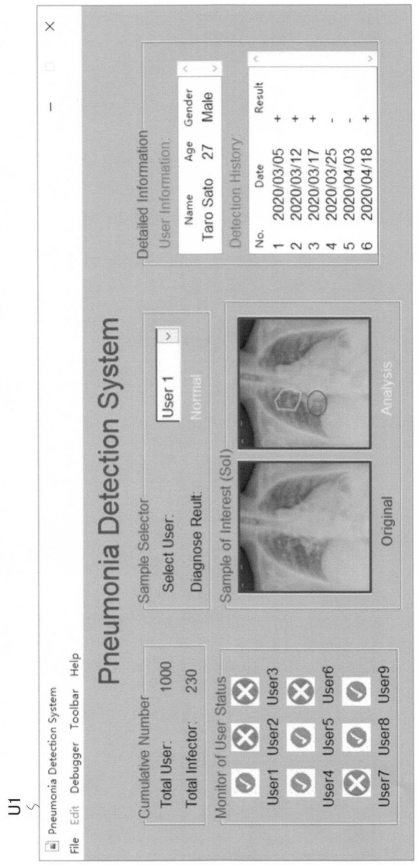
【図１】



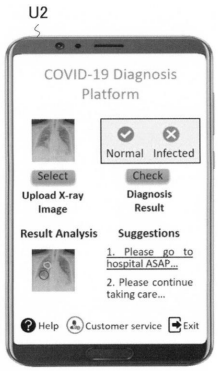
【図２】



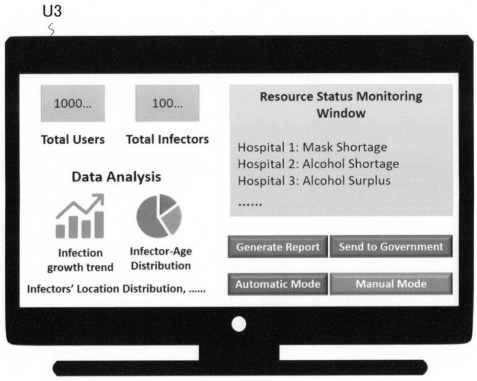
【図３】


U1
Pneumonia Detection System
File Edit Debugger Toolbar Help
Pneumonia Detection System
Cumulative Number
Total User: 1000
Total Infector: 230
Monitor of User Status
User1 User2 User3
User4 User5 User6
User7 User8 User9
Sample Selector
Select User:
User 1
Diagnose Reult:
Normal
Sample of Interest (SoI)
Original
Analysis
Detailed Information
User Information:
Name Age Gender
Taro Sato 27 Male
Detection History
No. Date Result
1 2020/03/05 +
2 2020/03/12 +
3 2020/03/17 +
4 2020/03/25 -
5 2020/04/03 -
6 2020/04/18 +

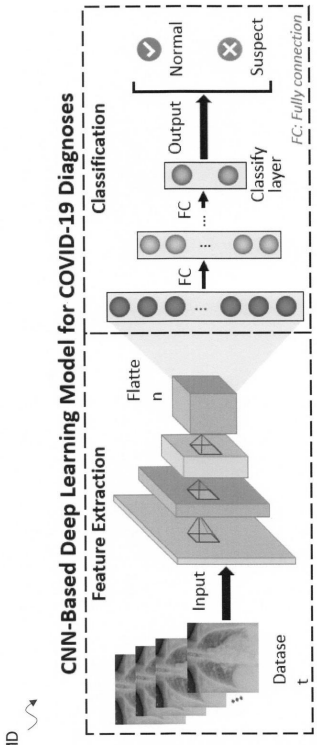
【図４】


U2
COVID-19 Diagnosis Platform
Normal Infected
Select
Upload X-ray Image
Check
Diagnosis Result
Result Analysis
Suggestions
1. Please go to hospital ASAP...
2. Please continue taking care...
Help
Customer service
Exit

