

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6700712号
(P6700712)

(45) 発行日 令和2年5月27日(2020.5.27)

(24) 登録日 令和2年5月8日(2020.5.8)

(51) Int.Cl.

G06F 17/10 (2006.01)

F 1

G06F 17/10

A

請求項の数 10 (全 37 頁)

(21) 出願番号 特願2015-207499 (P2015-207499)
 (22) 出願日 平成27年10月21日 (2015.10.21)
 (65) 公開番号 特開2017-79017 (P2017-79017A)
 (43) 公開日 平成29年4月27日 (2017.4.27)
 審査請求日 平成30年10月16日 (2018.10.16)

(73) 特許権者 000001007
 キヤノン株式会社
 東京都大田区下丸子3丁目30番2号
 (74) 代理人 100076428
 弁理士 大塚 康徳
 (74) 代理人 100115071
 弁理士 大塚 康弘
 (74) 代理人 100112508
 弁理士 高柳 司郎
 (74) 代理人 100116894
 弁理士 木村 秀二
 (74) 代理人 100130409
 弁理士 下山 治
 (74) 代理人 100134175
 弁理士 永川 行光

最終頁に続く

(54) 【発明の名称】畳み込み演算装置

(57) 【特許請求の範囲】

【請求項 1】

それぞれ、第1及び第2の入力に入力されたデータを乗算する複数の第1の乗算手段と、

前記複数の第1の乗算手段に対応して設けられ、それぞれ対応する前記第1の乗算手段の乗算結果を累積する複数の第1の累積加算手段と、

前記複数の第1の乗算手段のそれぞれの前記第1の入力に第1のデータとして2次元フィルターカーネルの係数データを分解した水平方向の係数データを供給する第1のデータ供給手段と、

前記複数の第1の乗算手段の前記第2の入力に複数の第2のデータとして入力画像データの水平方向のデータを供給する第2のデータ供給手段と、を有する第一の積和演算手段と、

それぞれ、第1及び第2の入力に入力されたデータを乗算する複数の第2の乗算手段と、

前記複数の第2の乗算手段に対応して設けられ、それぞれ対応する前記第2の乗算手段の乗算結果を累積する複数の第2の累積加算手段と、

前記複数の第2の乗算手段のそれぞれの前記第1の入力に第3のデータとして前記2次元フィルターカーネルの係数データを分解した垂直方向の係数データを供給する第3のデータ供給手段と、

前記複数の第1の累積加算手段のそれぞれの出力から前記2次元フィルターカーネルの垂

10

20

直方向の列単位で必要な参照データをリングバッファにロードして一括して保持し、当該各ロードした参照データを動作クロックに応じてリング状にシフトして複数の第4のデータとして前記複数の第2の乗算手段の前記第2の入力に供給する第4のデータ供給手段と、を有する第二の積和演算手段と、

前記第一及び第二の積和演算手段による積和演算処理を並行して実行するように制御する制御手段と、を有し、

前記複数の第2の累積加算手段の出力として、前記入力画像データに対する前記2次元フィルタカーネルによる畳み込み演算の結果を得ることを特徴とする畳み込み演算装置。

【請求項2】

前記複数の第1の乗算手段、前記複数の第1の累積加算手段、前記複数の第2の乗算手段、及び前記複数の第2の累積加算手段の個数は、前記2次元フィルタカーネルのサイズに対応していることを特徴とする請求項1に記載の畳み込み演算装置。 10

【請求項3】

前記複数の第1の乗算手段と前記複数の第1の累積加算手段の個数は、前記複数の第2の乗算手段と前記複数の第2の累積加算手段の個数と等しいことを特徴とする請求項1又は2に記載の畳み込み演算装置。

【請求項4】

前記2次元フィルタカーネルの水平方向のサイズと垂直方向のサイズが異なる場合、前記制御手段は、前記2次元フィルタカーネルの水平方向のサイズと垂直方向との差に応じたストールステージを、前記第一の積和演算手段の積和演算処理に挿入することを特徴とする請求項1に記載の畳み込み演算装置。 20

【請求項5】

前記第二の積和演算手段の後段に、更に、当該第二の積和演算手段と同じ構成の第三の積和演算手段を直列に接続し、

前記第三の積和演算手段の、前記第二の積和演算手段の前記第4のデータ供給手段に相当するデータ供給手段は、前記第二の積和演算手段の前記複数の第2の累積加算手段のそれぞれの出力を入力して保持し、当該各出力を複数の第4のデータとして、前記第二の積和演算手段の前記複数の第2の乗算手段の相当する前記第三の積和演算手段の乗算手段の第2の入力に供給することを特徴とする請求項1乃至4のいずれか1項に記載の畳み込み演算装置。 30

【請求項6】

前記第三の積和演算手段の後段に、更に、前記第三の積和演算手段と同じ構成の第四の積和演算手段を直列に接続し、

前記第四の積和演算手段の、前記第二の積和演算手段の前記第4のデータ供給手段に相当するデータ供給手段は、前記第三の積和演算手段の複数の累積加算手段のそれぞれの出力を入力して保持し、当該各出力を複数の第4のデータとして、前記第二の積和演算手段の前記複数の第2の乗算手段の相当する前記第四の積和演算手段の乗算手段の第2の入力に供給することを特徴とする請求項5に記載の畳み込み演算装置。

【請求項7】

前記第二の積和演算手段の後段に前記第一の積和演算手段が接続され、前記第2のデータ供給手段は、前記複数の第2の累積加算手段のそれぞれの出力を前記複数の第2のデータとして前記複数の第1の乗算手段の前記第2の入力に供給することを特徴とする請求項1乃至4のいずれか1項に記載の畳み込み演算装置。 40

【請求項8】

前記複数の第2の累積加算手段の出力に対して非線形変換を行う非線形変換手段を、更に有することを特徴とする請求項1乃至4のいずれか1項に記載の畳み込み演算装置。

【請求項9】

前記第1の積和演算手段の前記第2のデータ供給手段は、あるタイミングで前記複数の第1の乗算手段のいずれかの第1の乗算手段の前記第2の入力に供給した前記第2のデータを、他のタイミングで、前記複数の第1の乗算手段のうちの他の第1の乗算手段の前記

第2の入力に供給することを特徴とする請求項1乃至4のいずれか1項に記載の畳み込み演算装置。

【請求項10】

前記第二の積和演算手段の前記第4のデータ供給手段は、あるタイミングで前記複数の第2の乗算手段のいずれかの第2の乗算手段の前記第2の入力に供給した前記第4のデータを、他のタイミングで、前記複数の第2の乗算手段の同じ第2の乗算手段の前記第2の入力に供給することを特徴とする請求項1乃至4のいずれか1項に記載の畳み込み演算装置。

【発明の詳細な説明】

【技術分野】

10

【0001】

本発明は、畳み込み演算装置に関する。

【背景技術】

【0002】

パターン認識装置として、ニューラルネットワーク技術を応用したものが広く提案されている。特にニューラルネットワークの中でも、Convolutional Neural Networks（以下、CNN）と呼ばれる演算処理方法は、認識対象の変動に対して頑健なパターン認識を可能にする方法として知られている。このような方法を適用した例として、特許文献1には、画像データを用いた顔認識を行う技術が提案されている。

【0003】

20

ここで、CNN演算の一例を説明する。

【0004】

図23は、画像データに対するCNN演算をニューラルネットワークで実現した一例を示すブロック図である。

【0005】

図23では、画像データに対してCNN演算を行う場合を示しているため、入力層2301は、ラスタスキャンされた所定サイズの画像データである。特徴面2303a～2303cは、第1段目の階層2308の特徴面を示す。ここで特徴面とは、所定の特徴抽出フィルタ（畳み込み演算及び非線形処理）の検出結果を示すデータ面であり、例えば顔を検出する場合、目、口、又は鼻等の検出結果を示すデータ面である。このデータ面は、ラスタスキャンで得られた画像データに対する特徴抽出の検出結果であるため、検出結果も面で表される。特徴面2303a～2303cは、入力層2301に対する畳み込み演算及び非線形処理により生成される。例えば、特徴面2303aは、カーネル2311aで模式的に示す畳み込み演算及び、その演算結果の非線形変換により得られる。尚、図23のフィルタのカーネル2311b及び2311cはそれぞれ、特徴面2303b及び2303cを生成する際に使用されるカーネルである。また特徴面2305a～2305bは第2段目の階層2309の特徴面、特徴面2307は、第3段目の階層2310の特徴面を示す。

30

【0006】

図24は、畳み込みフィルタのカーネル2442の一例を示す図である。

40

【0007】

図24において、データ列2441は、ラスタスキャンで得られた画像データの参照画素を示すデータ列であり、フィルタのカーネル2442は、参照画素に対するカーネルの例を示している。この例は、カーネルのサイズが5×5のFIR（finite Impulse Response）フィルタ演算を行うことに相当する。FIRフィルタ演算は、以下の式（1）に示す積和演算により処理される。

【0008】

【数1】

$$output(x, y) = \sum_{row=0}^{rowSize-1} \sum_{column=0}^{columnSize-1} (input(x + column, y + row) \times weight(column, row))$$

10

【0009】

ここで、「input(x, y)」は座標(x, y)での参照画素値を示し、「output(x, y)」は、座標(x, y)でのFIRフィルタ演算結果を示す。また「weight(column, row)」は、座標(x + column, y + row)でのFIRフィルタ係数を示し、「columnSize」及び「rowSize」はカーネルのサイズを示し、図24の例ではいずれも「5」である。

【0010】

そして、図23の特徴面2303aを算出する場合、データ列2441は入力層2301に相当し、カーネル2442は、カーネル2311aに相当する。CNN演算では、複数のフィルタのカーネルを画素単位で走査しながら積和演算を繰り返し、最終的な積和結果を得て、その積和結果を更に非線形変換することにより特徴面を生成する。尚、特徴面2303aを算出する場合、前階層との結合数が「1」であるため、カーネルは1つである。

20

【0011】

次に、第2段目の階層2309の特徴面2305aを生成する演算を説明する。

【0012】

特徴面2305aは、前段の第1段目の階層2308の3つの特徴面2303a～2303cと結合している。従って、特徴面2305aのデータを算出する場合、特徴面2303aに対してはカーネル2312aで模式的に示すカーネルを用いたフィルタ演算を行い、この結果を累積加算器に保持する。同様に、特徴面2303b及び2303cに対して、夫々カーネル2313a及び2314aのフィルタ演算を行い、これらの結果を累積加算器に蓄積する。これらの3種類のフィルタ演算の終了後、ロジスティック関数又は双曲正接関数(tanh関数)を利用した非線形変換処理を行う。以上の処理を画像全体に対して1画素ずつ走査しながら処理することにより、特徴面2305aを生成する。

30

【0013】

同様に、特徴面2305bを生成する際は、前段の階層2308の特徴面2303a～2303cに対するカーネル2312b, 2313b及び2314bによる3つの畳込みフィルタ演算を行う。また第3段目の階層2310の特徴面2307を生成する際は、前段の階層2309の特徴面2305a～2305bに対するカーネル2315及び2316による2つの畳込みフィルタ演算を行う。尚、各フィルタ係数は、バックプロパゲーション学習又は深層学習等の一般的な方法を用いて、予め学習により決定されている。物体の検出又は認識等においては、10×10以上の大きなサイズのカーネルを使用することが多い。

40

【0014】

またCNN演算処理では、多数の大きなカーネルサイズのフィルタが階層的に使用され、膨大な回数の畳込み演算が必要とされる。この課題に対する対処方法としては、例えば非特許文献1では、フィルタ係数を1次元の基底フィルタ係数に分解し、畳込み演算における積和演算回数の削減を図ったものが提案されている。

【0015】

50

一方、CNNにおける畳込み演算をプロセッサ上で動作するソフトウェアとして実現した場合、前述したように畳込み演算の回数が膨大であるため、所望の演算速度を満たせないケースが有り得る。この課題に対する対処方法としては、例えば特許文献2では、CNN演算を、ディジタルハードウェアで実現する技術が提案されている。

【先行技術文献】

【特許文献】

【0016】

【特許文献1】特開平10-021406号公報

【特許文献2】米国特許公開2012/0303932号公報

【特許文献3】特許第5376920号公報

10

【非特許文献】

【0017】

【非特許文献1】Exploiting Linear Structure Within Convolutional Networks for Efficient Evaluation, CoRR2014, Denton, et al.