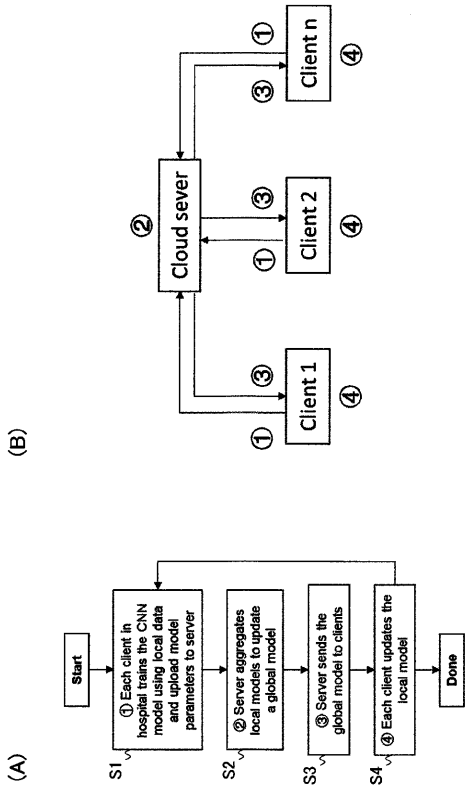
【図５】


U3
1000...
Total Users
100...
Total Infectors
Data Analysis
Infection growth trend
Infector-Age Distribution
Infectors' Location Distribution, ...
Resource Status Monitoring Window
Hospital 1: Mask Shortage
Hospital 2: Alcohol Shortage
Hospital 3: Alcohol Surplus
.....
Generate Report
Send to Government
Automatic Mode
Manual Mode

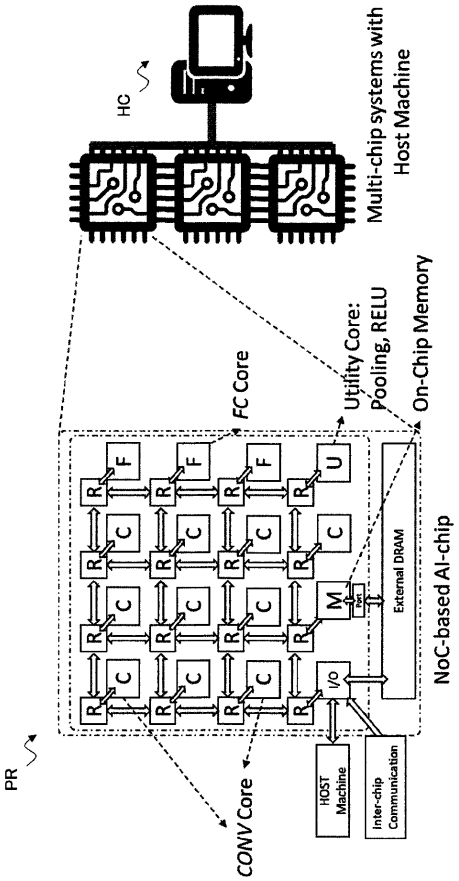
【図６】


MD
CNN-Based Deep Learning Model for COVID-19 Diagnoses
Feature Extraction
Classification
Datase t
Input
Flatte n
FC
FC
Classify layer
Output
Normal
Suspect
FC: Fully connection

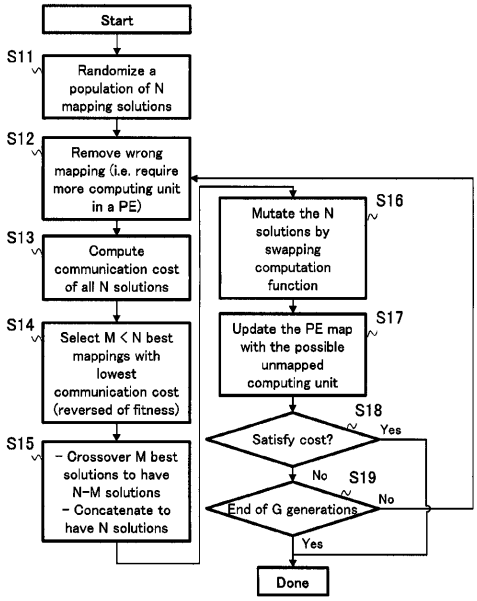
【図７】


(B)
Cloud sever
Client 1
Client 2
Client n
(A)
Start
① Each client in hospital trains the CNN model using local data and upload model parameters to server
② Server aggregates local models to update a global model
③ Server sends the global model to clients
④ Each client updates the local model
Done
S1
S2
S3
S4

【図８】



【図９】



【図１０】

Mapping-1		Column	
		1	2
Row	1	L1 : 3 neurons L2 : 2 neurons L3 : 3 neurons	L1 : 3 neurons L2 : 1 neurons L3 : 3 neurons F : 1
	2	L1 : 2 neurons L2 : 3 neurons L3 : 3 neurons	L1 : 2 neurons L2 : 4 neurons L3 : 1 neurons F : 1

【図１１】

Mapping-1		Column	
		1	2
Row	1	L1 : 2, 3, 5 L2 : 11, 12 L3 : 22, 23, 28	L1 : 6, 7, 8 L2 : 13 L3 : 21, 29, 30 F : 1
	2	L1 : 1, 4 L2 : 15, 17, 19 L3 : 24, 25, 26	L1 : 9, 10 L2 : 14, 16, 18, 20 L3 : 27 F : 1

【図１２】

Mapping-1		Column	
		1	2
Row	1	L1 : 2, 3, 5, 8 L2 : 11, 12 L3 : 22, 23, 28	L1 : 6, 7 L2 : 13 L3 : 21, 29, 30 F : 1
	2	L1 : 1, 4 L2 : 15, 17, 19 L3 : 24, 25, 26	L1 : 9, 10 L2 : 14, 16, 18, 20 L3 : 27 F : 1

【図１３】

Mapping-1		Column	
		1	2
Row	1	L1 : 2, 3, 8 L2 : 11, 12 L3 : 22, 23, 28	L1 : 6, 7, 8 L2 : 13 L3 : 21, 29, 30 F : 1
	2	L1 : 1, 4 L2 : 15, 17, 19 L3 : 24, 25, 26	L1 : 9, 10 L2 : 14, 16, 18, 20 L3 : 27 F : 1

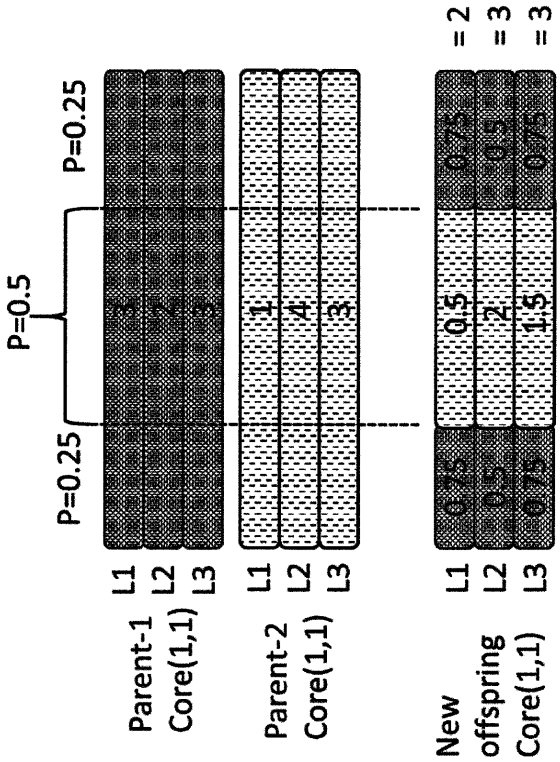
【図１４】

Parent-1		Column	
		1	2
Row	1	L1 : 3 neurons L2 : 2 neurons L3 : 3 neurons	L1 : 3 neurons L2 : 1 neurons L3 : 3 neurons F : 1
	2	L1 : 2 neurons L2 : 3 neurons L3 : 3 neurons	L1 : 2 neurons L2 : 4 neurons L3 : 1 neurons F : 1

【図１５】

Parent-2		Column	
		1	2
Row	1	L1 : 1 neurons L2 : 4 neurons L3 : 3 neurons	L1 : 0 neurons L2 : 3 neurons L3 : 3 neurons F : 2
	2	L1 : 4 neurons L2 : 1 neurons L3 : 3 neurons	L1 : 5 neurons L2 : 2 neurons L3 : 1 neurons