【発明の概要】

【発明が解決しようとする課題】

【0018】

以上説明したように、CNN演算のように膨大な回数の畳込み演算が必要な場合に、その演算量を削減する方法としては、フィルタ係数を低次元フィルタ係数に分解して畳込み演算を実行するのが有効である。また、膨大な回数の畳込み演算を実行する際には、プロセッサ上で動作するソフトウェアよりも、ディジタルハードウェアとして構成することが、高速化及び低消費電力の点で有効である。しかしながらこれまで、低次元フィルタ係数に分解して畳込み演算を実行する際に、効率的に演算を実行することが可能なディジタルハードウェアの構成は提案されていない。

20

【0019】

本発明の目的は、上記従来技術の課題を解決し、効率的な畳み込み演算処理を実現できる畳み込み演算装置を提供することにある。

【課題を解決するための手段】

【0020】

上記目的を達成するために本発明の一態様に係る畳み込み演算装置は以下のような構成を備える。即ち、

30

、 それぞれ、第1及び第2の入力に入力されたデータを乗算する複数の第1の乗算手段と、

前記複数の第1の乗算手段に対応して設けられ、それぞれ対応する前記第1の乗算手段の乗算結果を累積する複数の第1の累積加算手段と、

前記複数の第1の乗算手段のそれぞれの前記第1の入力に第1のデータとして2次元フィルタカーネルの係数データを分解した水平方向の係数データを供給する第1のデータ供給手段と、

前記複数の第1の乗算手段の前記第2の入力に複数の第2のデータとして入力画像データの水平方向のデータを供給する第2のデータ供給手段と、を有する第一の積和演算手段と、

40

、 それぞれ、第1及び第2の入力に入力されたデータを乗算する複数の第2の乗算手段と、

前記複数の第2の乗算手段に対応して設けられ、それぞれ対応する前記第2の乗算手段の乗算結果を累積する複数の第2の累積加算手段と、

前記複数の第2の乗算手段のそれぞれの前記第1の入力に第3のデータとして2次元フィルタカーネルの係数データを分解した垂直方向の係数データを供給する第3のデータ供給手段と、

前記複数の第1の累積加算手段のそれぞれの出力から前記2次元フィルタカーネルの垂直方向の列単位で必要な参照データをリングバッファにロードして一括して保持し、当該

50

ロードした参照データを動作クロックに応じてリング状にシフトして複数の第4のデータとして前記複数の第2の乗算手段の前記第2の入力に供給する第4のデータ供給手段と、を有する第二の積和演算手段と、

前記第一及び第二の積和演算手段による積和演算処理を並行して実行するように制御する制御手段と、を有し、

前記複数の第2の累積加算手段の出力として、前記入力画像データに対する前記2次元フィルタカーネルによる畳み込み演算の結果を得ることを特徴とする。

【発明の効果】

【0021】

本発明によれば、効率的な畳み込み演算処理を実現でき、演算処理速度の高速化及び低消費電力化を実現できる。 10

【0022】

本発明のその他の特徴及び利点は、添付図面を参照とした以下の説明により明らかになるであろう。尚、添付図面においては、同じ若しくは同様の構成には、同じ参照番号を付す。

【図面の簡単な説明】

【0023】

添付図面は明細書に含まれ、その一部を構成し、本発明の実施形態を示し、その記述と共に本発明の原理を説明するために用いられる。

【図1】本発明の実施形態1に係る階層的畳み込み演算回路を具備した物体検出装置の構成を示すブロック図。 20

【図2】第1の実施形態に係るCNN処理部の構成を示すブロック図である。

【図3】実施形態1に係るシフトレジスタの構成例を示す図。

【図4】実施形態1に係るリングバッファの構成例を説明する図。

【図5】実施形態1に係る第一の積和演算回路の累積加算器及び第二の積和演算回路の累積加算器の構成例を示すブロック図。 30

【図6】実施形態1に係る非線形変換処理部の構成を示すブロック図。

【図7】実施形態1に係るCNN処理部の制御部の構成を示すブロック図。

【図8】実施形態1に係る制御部のレジスタ群に設定される情報の一例を示す図。

【図9】実施形態1に係るRAMに格納される参照データ、重み係数データ及び演算結果のメモリマップの一例を示す図。 30

【図10】実施形態1に係る物体検出装置の動作を説明するフローチャート(A)とCNN処理部からの終了通知割り込みによりCPUが実行する処理を説明するフローチャート(B)。

【図11】実施形態1に係る畳み込み演算処理の一例を説明する図。

【図12】実施形態1に係る畳み込み演算処理の一例を説明する図。

【図13】実施形態1に係る畳み込み演算処理の一例を説明する図。

【図14】一般的な畳み込み演算の具体例を示す図。

【図15】実施形態1に係るCNN処理部における畳み込み演算の処理時の動作を説明するタイムチャート。 40

【図16】実施形態1及び2に係る第一の積和演算回路による積和演算処理と、第二の積和演算回路による積和演算処理のパイプライン動作のタイミングを説明する模式図。

【図17】本発明の実施形態3に係るCNN処理部の構成を説明するブロック図。

【図18】実施形態3における畳み込み演算処理時の動作を説明するタイムチャート。

【図19】実施形態4に係るCNN処理部の構成を説明するブロック図。

【図20】実施形態4に係る畳み込み演算の処理時の動作を説明するタイムチャート。

【図21】実施形態4に係る第一の積和演算回路の積和演算処理と、第二の積和演算回路の積和演算処理と、第三の積和演算回路の積和演算処理のパイプライン動作の模式図。

【図22】実施形態5に係るCNN処理部の構成を説明するブロック図。

【図23】画像データに対するCNN演算をニューラルネットワークで実現した一例を示す。 50

すブロック図。

【図24】畳込みフィルタのカーネルの一例を示す図。

【発明を実施するための形態】

【0024】

以下、添付図面を参照して本発明の実施形態を詳しく説明する。尚、以下の実施形態は特許請求の範囲に係る本発明を限定するものでなく、また本実施形態で説明されている特徴の組み合わせの全てが本発明の解決手段に必須のものとは限らない。

【0025】

【実施形態1】

先ず、本発明の実施形態1を説明する。

10

【0026】

図1は、本発明の実施形態1に係る階層的畳込み演算回路を具備した物体検出装置の構成を示すブロック図である。この物体検出装置（物体認識装置）は、2次元の畳込み演算を行い、画像データから特定の物体を検出する機能を有する。

【0027】

この物体検出装置（物体認識装置）は、画像入力部120、CNN処理部122、ブリッジ124、前処理部125、DMAC(Direct Memory Access Controller)126及びRAM100を有している。更に、CPU(Central Processing Unit)127、ROM128及びRAM129も設けられている。そして画像入力部120、CNN処理部122、前処理部125及びDMAC126が画像バス123を介して互いに接続され、CPU127、ROM128及びRAM129がCPUバス130を介して互いに接続されている。またブリッジ124により画像バス123とCPUバス130との間のデータ転送が可能となっている。

20

【0028】

画像入力部120は、光学系、CCD(Charge-Coupled Devices)又はCMOS(Complementary Metal Oxide Semiconductor)センサ等の光電変換デバイスを有する。更に、そのセンサを制御するドライバ回路、ADコンバータ、各種画像補正を司る信号処理回路及びフレームバッファ等も設けられている。CNN処理部122は、階層的畳込み演算回路として機能する。RAM100は、CNN処理部122の演算作業バッファとして各種データを一時的に保持するのに使用される。尚、CNN処理部122の構成等の詳細は、図2を参照して詳しく後述する。前処理部125は、CNN演算による検出処理を効果的に行うための種々の前処理を行う。例えば、色変換処理及びコントラスト補正処理等の画像データ変換処理を、ハードウェアにより処理する。DMAC126は、画像バス123上の画像入力部120、CNN処理部122及び前処理部125とCPUバス130との間のデータ転送を司る。

30

【0029】

ROM(Read Only Memory)128は、CPU127の動作を規定する命令(プログラム)及びパラメータ等を格納しており、CPU127は、これらを読み出してそれら命令を実行することにより、この物体検出装置の全体の動作を制御する。その際、RAM129がCPU127の作業領域として使用される。尚、CPU127はブリッジ124を介して画像バス123上のRAM100にアクセスすることも可能である。

40

【0030】

次に、CNN処理部122を詳しく説明する。

【0031】

図2は、実施形態1に係るCNN処理部122の構成例を説明するブロック図である。上述したように、CNN処理部122は、階層的畳込み演算回路として機能する。

【0032】

CNN処理部122は、第一の積和演算処理部200と第二の積和演算処理部220、それぞれの積和演算回路の演算処理を制御する制御部201、記憶部203、シフトレジスタ206、非線形変換処理部209を有している。ここで、第二の積和演算回路220

50

は、第一の積和演算回路 200 の後段に直列に接続されている。

【0033】

第一の積和演算回路 200 は、記憶部 202、シフトレジスタ 204, 205、それらが 2 つの入力を備えた複数の乗算器 207、及び複数の累積加算器 208 を有している。また第二の積和演算回路 220 は、記憶部 210、シフトレジスタ 211、(複数の) リングバッファ 212、それらが 2 つの入力を備えた複数の乗算器 213、複数の累積加算器 214 を含んでいる。

【0034】

制御部 201 には、CNN 处理部 122 の基本的な動作を決定するレジスタ群、そのレジスタ群に設定された値に基づいて種々の信号のタイミングを制御するシーケンス制御部、及び RAM 100 へのアクセス調停を行うメモリ制御部等が設けられている。尚、制御部 201 の構成等の詳細は図 7 を参照して後述する。

10

【0035】

第一及び第二の積和演算回路 200, 220 の記憶部 202 及び 210 は、RAM 100 から出力された重み係数データを一時的に保持する。いま重み係数が 8 ビットで表されるデータの場合、記憶部 202 及び 210 は 8 ビット幅の複数のレジスタで構成される。また、記憶部 202 及び 210 は、フィルタのカーネルサイズと同じサイズ以上のレジスタ(記憶容量)を有する。例えば、カーネルサイズが「12」の場合、そのレジスタの数は少なくとも「12」あればよい。つまり、想定される最大フィルタサイズのレジスタ数で構成することが好ましい。尚、実施形態 1 では、前述したフィルタのカーネルは、2 次元構造を有するカーネルの係数データを、水平方向及び垂直方向の 1 次元構造を有する 2 種類の係数データに分解したものを指す。2 次元構造を有するカーネルと 1 次元構造を有するカーネルの詳細に関しては、別途後述する。

20

【0036】

次に記憶部 203 は、RAM 100 に格納された参照データ(画像データ)を一時的に保持する。参照データが 8 ビットで表されるデータの場合、記憶部 203 は 8 ビット幅の複数のレジスタで構成される。記憶部 203 は「並列に処理するデータの数(レジスタ長)」+「カーネルサイズ - 1」以上の個数のレジスタ(記憶容量)を有する。ここでレジスタの個数は、一度に処理する(並列演算する)位置の特徴面データが参照するデータを得るために必要な値であり、この値以上の個数のレジスタが設けられていればよい。例えば、カーネルサイズが「12」、演算並列度が「12」の場合、少なくとも 23 (= 12 + 11) 個の 8 ビットレジスタが設けられていればよい。

30

【0037】

ここで第一及び第二の積和演算回路 200, 220 のシフトレジスタ 204, 205 及び 211 と、シフトレジスタ 206 は、データロード機能を備えている。例えば、シフトレジスタ 204, 205 及び 211 は、夫々記憶部 202, 203 及び 210 と同じビット幅の複数のレジスタを有し、リングバッファ 212 とともに各乗算器へのデータ供給を行っている。またシフトレジスタ 206 は、それらが、累積加算器 214 の出力の有効ビット数と同じビット数以上のレジスタを、複数有している。