【図１６】



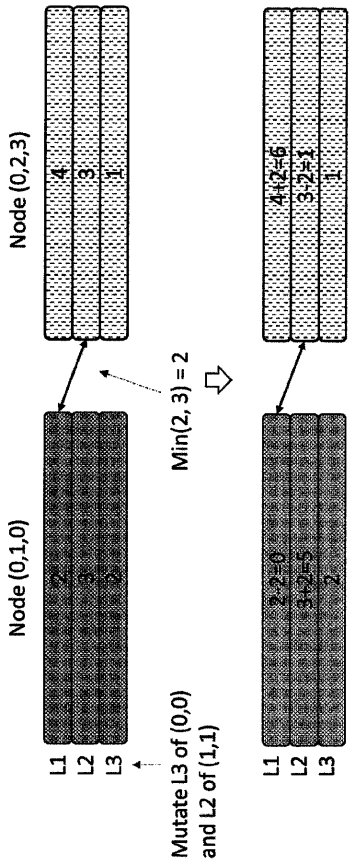
【図１７】

Parent-2		Column	
		1	2
Row	1	L1 : 2 neurons L2 : 3 neurons L3 : 3 neurons	L1 : 1.5 neurons L2 : 2 neurons L3 : 3 neurons F : 1.5
	2	L1 : 3 neurons L2 : 2 neurons L3 : 3 neurons	L1 : 3.5 neurons L2 : 3 neurons L3 : 1 neurons F : 0.5

【図１８】

Parent-2		Column	
		1	2
Row	1	L1 : 2 neurons L2 : 3 neurons L3 : 3 neurons	L1 : 1 neurons L2 : 2 neurons L3 : 3 neurons F : 2
	2	L1 : 3 neurons L2 : 2 neurons L3 : 3 neurons	L1 : 4 neurons L2 : 3 neurons L3 : 1 neurons F : 0

【図１９】



【図２０】

Parent-2		Column	
		1	2
Row	1	L1 : 0 neurons L2 : 5 neurons L3 : 3 neurons	L1 : 1 neurons L2 : 2 neurons L3 : 3 neurons F : 2
	2	L1 : 3 neurons L2 : 2 neurons L3 : 3 neurons	L1 : 6 neurons L2 : 1 neurons L3 : 1 neurons F : 0

【図２１】

Parent-2		Column	
		1	2
Row	1	L2 : 11, 12, 13, 17, 18 L3 : 28, 29, 30	L1 : 10 L2 : 15, 16 L3 : 25, 26, 27 F : 2
	2	L1 : 4, 5, 6 L2 : 14, 20 L3 : 21, 22, 23	L1 : 1, 2, 3, 7, 8, 9 L2 : 19 L3 : 24

フロントページの続き

(72)発明者　ダン　ナム　カイン
ベトナム社会主義共和国　ハノイ　カウ　ザイ　ディストリクト　スアン　トゥイ　ストリート　１４４　ベトナム国家大学ハノイ校内
(72)発明者　ジャンニン　ソン
オーストラリア連邦　３８００　ビクトリア　クレイトン　ウェリントン　ロード　モナシュ大学内

審査官　大倉　崚吾

(56)参考文献　米国特許出願公開第２０２０／００８９５３４（ＵＳ，Ａ１）
特開２００４－１８５２７１（ＪＰ，Ａ）
中国特許出願公開第１１１６５２８６３（ＣＮ，Ａ）
国際公開第２０２０／０５３８８７（WO，Ａ１）
特開２０２０－０４６８２１（ＪＰ，Ａ）
特開２０２０－１６０５６４（ＪＰ，Ａ）
国際公開第２０１９／２２０６９２（WO，Ａ１）
国際公開第２０２０／２２５８７９（WO，Ａ１）
特開２００３－０５８５２０（ＪＰ，Ａ）
樽林亮介　ほか，"データ駆動プロセッサによる実時間処理のためのプログラム割当手法"，電子情報通信学会論文誌，Vol. J86-D-I，No. 10，2003年10月，p. 721-732

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｎ　　３／００－９９／００
Ｇ０６Ｆ　　９／５０
Ｇ０６Ｆ　　９／５４