40

【0038】

図 3 は、実施形態 1 に係るシフトレジスタ 204, 205, 211 及び 206 の構成例を示す図である。

【0039】

この例では、シフトレジスタは 4 個のレジスタ(図 3 ではフリップフロップ 301 ~ 304)を有している。シフトレジスタには、4 個の多ビットフリップフロップ 301 ~ 304 が設けられており、これらが CLOCK 信号に同期して所定ビット数のデータをラッチするレジスタとして機能している。フリップフロップ 301 ~ 304 にはイネーブル信号(Enable 信号)が供給されており、フリップフロップ 301 ~ 304 は、Enable 信号が 1(ハイレベル)のときに CLOCK 信号の立ち上がりでデータをラッチする。一方、Enable 信号が 0(ローレベル)の場合は、前クロックでラッチしたデータを出力する。

50

タをそのまま保持する。つまり状態の遷移が生じない。また3個のセレクタ305～307が設けられており、これらは選択信号(Lo ad信号)が0(ロウレベル)のときに、前段のフリップフロップの出力信号OUTx(x:0～2)を選択し、1(ハイレベル)のときに入力信号INx(x:1～3)を選択する。即ち、セレクタ305～307は、Lo ad信号のレベルに応じて、フリップフロップに供給するデータを前段のフリップフロップの出力とするか、入力信号INxとするかを選択している。こうして、このシフトレジスタは、選択信号(Lo ad信号)が0で、かつEnable信号が1のときにCLK信号の立ち上がりで、保持しているデータのシフト動作を行う。

【0040】

ここで図2のLo ad2信号、Lo ad4信号、Lo ad7信号及びLo ad5信号は10、図3のLo ad信号に相当し、図2のEnable1信号、Enable2信号、Enable4信号及びEnable3信号が、図3のEnable信号に相当する。このような構成により、並列度が多い場合でも、参照データや重み係数データを選択するセレクタの構成を簡素化でき、配線数の増大及び遅延の増大を招くことなく、高速な回路を構築できる。

【0041】

次に図2のリングバッファ212について説明する。リングバッファ212は、データロード機能を備えており、それぞれが、第一の積和演算回路200の累積加算器208の出力の有効ビット数と同じビット数以上のデータを保持できるバッファ401を複数有している。

20

【0042】

図4は、実施形態1に係るリングバッファ212の構成例を説明する図である。この例では、リング状にデータをシフトする機能を有する単一リングバッファ401が、保持するデータの個数に相当する4本分設けられている。ここでは、4本の単一リングバッファは、制御信号により同期して同一の動作を実行する。従って、以下では単一リングバッファ401の動作を詳しく説明する。

【0043】

単一リングバッファ401には、4個の多ビットフリップフロップ402a～402dが含まれており、これらがCLK信号に同期して所定ビット数のデータをラッチする。フリップフロップ402a～402dにはイネーブル信号(Enable信号)が与えられ、フリップフロップ402a～402dは、Enable信号が1(ハイレベル)のときにCLK信号の立ち上がりでデータをラッチする。一方、Enable信号が0(ロウレベル)のときは、前クロックでラッチしたデータをそのまま保持する。つまり、状態遷移が生じない。また、セレクタ403が設けられている。

30

【0044】

セレクタ403は、選択信号(Lo ad信号)が0(ロウレベル)のときに、フリップフロップ402bの出力信号を選択し、1(ハイレベル)の場合に、入力信号IN0を選択する。即ち、セレクタ403は、Lo ad信号のレベルに応じて、リングバッファ401にシフト動作、或いはロード動作を実行させる。図2のLo ad8信号は図4のLo ad信号に相当し、図2のEnable5信号が図4のEnable信号に相当している。ここで図4より明らかのように、フリップフロップ402aの出力は、フリップフロップ402dの入力に接続されている。従って、Enable信号が1でLo ad信号が0のとき、単一リングバッファ401に保持されたデータが、CLK信号に同期してリング状にシフト動作する。尚、図4では、単一リングバッファ401が4個設けられ、これらリングバッファが制御信号により同期して同一の動作を行うため、結果的に4個のデータが同期してリング状にシフト動作することがわかる。尚、単一リングバッファ401で保持されたデータを外部に出力するのは、フリップフロップ402aのみである。従って、本実施形態1に係るリングバッファ212は、一度に4個のデータを出力することとなる。

40

【0045】

50

続いて制御部 201 は、第一及び第二の積和演算回路 200, 220 のシフトレジスタ 204 及び 211 のシフト動作中に、次の積和演算処理に必要な重み係数データを RAM 100 から記憶部 202 にロードする。また制御部 201 は、シフトレジスタ 205 のシフト動作中に、次の積和演算処理に必要な参照データを、RAM 100 から記憶部 203 にロードする。そしてシフトレジスタ 204 は、初期データのロード（記憶部 202 からの一括ロード）後に、それぞれのカーネルのサイズと同じクロック数だけシフト動作を実行し、乗算器 207 に対して重み係数データを連続して供給する。つまり、図 3 の信号 OUTn（シフトレジスタ 204 の最終段出力）が全ての乗算器 207 の第 1 の入力に共通して供給される。

【0046】

10

またシフトレジスタ 205 は、記憶部 203 から参照データがロードされると、列方向のカーネルサイズと同じクロック数だけシフト動作を実行し、乗算器 207 に対して複数の参照データを同時に供給する。つまり、図 3 の信号 OUT1 ~ OUTn のそれぞれが、各対応する乗算器 207 の第 2 の入力に並列に供給される。ここでシフトレジスタ 204 及びシフトレジスタ 205 は互いに同期して動作し、これらからのデータが、乗算器 207 の第 1 の入力及び第 2 の入力に同期して供給される。以上の処理により、積和演算処理及び RAM 100 からのデータロードを、フィルタのカーネルの列単位でパイプライン処理することが可能になる。

【0047】

20

また同時に、前述したシフトレジスタ 204 と同様に、シフトレジスタ 211 は、初期データのロード（記憶部 210 からの一括ロード）後に、それぞれのカーネルサイズと同じクロック数だけシフト動作を実行する。これにより、乗算器 213 に対して重み係数データを連続して供給する。つまり、図 3 の信号 OUTn（シフトレジスタ 211 の最終段出力）が全ての乗算器 213 に共通して供給される。

【0048】

30

またリングバッファ 212 は、第一の積和演算回路 200 の累積加算器 208 からデータが入力されると、制御部 201 からの制御信号に基づき、リングバッファ内に保持された中で最も古いデータを、その入力データで上書きする。そして、その入力データを乗算器 213 に対して同時に供給する。そして、次に第一の積和演算回路 200 の累積加算器 208 からデータが入力されるまでは、垂直方向のカーネルサイズと同じクロック数から 1 を引いた回数だけリング状のシフト動作を実行する。こうして乗算器 213 に対して複数の参照データを同時に供給する。つまり、図 4 の信号 OUT0 ~ OUTn のそれぞれが、各対応する乗算器 213 の第 2 入力に並列に供給される。ここでリングバッファ 212 及びシフトレジスタ 211 は互いに同期して動作しており、これらからのデータが乗算器 213 の第 1 の入力及び第 2 の入力に供給される。

【0049】

40

またリングバッファ 212 が、垂直方向のカーネルサイズと同じクロック数から 1 を引いた回数だけリング状のシフト動作を実行すると、前述と同様に、第一の積和演算回路 200 の累積加算器 208 からデータがロードされる。即ち、リングバッファ 212 では、リング状のシフト動作を垂直方向のカーネルサイズと同じクロック数から 1 を引いた回数繰り返す。そしてその後に、リングバッファ 212 に保持されたデータの中で最も古いデータが、第一の積和演算回路 200 の累積加算器 208 から入力されるデータによって上書きされることになる。

【0050】

以上の処理により、第二の積和演算回路 220 での積和演算処理及び RAM 100 からの重み係数データのロードを、垂直方向のフィルタのカーネル単位で並行して処理することが可能になる。また前述した第一の積和演算回路 200 の演算処理と、第二の積和演算回路 220 の演算処理とはクロック信号に同期して実行されるため、パイプライン動作を行うことができる。

【0051】

50

図5は、実施形態1に係る第一の積和演算回路200の累積加算器208及び第二の積和演算回路220の累積加算器214の構成を示すブロック図である。尚、累積加算器208及び214は同様の構成を有する。

【0052】

累積加算器208及び214は、加算器501及びレジスタ502を有しており、累積加算器208及び214は、Latch Enable信号に従って入力データの累積和をレジスタ502に保持する。Latch Enable信号は、クロック信号に同期した信号である。第一の積和演算回路200の累積加算器208により得られた累積和は、対象とする特徴面に対応するカーネルの演算終了後、リングバッファ212にロードされ、所定のタイミングで乗算器213に送られる。また第二の積和演算回路220の累積加算器214により得られた累積和は、対象とする特徴面に対応するカーネルの演算終了後、シフトレジスタ206にロードされ、所定のタイミングで非線形変換処理部209に送られる。乗算器207及び213、累積加算器208及び214としては、例えば、夫々同一クロックで動作する同一のものが12個並んで設けられている。そして、シフトレジスタ206は、例えば、12個の累積加算器214の出力を保持することが可能なフリップフロップを含んでいる。また累積加算器208及び214の出力は、所定の有効ビットのみ、後段に接続するリングバッファ212又はシフトレジスタ206に出力される。

【0053】

図6は、実施形態1に係る非線形変換処理部209の構成を示すブロック図である。

【0054】

非線形変換処理部209は、ルックアップテーブル(LUT)601及びセレクタ602を有している。LUT601は、積和演算結果をアドレスデータ(In)としてROM等に保持されたデータを参照する。このROMには、例えば予めアドレス値に対応する出力の非線形変換した値が記憶されている。セレクタ602は、非線形変換した値(LUT601の出力値)、或いは非線形変換しない場合に積和演算結果(In)をそのまま出力する(Out)。セレクタ602への選択信号(Select)は、制御部201から供給されている。つまりセレクタ602は、制御部201の「非線形変換」レジスタ(後述)の値に従って制御される。尚、後述のように、「非線形変換」レジスタを含むレジスタセット801～803(図8参照)は、特徴面単位で構成される。このため、非線形変換処理の有無も、特徴面単位で選択可能である。このように、非線形変換の有無を特徴面単位で選択可能にすることで、非線形変換処理を含む階層と、含まない階層とが混在した大規模ネットワークを構築することができる。また非線形変換処理部209により得られたデータは、RAM100の所定のアドレスに格納される。このRAM100の格納アドレスも、制御部201のレジスタ群702の設定及びシーケンス制御部701(図7参照)の動作に従って制御される。

【0055】

次に制御部201の詳細について説明する。

【0056】

図7は、実施形態1に係るCNN処理部122の制御部201の構成を示すブロック図である。

【0057】

制御部201は、シーケンス制御部701、レジスタ群702(記憶領域)及びメモリ制御部705を有している。シーケンス制御部701は、レジスタ群702に設定された情報に従って、CNN処理部122の動作を制御する種々の制御信号704を入出力する。同様に、シーケンス制御部701は、メモリ制御部705を制御する制御信号706を生成する。シーケンス制御部701は、例えば、バイナリカウンタ又はジョンソンカウンタ等からなるシーケンサを有している。レジスタ群702は、複数のレジスタセットを含み、1つの階層的な処理を行うための情報が、レジスタセット毎に保持される。レジスタ群702は、外部からのアクセスが可能に構成されている。

【0058】

10

20

30

40

50

図8は、実施形態1に係る制御部201のレジスタ群702に設定される情報の一例を示す図である。

【0059】

この例では、3つのレジスタセット801, 802及び803がレジスタ群702に含まれており、これらのうちの1つが1つの特徴面を処理するために必要な情報を保持する。レジスタ群702には、ブリッジ124及び画像バス123を介してCPU127から予め所定の値が書き込まれる。ここでは、レジスタセット801～803の各レジスタが32ビット長であるとする。

【0060】

図8において、「最終層指定」レジスタは、当該レジスタセットに対応する特徴面が最終層か否かを指定するレジスタである。このレジスタの値が「1」の場合は、処理対象の特徴面が最終層であることを示し、その特徴面の処理を終了すると検出処理を終了する。「参照データ面の数」レジスタは、対象特徴面と接続する前階層の特徴面（データ領域）数を指定するレジスタであり、例えば、対象特徴面と接続する前階層の特徴面数が「3」の場合は「3」が設定される。「非線形変換」レジスタは、非線形変換の有無を指定するレジスタであり、当該レジスタに「1」が設定されている場合は非線形変換処理を行い、当該レジスタに「0」が設定されている場合は非線形変換処理を行わない。「演算結果格納先ポインタ」レジスタは、対象とする特徴面の演算結果を保持するためのRAM100の領域の先頭アドレスを指定するレジスタであり、このアドレス値を先頭ポインタとして演算結果をラスタスキャン順に格納する。「水平方向フィルタカーネルのサイズ」レジスタ及び「垂直方向フィルタカーネルのサイズ」レジスタは、対象の特徴面の演算に使用するフィルタのカーネルサイズを指定するレジスタである。「重み係数格納先ポインタ」レジスタは、RAM100に記憶されている、対象の特徴面の演算に使用する重み係数の格納先アドレスを示す。例えば、重み係数データは「参照データ面の数」レジスタと同じ数の係数の組を有し、「重み係数格納先ポインタ」レジスタで指定されるアドレスから順に格納されている。即ち、（「水平方向フィルタカーネルのサイズ」+「垂直方向フィルタカーネルのサイズ」）×「参照データ面の数」の個数の係数データがRAM100に格納されている。「参照データの水平サイズ」レジスタ及び「参照データの垂直サイズ」レジスタは、夫々参照データの水平方向の画素数及び垂直方向のライン数を示すレジスタである。また参照データは、「参照データ格納先ポインタ」レジスタの示すアドレスを先頭としてRAM100にラスタスキャン順に格納されている。即ち（「参照データの水平サイズ」×「参照データの垂直サイズ」）×「参照データ面の数」の個数の参照データがRAM100に格納されている。

【0061】

このような複数のレジスタが各特徴面単位に設けられている。演算対象とする特徴面の「参照データの格納先ポインタ」レジスタの内容が前階層の結合対象特徴面の「演算結果格納先ポインタ」である場合、前階層の特徴面と対象となる特徴面が結合されることになる。従って、ここでのレジスタ設定（ポインタ設定）だけで、任意の階層的結合関係を特徴面単位に構築することが可能になる。

【0062】

図9は、実施形態1に係るRAM100に格納される参照データ、重み係数データ及び演算結果のメモリマップの一例を示す図である。

【0063】

ここでは、前階層の結合数が「3」であるとする。このような場合、領域901～903は3つの参照データを格納する領域であり、領域904～906は参照データに対応するフィルタカーネル係数を格納する領域である。また領域907は、特徴面の演算結果を格納する領域を示す。例えば、各領域には、ラスタスキャンされたデータが格納される。従って、制御部201は、レジスタ群702に格納された先頭ポインタ情報及びサイズに関する情報に基づいて、必要な参照データ及びカーネルの係数データにアクセスすることが可能である。また、次の階層の特徴面の演算時には、演算結果が格納された領域907

10

20

30

40

50

を参照データの格納領域として指定する（レジスタ群702への設定）だけで、不要なデータの転送を行わずに、高速に実行することが可能である。

【0064】

シーケンス制御部701は、図8の「水平方向フィルタカーネルのサイズ」、「垂直方向フィルタカーネルのサイズ」、「参照データの水平サイズ」及び「参照データの垂直サイズ」等の内容に従って演算動作タイミングに関わるシーケンス制御を行う。

【0065】

メモリ制御部705は、シーケンス制御部701が生成する制御信号706に従って、参照データ707、重み係数データ708及び演算結果データ709のRAM100からの読み出し、及びRAM100への書き込みを調停する。具体的には、画像バス123を介したRAM100へのアクセス、参照データ707の読み出し、重み係数データ708の読み出し、演算結果データ709の書き出しを制御する。尚、RAM100のデータ幅及び各バス（データ707～709）のデータ幅は、例えば全て32ビットである。

【0066】

次に、このように構成された物体検出装置の動作について説明する。

【0067】

図10（A）は、実施形態1に係る物体検出装置の動作を説明するフローチャートである。

【0068】

先ずS1001でCPU127は、検出処理の開始に先立って各種初期化処理を実行する。ここで例えばCPU127は、CNN処理部122の動作に必要な重み係数をROM128からRAM100に転送すると共に、CNN処理部122の動作、即ちCNNネットワークの構成を定義するための各種レジスタ設定を行う。具体的には、CNN処理部122の制御部201に存在する複数のレジスタ（図8参照）に所定の値を設定する。同様に、前処理部125等のレジスタにも動作に必要な値を書き込む。次にS1002に進みCPU127は、画像入力部120により、画像センサの出力する信号を入力させてデジタルデータに変換させ、フレーム単位でフレームバッファ（図示せず）に格納する。次にS1003に進みCPU127は、前処理部125により、所定の信号に基づいて画像変換処理を実行させる。ここで前処理部125は、画像入力部120のフレームバッファの画像データから輝度データを抽出し、コントラスト補正処理を行う。輝度データを抽出する場合は、例えば、一般的な線形変換処理によりRGB画像データから輝度データを生成する。またコントラスト補正では、例えば、一般的に知られているコントラスト補正処理を適用してコントラストを強調する。こうして前処理部125によりコントラスト補正処理された輝度データは、検出用画像としてRAM100に格納される。こうして、1フレームの画像データに対して前処理が完了すると、前処理部125からの完了信号（図示せず）に基づいて、次にCNN処理部122により、物体の検出処理を実行させる（S1004～S1005）。S1004～S1005の処理は、CNN処理部122の制御部201により実行される。

【0069】

この物体の検出処理では、S1004、S1005でCNN処理部122は、特徴面単位（例えば図23に示す2303a～2303c等）で畳込み演算を行う。そして、全ての特徴面に対する処理を終了するとS1006に進み（図23の場合、特徴面2307の算出を終了した場合）、CPU127に対して割り込み信号を生成する。つまり、制御部201が、S1004で1つの特徴面の畳込み演算処理を終了するとS1005に進み、レジスタセット801の「最終層指定」レジスタの内容に基づいて最終層かどうか判定する。ここで最終層でないと判定したときはS1004に進んで、次の特徴面の処理を実行する。このとき制御部201は、次の特徴面を処理する場合、レジスタ群702の次のアドレスに存在するレジスタセットの内容に従って同様の畳込み演算処理を行う。そしてレジスタセットが最終層を示している場合、制御部201は、S1006で所定の畳込み演算処理の終了後、CPU127に対して終了通知割り込みを発生する。そしてS1007

10

20

30

40

50

に進み、CPU127は、検出処理に必要な全ての画像に対する処理が終了したかどうかを判定し、終了していないときはS1002に進む。ここでは例えば、動画像から所定の物体を検出するような場合、CPU127は、上述した処理を、対象するとする全ての画像に対してフレーム画像単位で連続して実行し、それらの処理が終了すると、この処理を終了する。

【0070】

図10(B)は、CNN処理部122からの終了通知割り込みによりCPU127が実行する処理を説明するフローチャートである。尚、この処理をCPU127に実行させるプログラムはROM128に格納されており、CPU127がそのプログラムを読み出して実行することにより、このフローチャートで示す処理が達成される。

10

【0071】

ここで図11～図13に示すように、実施形態1に係る畳込み演算処理は、水平方向の1次元フィルタカーネルと、垂直方向の1次元フィルタカーネルによるものとの2段階で実行される。ここで、水平方向及び垂直方向それぞれの1次元フィルタカーネルは、前述したように2次元構造を有するフィルタカーネルを水平方向及び垂直方向の2つの1次元構造を有するフィルタカーネルに分解したものを指す。

【0072】

そこで、2次元構造を有するフィルタカーネルと1次元構造を有するフィルタカーネルの詳細を説明する。まず一般的な畳込み演算処理に関して説明する。

【0073】

図14は、一般的な畳込み演算の具体例を示す図である。ここではラスタスキャンで得られたデータを2次元座標で示している。

20

【0074】

1401は参照データ面を表し、各ブロック（模式的に示す最小一升）がラスタスキャン順でRAM100に格納された前階層の演算結果(`input(x, y)`、`x`：水平方向位置、`y`：垂直方向位置)を示す。1405は算出対象となる特徴面データを表し、各ブロックがラスタスキャンされた演算結果(`output(x, y)`、`x`：水平方向位置、`y`：垂直方向位置)を示す。また1402、1403は夫々、`output(4, 4)`、`output(4, 5)`の特徴面データ1410、1411を算出する場合のフィルタ演算に必要な参照データ範囲を示す。ここで、フィルタのカーネルサイズは水平方向に「5」、垂直方向に「5」である。例えば、特許文献3には、この一般的なフィルタ演算を実行する畳込み演算処理装置が記載されている。その場合、例えば1404は、同時にフィルタ演算する特徴面データの領域を示し、例えば並列度が「5」の場合、`output(4, y)` : $y = 4 \sim 8$ の特徴面位置のデータを同時に演算する。従って、並列処理の対象である領域1404を水平方向に1画素単位、垂直方向に5ライン単位で走査させながらフィルタ演算を行い、二次元データ列に対してフィルタ演算を行う。このフィルタ演算処理は特許文献3に詳しく説明されているため、これ以上の詳細な説明は割愛する。

30

【0075】

続いて実施形態1に係る2次元構造を有するフィルタカーネルを水平方向及び垂直方向の2つの1次元構造を有するフィルタカーネルに分解して畳込み演算を実行する方法に関して説明する。

40

【0076】

2次元構造を有するフィルタカーネルの重み係数マトリクスは、1次元構造を有する水平方向及び垂直方向のフィルタカーネルを用いた式(2)に示す乗算によって近似可能であることが知られている。

【0077】

【数2】

$$output(x, y) = \sum_{row=0}^{rowSize-1} \sum_{column=0}^{columnSize-1} (input(x + column, y + row) \times weight_{horizontal}(column) \times weight_{vertical}(row))$$

【0078】

10

尚、実際には、複数の水平方向及び垂直方向のフィルタカーネルの組み合わせを用い、式(3)に示すようにこれらのフィルタ演算結果の総和を取ることで近似精度を高める手法が取られることが多い。式(3)では、フィルタカーネルの組み合わせを添え字「pair」で示している。

【0079】

【数3】

$$output(x, y) = \sum_{pair=0}^{pairSize-1} \sum_{row=0}^{rowSize-1} \sum_{column=0}^{columnSize-1} (input(x + column, y + row) \times weight_{pair, horizontal}(column) \times weight_{pair, vertical}(row))$$

20

【0080】

これらの手法は、非特許文献1に詳しく記載されているため、これ以上の説明は省略する。

【0081】

次に、前記式(2)及び(3)を実行するための畳込み演算方法に関して説明する。

【0082】

30

まず第一の積和演算回路200において、図11(A)(B)及び図12(A)に示すように、1次元構造を有する水平方向のフィルタカーネルを用いた畳込み演算を実行する。

【0083】

図11(A)は、1つの特徴面データにおける畳込み演算処理の具体例を説明する図である。図11(A)では、ラスタスキャンで得られたデータを2次元座標で示している。

【0084】

1101は参照データ面を表し、各ブロック(模式的に示す最小一升)がラスタスキャン順でRAM100に格納された前階層の演算結果(input(x, y)、x:水平方向位置、y:垂直方向位置)を示す。1102は、対象となる特徴面データを表し、各ブロックがラスタスキャンされた演算結果(output(x, y)、x:水平方向位置、y:垂直方向位置)を示す。また1103は、output(4, 4)位置の特徴面データ1110を算出する場合のフィルタ演算で畳込み演算が実行される参照データ範囲を示している。

40

【0085】

図11(A)では、水平方向のフィルタカーネルのサイズが「5」の場合を示しており、参照データ範囲1103は、水平方向のフィルタカーネルのサイズと等しいinput(x, 2):x=2~6となる。実際には、図11(A)で点線で示しているように、畳込み演算を垂直方向に5回繰り返すことにより、所定の範囲の参照データ面に対する演算を実行する。

50

【0086】

またさらに図11(B)は、並列にフィルタ演算する特徴面データ1104の領域を示している。例えば、特徴面データ1104は、同時にフィルタ演算する特徴面データの領域を示し、並列度が5の場合、`output(x, 4) : x = 4 ~ 8`の特徴面データを同時に演算する。この時、畳込み演算が実行される参照データの範囲は、図11(B)で太線で囲まれた参照面データ1105となる。尚、図11(B)では、各特徴面データと参照データ範囲の対応を示す点線は、それぞれ特徴面データ1104である`output(x, 4) : x = 4 ~ 5`についてのみ示している。

【0087】

ここで、元々のフィルタのカーネルサイズは水平方向に「5」、垂直方向に「5」である。特徴面データ1104を得るために、図12(A)に示すように、並列処理の対象である参照面データ領域1105を、垂直方向に1ライン単位で走査させながら二次元データ列に対してフィルタ演算を行う。

【0088】

続いて第二の積和演算回路220により、前述した水平方向のフィルタカーネルを用いた畳込み演算の結果に対して、図12(B)及び図13に示すように、1次元構造を有する垂直方向のフィルタカーネルを用いた畳込み演算を実行する。

【0089】

図12(B)は、1つの特徴面データを求める畳込み演算処理の具体例を示す図である。図12(B)では、参照データ面上の2次元座標で示している。

【0090】

1201は、水平方向のフィルタカーネルを用いた畳込み演算の結果である参照データ面を表している。ここで各ブロック(模式的に示す最小一升)は、水平方向のフィルタカーネルを用いた畳込み演算結果(`input(x, y)`、 x ：水平方向位置、 y ：垂直方向位置)を示す。尚、図12(B)及び図13では、水平方向のフィルタカーネルを用いた畳込み演算の結果を、参照面データ全体に配置して表示している。しかし、実際には水平方向のフィルタカーネルを用いた畳込み演算の結果は、第一の積和演算回路200の累積加算器208から第二の積和演算回路220のリングバッファ212に随時入力される。

【0091】

ここでは、前述の式(2)及び(3)を実現するための演算手法を理解し易いように、水平方向のフィルタカーネルを用いた畳込み演算の結果を参照面データ全体に配置して表示している。また、1203は、`output(4, 4)`位置の特徴面データ1204を算出する場合のフィルタ演算における、畳込み演算が実行される参照データの範囲を示している。図12(B)では、垂直方向のフィルタカーネルのサイズが「5」の場合を示しており、参照データ範囲1203は、垂直方向のフィルタカーネルサイズと等しい`input(4, y)`： $y = 2 ~ 6$ となる。

【0092】

実際には、図13(B)の点線で示すように、垂直方向のフィルタカーネルによる演算を1つの係数データごとに実行し、それを垂直方向に5回繰り返すことで畳込み演算における参照データ面に対する演算を実行する。またさらに図13(A)は、フィルタ演算する特徴面データの領域1301を示している。例えば特徴面データの領域1301は、同時にフィルタ演算する特徴面データの領域を示し、例えば並列度が「5」の場合、`output(x, 4) : x = 4 ~ 8`の特徴面データを同時に演算する。尚、図13(A)では、特徴面データと参照データ範囲の対応を示す点線及び一点鎖線は、それぞれ`output(x, 4) : x = 4`及び8についてのみ示している。但し、後述するように、畳込み演算は、垂直方向のフィルタカーネルの1つの係数データごとに実行されるため、参照面データの1ラインごとに積和演算が実行される。即ち、元々のフィルタカーネルのサイズが水平方向に「5」、垂直方向に「5」であるとすると、図13(B)に示すように、並列処理の対象である参照面データの領域1302を垂直方向に1ライン単位で走査させなが

10

20

30

40

50

らフィルタ演算を行う。こうして二次元データ列に対してフィルタ演算を行うことになる。この場合に、参照面データの垂直方向に1ライン単位で走査させる機能は、第二の積和演算回路220におけるリングバッファ212によって実現される。

【0093】

以上説明したように、参照データに対して水平方向のフィルタカーネルを用いた畳込み演算と、垂直方向のフィルタカーネルを用いた畳込み演算を実行した場合、その演算結果は式(2)で表わされる。ここで前述したように、2次元構造を有するフィルタカーネルの重み係数マトリクスは、式(2)に示す乗算によって近似可能である。このため、以上の演算処理により、一般的な2次元構造を有するフィルタカーネルによる畳込み演算処理の近似演算を実行することが可能となる。

10

【0094】

以上説明したように、水平方向及び垂直方向のフィルタカーネルによる畳込み演算は、特徴面データの領域1104及び1301を垂直方向に1ライン単位、水平方向に5画素単位でフィルタを同時に走査してフィルタ演算することによりなされる。こうして二次元データ列に対して高速にフィルタ演算を行うことができる。

【0095】

図15は、実施形態1に係るCNN処理部122における畳込み演算の処理時の動作を説明するタイムチャートである。図15では、1つの特徴面の畳込み演算処理動作(S1004)の一部を示してある。また、図15に示す信号は、全てクロック信号(図示せず)に基づいて同期して動作する。尚、前述したように実施形態1に係る畳込み演算処理は、水平方向のフィルタカーネルを用いた畳込み演算と、垂直方向のフィルタカーネルを用いた畳込み演算が並行して実行される。

20

【0096】

Load1信号及びLoad6信号は、記憶部202及び記憶部210に重みデータをロードするイネーブル信号を示す。制御部201は、これらLoad信号1及びLoad6信号が有効(信号レベルが1)の期間に、RAM100から水平方向及び垂直方向のそれぞれの重みデータを読み出し、記憶部202及び記憶部210にそれぞれを書き込む。このとき水平方向及び垂直方向のフィルタカーネルのサイズはレジスタ群702に保持されている。また制御部201は、レジスタ群702で指定する重み係数格納先ポインタ、カーネルサイズ及び参照データ面の数に基づいて、読み出す重みデータのアドレスを決定する。ここでRAM100のデータ幅は32ビットであり、重み係数のデータ幅は8ビットとする。この場合、記憶部202及び記憶部210に水平方向に12個の重み係数を書き込む場合、3クロックでロード処理を完了する。尚、以後、RAM100に対する読み出し及び書き込みサイクルは、全て1クロックで完了するものとする。

30

【0097】

また実施形態1では、水平方向フィルタのカーネルサイズと、垂直方向フィルタのカーネルサイズは等しいものとしているため、記憶部210に垂直方向の12個の重み係数を書き込むのに必要な時間は、前述した水平方向の場合と同一である。従って実施形態1では、水平方向フィルタカーネルの重み係数と、垂直方向フィルタカーネルの重み係数の読み出し及び書き込み処理は、同時に完了する。

40

【0098】

次に制御部201は、重み係数を各記憶部にロードすると、次に第一の積和演算回路200における参照データのロードを開始するためにLoad3信号を有効化する。Load3信号もLoad1信号と同様に、信号レベルが「1」のときに有効化された状態となる。制御部201はLoad3信号を「1」にするのと同時に、RAM100から参照データを読み出して記憶部203にセットする。セットするデータの数はレジスタ群702に保持されているフィルタカーネルのサイズ及び並列度から決定する。また制御部201は、レジスタ群702で指定する参照データの格納先ポインタ、参照データの水平及び垂直サイズ及び参照データ面の数に基づいて、RAM100から読み出す参照データのアドレスを決定する。ここで参照データの有効桁は8ビットであるため、記憶部203に例え

50

ば 23 個の参照データを書き込む場合、6 サイクルで書き込みシーケンスを完了する。例えば、水平方向フィルタカーネルのサイズが「12」であるとすると、並列度が 23 ($12 + 12 - 1 = 23$ 個) のデータをロードする必要がある。

【0099】

* CLR 信号は、第一及び第二の積和演算回路 200, 220 における累積加算器 208 及び 214 を初期化させる信号であり、* CLR 信号が 0 (ロウレベル) の場合、累積加算器 208 及び 214 のレジスタ 502 (図 5) が「0」に初期化される。制御部 201 は、新たな特徴面データの畳込み演算開始前に、この * CLR 信号を 0 にする。

【0100】

Load 2 信号は、シフトレジスタ 204 の初期化を指示する信号であり、Load 2 信号が 1 で、かつ Enable 1 信号が有効 (信号レベル 1) の場合、記憶部 202 に保持する複数の重み係数データがシフトレジスタ 204 に一括してロードされる。Enable 1 信号は、シフトレジスタ 204 のデータシフトを制御する信号であるが、図 15 に示すように動作中は常に 1 に設定されている。従って、Load 2 信号が 1 の場合、シフトレジスタ 204 はクロック信号に応じて記憶部 202 の出力をラッチし、Load 2 信号が 0 の場合はクロック信号に応じてシフト処理を継続する。制御部 201 のシーケンス制御部 701 は、フィルタカーネルのサイズに応じたクロック数をカウントすると Load 2 信号を有効化する。これにより、シフトレジスタ 204 のシフト動作を停止させると同時に、記憶部 202 に保持する重み係数データをシフトレジスタ 204 に一括ロードする。即ち、フィルタカーネルの水平方向単位で重み係数データを一括してシフトレジスタ 204 にロードし、ロードした係数を動作クロックに応じてシフトアウトする。

【0101】

また同様に Load 7 信号は、シフトレジスタ 211 の初期化を指示する信号であり、Load 7 信号が 1 で、かつ Enable 4 信号が有効 (信号レベル 1) の場合、記憶部 210 に保持する複数の重み係数データがシフトレジスタ 211 に一括ロードされる。Enable 4 信号は、シフトレジスタ 211 のデータシフトを制御する信号であり、図 15 に示すように動作中は常に 1 に設定されている。従って、Load 7 信号が 1 の場合、シフトレジスタ 211 は、クロック信号に応じて記憶部 210 の出力をラッチし、Load 7 信号が 0 の場合、クロック信号に応じてシフト処理を継続する。制御部 201 のシーケンス制御部 701 は、フィルタカーネルのサイズに応じたクロック数をカウントすると Load 7 信号を有効化する。これによりシフトレジスタ 211 のシフト動作を停止させると同時に、記憶部 210 に保持する重み係数データをシフトレジスタ 211 に一括ロードする。即ち、フィルタカーネルの垂直方向単位で重み係数データを一括してシフトレジスタ 211 にロードし、ロードした係数を動作クロックに応じてシフトアウトする。

【0102】

Load 4 信号は、シフトレジスタ 205 の初期化を指示する信号であり、Load 4 信号が 1 で、かつ Enable 2 信号が有効 (信号レベル 1) の場合、記憶部 203 に保持されている参照データがシフトレジスタ 205 に一括ロードされる。尚、Enable 2 信号はシフトレジスタ 205 のデータシフトを制御する信号であるが、図 15 に示すように動作中は常に 1 に設定されている。このため Load 4 信号が 1 の場合、シフトレジスタ 205 はクロック信号に応じて記憶部 203 の出力をラッチし、Load 4 信号が 0 の場合、クロック信号に応じてシフト処理を継続する。制御部 201 のシーケンス制御部 701 は、垂直方向フィルタのカーネルサイズに応じたクロック数をカウントすると Load 4 信号を有効化し、シフトレジスタ 205 のシフト動作を停止させると同時に、記憶部 203 に保持する参照データを一括ロードする。即ち、フィルタカーネルの 1 列単位で必要な参照データを一括してシフトレジスタ 205 にロードし、ロードした参照データを動作クロックに応じてシフトする。このように、制御部 201 は、Load 4 信号を Load 2 信号と同一のタイミングで制御する。

【0103】

累積加算器 208 は、クロックに同期して積和演算を実行する。従って、シフトレジス

10

20

30

40

50

タ204及び205のシフト動作に従って、算出する特徴面データの複数の点に対して同時にフィルタカーネルのサイズに応じた積和演算処理を実行する。具体的には、シフトレジスタ204及び205のシフト動作期間(図15の積和演算処理区間)中に水平方向フィルタカーネル分の積和演算がなされる。このような行単位の演算を、重み係数データ及び参照データを入れ替えながら水平方向に繰り返すことにより、並列度の数に応じた二次元の畳込み演算結果が生成される。また制御部201は、フィルタカーネルのサイズ及び並列度に応じて各信号を制御し、積和演算処理及び積和演算処理に必要なデータ(重み係数データ及び参照データ)のRAM100からの供給を並行して実行する。

【0104】

Load8信号は、リングバッファ212の先頭レジスタの初期化を指示する信号である。Load8信号が1で、かつEnable5信号が有効(信号レベル1)の場合、第一の積和演算回路200の累積加算器208に保持される参照データが、リングバッファ212の先頭レジスタに一括ロードされる。尚、Enable5信号はリングバッファ212のデータシフトを制御する信号であり、図15に示すように動作中は常に1に設定されている。このためLoad8信号が1の場合、リングバッファ212はクロック信号に応じて第一の積和演算回路200の累積加算器208の出力をラッチする。またLoad8信号が0の場合、リングバッファ212はクロック信号に応じてリング状のシフト処理を実行する。制御部201のシーケンス制御部701は、垂直方向フィルタカーネルのサイズに応じたクロック数をカウントするとLoad8信号を有効化する。これによりリングバッファ212のシフト動作を停止させると同時に、第一の積和演算回路200の累積加算器208に保持されている参照データを一括してリングバッファ212にロードする。即ち、リングバッファ212は、カーネル単位で必要な参照データを一括ロードし、ロードした参照データを動作クロックに応じてシフトする。このように、制御部201はLoad8信号をLoad7信号と同一のタイミングで制御する。

【0105】

累積加算器214は、クロックに同期して積和演算を継続している。従って、シフトレジスタ211及びリングバッファ212のシフト動作に従って算出する特徴面データの複数の点に対して同時にフィルタのカーネルサイズに応じた積和演算処理を実行する。具体的には、シフトレジスタ211及びリングバッファ212のシフト動作期間(図15の積和演算処理区間)中に垂直方向フィルタカーネル分の積和演算がなされる。このような列単位の演算を、重み係数データ及び参照データを入れ替ながら垂直方向に繰り返し、並列度の数に応じた畳込み演算結果が生成される。また制御部201は、フィルタのカーネルサイズ及び並列度に応じて各信号を制御し、積和演算処理及び積和演算処理に必要なデータ(重み係数データ)のRAM100からの供給を並行して実行する。

【0106】

続いてLoad5信号は、第二の積和演算回路220における累積加算器214の結果をシフトレジスタ206に並列にロードする信号である。制御部201は垂直方向フィルタカーネルの積和演算が終了すると、Load5信号(図示せず)及びEnable3信号を1にする。シフトレジスタ206は、Load5信号が1で、かつEnable3信号が1の場合、累積加算器214の出力を一括ロードする。尚、図15では、計算済みの畳込み演算結果がシフトレジスタ206にラッチされているものとする。制御部201は、シフトレジスタ204, 205及び211とリングバッファ212のシフト動作中に記憶部202, 210及び203へのデータロードが完了している場合、Enable3信号を有効化(1)する。これによりシフトレジスタ206に保持している演算結果が、シフトアウトされる。つまり、信号OUTn(シフトレジスタ206の最終段出力)を非線形変換処理部209に向けて出力させる。こうしてシフトレジスタ206からシフトアウトされた演算結果は、非線形変換処理部209によって変換処理される。その後、制御部201は、レジスタ群702に記された演算結果格納先ポインタ及び参照データのサイズに従って、変換処理された演算結果をRAM100の所定のアドレスに格納する。尚、式(3)に示すような、複数組の水平方向及び垂直方向のフィルタカーネルによる畳込み演

10

20

30

40

50

算結果の累積値を算出する場合、累積加算器により、複数組の水平方向及び垂直方向のフィルタカーネルによる畳込み演算結果の累積加算を実行する。そして、複数組の水平方向及び垂直方向のフィルタカーネルによる畳込み演算結果の累積演算が完了した時点で、前述と同様にして、累積加算器 214 の演算結果をシフトレジスタ 206 に並列にロードする。Load 5 信号により累積加算器の結果をシフトレジスタ 206 に並列にロードする処理以降の処理は、上述した処理内容と同一である。

【0107】

このように制御部 201 は、第一の積和演算回路 200 の記憶部 202 及び記憶部 203 の RAM 100 に対するアクセス、第二の積和演算回路 220 の記憶部 210 の RAM 100 に対するアクセスを調停する。更に、非線形変換処理部 209 の RAM 100 に対するアクセスを調停し、積和演算処理及び当該 4 つの処理部 (202, 203, 210 及び 209) の RAM 100 へのアクセスをパイプライン化する。例えば、水平方向及び垂直方向のフィルタのカーネルサイズがともに「24」で、演算並列度が 12 の場合を考える。「水平方向フィルタカーネル及び垂直方向フィルタカーネルの積和演算に必要なクロック数 (24)」は、「重み係数ロードクロック数 (3) × 2 + 参照データロードクロック数 (9) + 結果データ格納クロック数 (1)」より大きい。このため、メモリアクセスに要する時間は、積和演算時間内に隠蔽される。

【0108】

尚、非線形変換処理部 209 は記憶部 202, 203 及び 210 に比べて RAM 100 に対するアクセス頻度が低いため、最も低い優先順位で動作する。即ち、非線形変換処理部 209 は、記憶部 202, 203 及び 210 のアクセスの間隙となるタイムスロットで RAM 100 にアクセスする。

【0109】

また実施形態 1 では、RAM 100 からの参照データの読み出し、重み係数データの読み出し及び演算結果の書き出しを積和演算処理期間に並行して実行する。従って、「積和演算時間 (フィルタカーネルの水平サイズ + フィルタカーネルの垂直サイズ) + シフトレジスタへのロード時間 (水平方向フィルタカーネルのサイズ)」×結合する前階層の特徴面数」分のクロックで、並列度分の数の畳込み演算処理が完了する。

【0110】

また、フィルタカーネルのサイズが小さい場合等、並列度とフィルタカーネルとの関係によっては、RAM 100 へのアクセスを積和演算期間中に完全にパイプライン化できない場合もある。このような場合、制御部 201 は RAM 100 へのアクセス完了を優先する。そして Enable 1, Enable 2, Enable 3, Enable 4 信号及び Enable 5 信号、及び累積加算器の Latch Enable 信号を制御して積和演算処理の開始を遅延させることが好ましい。つまり、記憶部 202, 203 及び 210 へのデータロード及び非線形変換処理部 209 のデータセーブを行うことが好ましい。

【0111】

図 16 (A) は、実施形態 1 に係る第一の積和演算回路 200 における積和演算処理と、第二の積和演算回路 220 における積和演算処理のパイプライン動作のタイミングを説明する模式図である。図 16 (A) では、紙面の都合により、水平方向及び垂直方向のフィルタカーネルのサイズがそれぞれ「3」である場合を示している。

【0112】

図 16 (A) で、h1~h3 は、第一の積和演算回路 200 による水平方向のフィルタカーネルのフィルタ係数を示し、図中のフィルタ係数が記述された各四角形の幅が示す 1 クロックで、そのフィルタ係数による積和演算が実行されることを示している。また、v1~v3 は、第二の積和演算回路 220 における垂直方向のカーネルのフィルタ係数を示している。図中のフィルタ係数が記述された各四角形の幅が示す 1 クロック内で、そのフィルタ係数による積和演算が実行されることを示している。

【0113】

前述したように第一の積和演算回路 200 では、各フィルタ係数 h1~h3 による積和演

10

20

30

40

50

算が、それぞれ 1 クロックごとに実行される。こうしてフィルタのカーネルサイズに一致する 3 クロックで 1 ライン分の参照データの演算が完了し、積和演算結果は第二の積和演算回路 220 に対して出力される。第二の積和演算回路 220 では、第一の積和演算回路 200 から入力されたデータの 3 ライン分に対して、フィルタのカーネルサイズに一致する 3 クロックで 3 ライン分の参照データの演算が完了する。図 16 (A) では、それぞれ算出された積和演算結果が出力されるタイミングを矢印で示している。

【0114】

このように、第一の積和演算回路 200 における積和演算処理と、第二の積和演算回路 220 における積和演算処理は、図 16 (A) に示すようにパイプライン処理として実行される。

10

【0115】

次に、畳込み演算処理の終了に伴う割り込み処理について説明する。

【0116】

図 10 (B) は、CPU127 による割り込み処理時の動作を示すフローチャートである。

【0117】

割り込み処理時、CPU127 が制御部 201 からの終了通知割り込みを受信すると S 1010 で CPU127 は DMA C26 を起動し、RAM100 の最終特徴面データを CPU バス 130 上の RAM129 に転送する。次に CPU127 は、RAM129 に記憶されている最終層検出結果から、検出対象である所定の物体の位置及び大きさ等の情報を取得する。具体的には、最終の検出結果を二値化処理し、ラベリング等の処理によりオブジェクト位置及びサイズ等を抽出する。このようにして一連の処理が終了する。

20

【0118】

そして実施形態 1 によれば、少ない回路規模 (レジスタサイズ等) でフィルタカーネルのサイズ単位で演算及びメモリアクセスをパイプライン化できる。そして 2 次元構造を有するフィルタカーネルを、水平方向及び垂直方向の 2 つの 1 次元構造を有するフィルタカーネルに分解した階層的な畳込み演算を高速に実行できる。また参照データ及び演算結果データを、同一のメモリ上に配置する構成としているため、レジスタの設定だけで複雑な階層的な畳込み演算処理に柔軟に対応することができる。

【0119】

30

つまり、階層的な畳込み演算を低コストで高速に実行でき、さらに低消費電力化を実現することが可能となる。また、同一の回路で様々な階層的結合を有する畳込み演算を高速に実行することができる。

【0120】

[実施形態 2]

次に、本発明の実施形態 2 を説明する。実施形態 2 に係る演算装置は、水平方向フィルタのカーネルサイズと、垂直方向フィルタのカーネルサイズとが異なる点が実施形態 1 と相違しており、他の構成等は実施形態 1 と同様である。従って、実施形態 2 では、実施形態 1 と相違する部分についてのみ説明し、その他の部分に関しては実施形態 1 と同様として説明を省略する。実施形態 2 では、水平方向フィルタのカーネルサイズが「4」、垂直方向フィルタのカーネルサイズが「3」とする。

40

【0121】

実施形態 1 で説明した畳込み演算装置は、図 16 (A) で説明したように第一の積和演算回路 200 と第二の積和演算回路 220 がパイプライン動作を行う。しかしながら実施形態 2 のように、水平方向フィルタのカーネルサイズと、垂直方向フィルタのカーネルサイズとが異なる場合、第一の積和演算回路 200 と第二の積和演算回路 220 におけるそれぞれの積和演算処理ステップ数が一致しない。即ち、第一の積和演算回路 200 では、シフトレジスタ 205 が参照データを 4 回シフトする間に積和演算が完了するのに対して、第二の積和演算回路 220 では、リングバッファ 212 が参照データを 3 回シフトする間に積和演算が完了する。このように実施形態 2 では、第一の積和演算回路 200 と第二

50

の積和演算回路 220 それぞれの積和演算処理のステップ数において、(4 - 3 =) 1 回のズレが生じる。そこで実施形態 2 では、図 16 (C) に示すように、制御部 201 によって制御されるパイプライン処理中の、第二の積和演算回路 220 の演算処理に対して、演算を実行しないストールを挿入する。これにより、図 16 (C) から分るように、第一の積和演算回路 200 で積和演算処理が完了するタイミングと、第二の積和演算回路 220 で積和演算処理が完了するタイミングを一致させることができる。尚、実施形態 2 では、水平方向フィルタのカーネルサイズが、垂直方向フィルタのカーネルサイズよりも大きい場合で説明した。しかし、逆に垂直方向フィルタのカーネルサイズが水平方向フィルタのカーネルサイズよりも大きい場合は、図 16 (B) に示すように、積和演算処理ステップ数のズレ分のストールステージを、第一の積和演算回路 200 の演算ステージに挿入すれば良い。また当然のことながら、水平方向及び垂直方向フィルタのカーネルサイズは、上記の例に限るものでは無く、任意のサイズで構わない。その場合は、水平方向及び垂直方向フィルタのカーネルサイズの差に対応するストールステージを挿入すれば良い。10

【0122】

以上説明したように実施形態 2 によれば、水平方向フィルタのカーネルサイズと、垂直方向フィルタのカーネルサイズとが異なる場合でも、階層的な畳込み演算を低成本で高速に実行できる。更に、低消費電力化を実現することが可能となる。また、同一の回路で様々な階層的結合を有する畳込み演算を高速に実行できる。

【0123】

【実施形態 3】

20

次に本発明の実施形態 3 について説明する。実施形態 3 に係る畳込み演算装置は、実施形態 1 の第一の積和演算回路 200 が第二の積和演算回路 220 の後段に直列に接続されている点が実施形態 1 と相違しており、他の構成等は実施形態 1 と同様である。従って、実施形態 3 では実施形態 1 と相違する部分についてのみ説明し、その他の部分に関しては実施形態 1 と同様として説明を省略する。

【0124】

図 17 は、本発明の実施形態 3 に係る CNN 処理部 122 の構成を説明するブロック図である。ここでは、第二の積和演算回路 220 の後段に第一の積和演算回路 200 が直列に接続されている。尚、図 17 において、前述の実施形態 1 に係る図 2 と共通する部分は同じ参照番号で示している。30

【0125】

図 17 では、第二の積和演算回路 220 のリングバッファ 212 に対して、記憶部 203 に保持されている参照データが一括してロードされる。また第一の積和演算回路 200 のシフトレジスタ 205 には、第二の積和演算回路 220 の累積加算器 214 に保持される参照データが一括してロードされる。以上の実施形態 1 との相違点に関して、さらに詳細に説明する。

【0126】

図 18 は、実施形態 3 における畳込み演算処理時の動作を説明するタイムチャートである。図 18 は、1 つの特徴面データの畳込み演算処理動作 (S1004) の一部を示している。また図 18 に示す信号は、全てクロック信号 (図示せず) に基づいて同期動作する。40

【0127】

Load 9 信号は、リングバッファ 212 の先頭レジスタの初期化を指示する信号である。Load 9 信号が 1 で、かつ Enable 6 信号が有効 (1) の場合、記憶部 203 に保持されている参照データがリングバッファ 212 の先頭レジスタに一括してロードされる。尚、Enable 6 信号は、リングバッファ 212 のデータシフトを制御する信号であるが、図 18 に示すように動作中は常に 1 に設定されている。このため、Load 9 信号が 1 になると、リングバッファ 212 はクロック信号に応じて記憶部 203 の出力をラッチし、Load 9 信号が 0 のときはクロック信号に応じてリング状のシフト処理を継続する。50

【0128】

制御部201のシーケンス制御部701は、垂直方向フィルタのカーネルサイズに応じたクロック数をカウントするとLoad9信号を有効化(1)して、リングバッファ212のシフト動作を停止させる。これと同時に、記憶部203に保持している参照データを一括してリングバッファ212にロードする。即ち、フィルタのカーネル単位で必要な参照データを、一括してリングバッファ212にロードし、ロードした参照データを動作クロックに応じてリング状にシフトする。ここで制御部201は、Load9信号を、実施形態1で説明した重み係数データを制御するLoad7信号と同一のタイミングで制御する。

【0129】

累積加算器214は、クロックに同期して積和演算を継続している。従って、シフトレジスタ211及びリングバッファ212のシフト動作に従って、算出する特徴面データの複数の点に対して、同時にカーネルサイズに応じた積和演算処理を実行する。これは実施形態1と同様である。具体的には、シフトレジスタ211及びリングバッファ212のシフト動作期間(図18の積和演算処理区間)中に、垂直方向フィルタのカーネル分の積和演算がなされる。このような垂直方向フィルタのカーネル単位の演算を、重み係数データ及び参照データを入れ替えながら垂直方向に繰り返し、並列度の数に応じた畳込み演算結果が生成される。また制御部201は、フィルタのカーネルサイズ及び並列度に応じて各信号を制御し、積和演算処理及び積和演算処理に必要なデータ(重み係数データ)のRAM100からの供給を並行して実行する。

10

【0130】

続いて、Load10信号は、シフトレジスタ205の初期化を指示する信号である。Load10信号が1で、かつEnable7信号が有効(信号レベル1)の場合、第二の積和演算回路220の累積加算器214に保持される参照データがシフトレジスタ205に一括してロードされる。尚、Enable7信号は、シフトレジスタ205のデータシフトを制御する信号であるが、図18に示すように、動作中は常に1に設定されている。このためLoad10信号が1の場合、シフトレジスタ205は、クロック信号に応じて第二の積和演算回路220の累積加算器214の出力をラッチし、Load10信号が0の場合、クロック信号に応じてシフト処理を継続する。

20

【0131】

制御部201のシーケンス制御部701は、水平方向フィルタのカーネルサイズに応じたクロック数をカウントするとLoad10信号を有効化してシフトレジスタ205のシフト動作を停止させる。これと同時に、第二の積和演算回路220の累積加算器214に保持されている参照データを一括してシフトレジスタ205にロードする。即ち、水平方向フィルタのカーネル単位で必要な参照データを一括してシフトレジスタ205にロードし、ロードした参照データを動作クロックに応じてシフトする。このように制御部201は、Load10信号を実施形態1で説明した重み係数データを制御するLoad2信号と同一のタイミングで制御する。

30

【0132】

累積加算器208は、クロックに同期して積和演算を継続している。従って、シフトレジスタ204及び205のシフト動作に従って、算出する特徴面データの複数の点に対して、同時にフィルタのカーネルサイズに応じた積和演算処理を実行する。これは実施形態1と同様である。具体的には、シフトレジスタ204及び205のシフト動作期間(図18の積和演算処理区間)中に水平方向フィルタのカーネル分の積和演算がなされる。このような水平方向単位の演算を、重み係数データ及び参照データを入れ替えながら水平方向に繰り返し、並列度の数に応じた畳込み演算結果が生成される。また制御部201は、フィルタのカーネルサイズ及び並列度に応じて各信号を制御し、積和演算処理及び積和演算処理に必要なデータ(重み係数データ及び参照データ)のRAM100からの供給を並行して実行する。

40

【0133】

50

以上説明したように実施形態3によれば、第二の積和演算回路220の後段に第一の積和演算回路200が直列に接続されている。この実施形態3に係る畳込み演算装置を用いることにより、実施形態1と同様に、階層的な畳込み演算を低コストで高速に実行できる。また更に、低消費電力化を実現することが可能となる。また、同一の回路で様々な階層的結合を有する畳込み演算を高速に実行できる。

【0134】

【実施形態4】

次に本発明の実施形態4について説明する。実施形態4に係るCNN処理部は、実施形態1における第二の積和演算回路220と同一の構成を有する第三の積和演算回路1900が、第二の積和演算回路220の後段に直列に接続されている点が実施形態1と相違している。他の構成等は実施形態1と同様である。従って、実施形態4では、前述の実施形態1と相違する部分についてのみ説明し、その他の部分に関しては実施形態1と同様として説明を省略する。

【0135】

図19は、実施形態4に係るCNN処理部122の構成を説明するブロック図である。ここでは、第二の積和演算回路220の後段に第三の積和演算回路1900が直列に接続されている。ここで、第三の積和演算回路1900の構成は、第二の積和演算回路220と同様である。

【0136】

次に実施形態4に係る畳込み演算について説明する。実施形態4に係る畳込み演算処理では、複数の特徴面データに対応する、複数の2次元構造を有するフィルタカーネルを水平方向、垂直方向、及び特徴面方向の3つの1次元構造を有するフィルタカーネルに分解して畳込み演算を実行する。3次元構造を有するフィルタカーネル（水平方向、垂直方向及び特徴面データ方向）の重み係数マトリクスは、1次元構造を有する水平方向、垂直方向及び特徴面方向のフィルタカーネルを用いた式（4）で示す乗算によって近似できることが知られている。

【0137】

【数4】

$$output(x, y) = \sum_{featureIndex=0}^{featureSize-1} \sum_{row=0}^{rowSize-1} \sum_{column=0}^{columnSize-1} (input(x + column, y + row, feature) \times weight_{horizontal}(column) \times weight_{vertical}(row) \times weight_{feature}(featureIndex))$$

【0138】

尚、実際には、複数の水平方向、垂直方向及び特徴面方向のフィルタカーネルの組み合わせを用い、式（5）に示すようにそれらのフィルタ演算結果の総和を取ることで近似精度を高める手法が取られることが多い。式（5）では、フィルタカーネルの組み合わせを添え字「pair」で示している。

【0139】

【数5】

$$\begin{aligned}
 output(x, y) = & \sum_{pair=0}^{pairSize-1} \sum_{featureIndex=0}^{featureSize-1} \sum_{row=0}^{rowSize-1} \sum_{column=0}^{columnSize-1} \\
 & (input(x + column, y \\
 & + row, feature) \times weight_{pair, horizontal}(column) \\
 & \times weight_{pair, vertical}(row) \times weight_{pair, feature}(featureIndex))
 \end{aligned}$$

【0140】

10

これらの手法に関しては、非特許文献1で詳細が説明されているため、これ以上の説明は省略する。続いて、前記式(2)及び(3)を実行するための畳込み演算方法に関して説明する。

【0141】

まず第一の積和演算回路200及び第二の積和演算回路220における演算処理は実施形態1と同様に実行されるため、説明を省略する。また第三の積和演算回路1900における演算処理は、実施形態1で説明した第二の積和演算回路220と同様に実行される。第三の積和演算回路1900の参照番号1915～1919で示す各部は、第二の積和演算回路220の210～214で示す各部と同じであるため、その説明を省略する。

20

【0142】

図20は、実施形態4に係る畳込み演算の処理時の動作を説明するタイムチャートである。図20では、1つの特徴面データの畳込み演算処理動作(51004)の一部を示してある。また、図20に示す信号は全てクロック信号(図示せず)に基づいて同期動作する。

【0143】

まずLoad11信号は、記憶部1915に重み係数データをロードするイネーブル信号を示す。制御部201はLoad11信号が有効(信号レベルが1)の期間に、RAM100から特徴面方向の重み係数データを読み出して記憶部1915に書き込む。特徴面方向のフィルタのカーネルサイズは、レジスタ群702に保持されている。また制御部201は、レジスタ群702で指定する重み係数格納先ポインタ、カーネルサイズ及び参照データ面の数に基づいて、読み出す重み係数データのアドレスを決定する。RAM100のデータ幅は32ビットであり、重み係数データのデータ幅は8ビットであるので、記憶部1915に特徴面方向の5個の重み係数データを書き込む場合、3クロックでロード処理を完了する。尚、以後、RAM100に対する読み出し及び書き込みサイクルは全て1クロックで完了するものとする。

30

【0144】

*CLR信号は、第三の積和演算回路1900の累積加算器1919を初期化させる信号であり、*CLR信号が0の場合、累積加算器1919のレジスタ502(図5)が0に初期化される。制御部201は、新たな特徴面データの畳込み演算開始前に、この*CLR信号を0にする。またLoad12信号は、シフトレジスタ1916の初期化を指示する信号である。Load12信号が1で、かつEnable8信号が有効(信号レベル1)のとき、記憶部1915に保持されている複数の重み係数データがシフトレジスタ1916に一括してロードされる。Enable8信号は、シフトレジスタ1916のデータシフトを制御する信号であるが、図20に示すように動作中は常に1に設定されている。従って、Load12信号が1のとき、シフトレジスタ1916は、クロック信号に応じて記憶部1915の出力をラッチし、Load12信号が0の場合、クロック信号に応じてシフト処理を継続する。制御部201のシーケンス制御部701は、フィルタのカーネルサイズに応じたクロック数をカウントするとLoad12信号を有効化してシフトレジスタ1916のシフト動作を停止させる。これと同時に、記憶部1915に保持されて

40

50

いる重み係数データをシフトレジスタ1916に一括してロードする。即ち、フィルタカーネルの特徴面方向単位で、重み係数データを一括してシフトレジスタ1916にロードし、ロードした係数を動作クロックに応じてシフトアウトする。

【0145】

Load13信号は、リングバッファ1917の先頭レジスタの初期化を指示する信号である。Load13信号が1で、かつEnable9信号が有効（信号レベル1）のとき、第二の積和演算回路220の累積加算器214に保持される参照データがリングバッファ1917の先頭レジスタに一括してロードされる。尚、Enable9信号は、リングバッファ1917のデータシフトを制御する信号であり、図20に示すように動作中は常に1に設定されている。このためLoad13信号が1の場合、リングバッファ1917は、クロック信号に応じて第二の積和演算回路220の累積加算器214の出力をラッチし、Load13信号が0の場合、クロック信号に応じてリング状のシフト処理を継続する。制御部201のシーケンス制御部701は、特徴面方向のフィルタのカーネルサイズに応じたクロック数をカウントするとLoad13信号を有効化してリングバッファ1917のシフト動作を停止させる。これと同時に、第二の積和演算回路220の累積加算器124に保持されている参照データを一括してリングバッファ1917にロードする。即ち、フィルタのカーネル単位で必要な参照データを一括してリングバッファ1917にロードし、ロードした参照データを動作クロックに応じてシフトする。このように制御部201は、Load13信号をLoad12信号と同一のタイミングで制御する。

【0146】

累積加算器1919は、クロックに同期して積和演算を継続している。従って、シフトレジスタ1916及びリングバッファ1917のシフト動作に従って、算出する特徴面データの複数の点に対して、同時にフィルタのカーネルサイズに応じた積和演算処理を実行する。具体的には、シフトレジスタ1916及びリングバッファ1917のシフト動作期間（図20の積和演算処理区間）中に特徴面方向のカーネル分の積和演算がなされる。このような特徴面方向のカーネル単位の演算を、重み係数データ及び参照データを入れ替ながら繰り返すことにより、並列度の数に応じた畳込み演算結果が生成される。また制御部201は、フィルタのカーネルサイズ及び並列度に応じて各信号を制御し、積和演算処理及び積和演算処理に必要なデータ（重み係数データ）のRAM100からの供給を並行して実行する。

【0147】

次にLoad5信号は、第三の積和演算回路1900における累積加算器1919の結果をシフトレジスタ206に並列にロードする信号である。制御部201は、特徴面方向フィルタカーネルの積和演算が終了すると、Load5信号（図示せず）及びEnable3信号に1を出力する。シフトレジスタ206は、Load5信号が1で、かつEnable3信号が1の場合、累積加算器1919の出力を一括してロードする。尚、図20では、計算済みの畳込み演算結果がシフトレジスタ206にラッチされているものとする。制御部201は、シフトレジスタ1916及びリングバッファ1917のシフト動作中に、記憶部202, 203, 210及び1915へのデータのロードが完了している場合、Enable3の信号を有効化する。そしてシフトレジスタ206が保持する演算結果をシフトアウトする。つまり、信号OUTn（シフトレジスタ206の最終段出力）を非線形変換処理部209に向けて出力させる。こうしてシフトアウトされた演算結果は、非線形変換処理部209によって変換処理され、その後、制御部201により、レジスタ群702に記された演算結果格納先ポインタ及び参照データのサイズに従ってRAM100の所定のアドレスに格納される。

【0148】

尚、ここで、前述したように、式(5)に示すように複数組の水平方向、垂直方向及び特徴面方向のフィルタカーネルによる畳込み演算結果の累積値を算出する。このとき累積加算器1919は、それら複数組の水平方向及び垂直方向のフィルタカーネルによる畳込み演算結果の累積演算を実行する。そして、複数組の水平方向、垂直方向及び特徴面方向

10

20

30

40

50

のフィルタカーネルによる畳込み演算結果の累積演算が完了した時点で、前述と同様にして、累積加算器 1919 の演算結果をシフトレジスタ 206 に並列にロードする。Load 5 信号により累積加算器 1919 の結果をシフトレジスタ 206 に並列にロードする処理以降の処理は、上述した処理内容と同一である。

【0149】

このように制御部 201 は、第一の積和演算回路 200 における記憶部 202 及び記憶部 203 から RAM100 に対するアクセスと、第二の積和演算回路 220 における記憶部 210 からの RAM100 に対するアクセスとを調停する。更に、第三の積和演算回路 1900 の記憶部 1915 の RAM100 に対するアクセスと、更に非線形変換処理部 209 の RAM100 に対するアクセスを調停する。こうして積和演算処理及び当該 3 つの処理部 (202, 203, 209, 210 及び 1915) の RAM100 へのアクセスを 10 パイプライン化する。

【0150】

例えば、水平方向、垂直方向及び特徴面方向のフィルタのカーネルサイズが 24 で、演算並列度が 12 の場合を考える。「水平方向フィルタカーネル、垂直方向フィルタカーネル及び特徴面方向フィルタカーネルの積和演算に必要なクロック数 (24)」が「重み係数ロードクロック数 (3) × 3 + 参照データロードクロック数 (9) + 結果データ格納クロック数 (1)」より大きい。このため、RAM100 へのアクセスに要する時間は、実施形態 1 と同様に、積和演算時間内に隠蔽される。尚、非線形変換処理部 209 は、記憶部 202, 203, 210 及び 1915 に比べて RAM100 に対するアクセス頻度が 20 低いため、最も低い優先順位で動作する。即ち、記憶部 202, 203, 210 及び 1915 のアクセスの間隙となるタイムスロットでアクセスを行う。

【0151】

また実施形態 4 では、RAM100 からの参照データの読み出し、重み係数データの読み出し、及び演算結果の書き出しを積和演算処理期間に並行して実行する。従って、「積和演算時間 (水平方向フィルタのカーネルサイズ + 垂直方向フィルタのカーネルサイズ + 特徴面方向フィルタのカーネルのサイズ) + シフトレジスタへのロード時間 (水平方向フィルタカーネルのサイズ)」× 結合する前階層の特徴面数」分のクロックで、並列度分の数の畳込み演算処理が完了する。

【0152】

但し、フィルタのカーネルサイズが小さい場合等、並列度とフィルタカーネルとの関係によっては、RAM100 へのアクセスを積和演算期間中に完全にパイプライン化できない場合もある。このような場合、制御部 201 は RAM100 へアクセス完了を優先し、Enable1 ~ Enable5 信号、Enable8 ~ Enable9 信号及び累積加算器の Latch Enable 信号を制御して積和演算処理の開始を遅延させるのが好ましい。つまり、記憶部 202, 203, 210 及び 1915 へのデータロード及び非線形変換処理部 209 のデータセーブを行うことが好ましい。

【0153】

図 21 は、実施形態 4 に係る第一の積和演算回路 200 の積和演算処理と、第二の積和演算回路 220 の積和演算処理と、第三の積和演算回路 1900 の積和演算処理のパイプライン動作の模式図である。尚、図 21 では、紙面の都合により、水平方向、垂直方向及び特徴面方向のフィルタカーネルのサイズがそれぞれ 3 である場合を示している。

【0154】

図 21 中で、h1 ~ h3 は、第一の積和演算回路 200 における水平方向フィルタの重み係数データを示し、図中の係数データが記述された各四角形の幅が示す 1 クロック内で、当該係数データによる積和演算が実行されることを示している。また、v1 ~ v3 は、第二の積和演算回路 220 における垂直方向フィルタの重み係数データを示し、図中の係数データが記述された各四角形の幅が示す 1 クロック内で、当該係数データによる積和演算が実行されることを示している。また、f1 ~ f3 は、第三の積和演算回路 1900 における特徴面方向フィルタの重み係数データを示し、図中の係数データが記述された各四角形の 50

幅が示す1クロック内で、当該係数データによる積和演算が実行されることを示している。

【0155】

前述したように第一の積和演算回路200では、各係数データh1～h3による積和演算が、それぞれ1クロックごとに実行される。またフィルタのカーネルサイズに一致する3クロックで、1ライン分の参照データの演算が完了し、積和演算結果は第二の積和演算回路220に出力される。第二の積和演算回路220では、第一の積和演算回路200より入力された3ライン分のデータに対して、フィルタのカーネルサイズに一致する3クロックで、3ライン分の参照データの演算が完了する。第三の積和演算回路1900は、第二の積和演算回路220から入力された3ライン分のデータに対して、フィルタのカーネルサイズに一致する3クロックで、3ライン分の参照データの演算が完了する。図21では、それぞれ算出された積和演算結果が出力されるタイミングを矢印で示している。

【0156】

このように、第一の積和演算回路200の積和演算処理と、第二の積和演算回路220の積和演算処理と、第三の積和演算回路1900の積和演算処理は、図21に示すようにパイプライン処理として実行される。尚、畳込み演算処理の終了に伴う割り込み処理については、実施形態1と同様であるため、その説明を省略する。

【0157】

尚、以上説明した実施形態4では、特徴面方向フィルタのカーネルに関する演算を第二の積和演算回路220で実行し、垂直方向フィルタのカーネルに関する演算を第三の積和演算回路1900で実行しても良い。この場合、それぞれの積和演算回路で演算対象とするフィルタのカーネルが入れ替わるだけで、上記で説明した演算方法は全て同一である。

【0158】

また更に前述の実施形態3で説明したように、第一の積和演算回路200が第二の積和演算回路220もしくは第三の積和演算回路1900の後段に直列に接続されるものであっても構わない。

【0159】

以上説明したように実施形態4によれば、少ない回路規模（レジスタサイズ等）でフィルタのカーネルサイズ単位で演算及びメモリアクセスをパイプライン化できる。こうして、3次元構造を有するフィルタカーネルを、水平方向、垂直方向及び特徴面方向の3つの1次元構造を有するフィルタカーネルに分解した階層的な畳込み演算を高速に実行できる。また、参照データ及び演算結果データを、同一のメモリ空間上に配置する構成であるため、レジスタの設定だけで複雑な階層的畳込み演算処理に柔軟に対応できる。つまり、階層的な畳込み演算を低コストで高速に実行でき、更に、低消費電力化を実現することが可能となる。また、同一の回路で様々な階層的結合を有する畳込み演算を高速に実行できるという効果がある。

【0160】

【実施形態5】

次に本発明の実施形態5について説明する。実施形態5に係る畳込み演算装置は、第二の積和演算回路220と同一の構成を有する第四の積和演算回路が、第三の積和演算回路1900の後段に直列に接続されている点が実施形態4と相違しており、他の構成等は実施形態4と同様である。従って、実施形態5では、実施形態4と相違する部分についてのみ説明し、その他の部分に関しては実施形態1と同様として説明を省略する。

【0161】

図22は、実施形態5に係るCNN処理部122の構成を説明するブロック図である。ここでは、第三の積和演算回路1900の後段に第四の積和演算回路2200が直列に接続されている。ここで、第四の積和演算回路2200の構成は、第二の積和演算回路220と同様であるため、その説明を省略する。即ち、第四の積和演算回路2200の参照番号2220～2224で示す各部は、第二の積和演算回路220の参照番号210～214で示す部分と同じである。

10

20

30

40

50

【 0 1 6 2 】

次に、実施形態 5 に係る畳込み演算について説明する。実施形態 5 における畳込み演算処理では、複数の時系列の参照データ（画像データ）に対して畳込み演算を実行する。即ち、複数の特徴面に対応する、複数の 2 次元構造を有するフィルタカーネルを、水平方向、垂直方向、特徴面方向及び時間方向の 4 つの 1 次元構造を有するカーネルに分解し、参照データとの畳込み演算を実行する。一般に、4 次元構造を有するカーネル（水平方向、垂直方向、特徴面方向及び時間方向）の重み係数マトリクスは、1 次元構造を有する水平方向、垂直方向、特徴面方向及び時間方向のカーネルを用いた式（6）に示す乗算によって近似できることが知られている。

【 0 1 6 3 】

【 数 6 】

$$output(x, y) = \sum_{timeIndex=0}^{timeSize-1} \sum_{featureIndex=0}^{featureSize-1} \sum_{row=0}^{rowSize-1} \sum_{column=0}^{columnSize-1} (input(x + column, y + row, feature, time) \times weight_{horizontal}(column) \times weight_{vertical}(row) \times weight_{feature}(featureIndex) \times weight_{time}(timeIndex))$$

【 0 1 6 4 】

10

20

尚、実際には、水平方向、垂直方向、特徴面方向及び時間方向のフィルタカーネルの組み合わせを用いて式（6）に示すようにそれらの乗算結果の総和を取ることで近似精度を高める手法が取られることが多い。式（7）では、フィルタカーネルの組み合わせを添え字「pair」で示している。

【 0 1 6 5 】

【 数 7 】

$$output(x, y) = \sum_{pair=0}^{pairSize-1} \sum_{timeIndex=0}^{timeSize-1} \sum_{featureIndex=0}^{featureSize-1} \sum_{row=0}^{rowSize-1} \sum_{column=0}^{columnSize-1} (input(x + column, y + row, feature, time) \times weight_{pair, horizontal}(column) \times weight_{pair, vertical}(row) \times weight_{pair, feature}(featureIndex) \times weight_{pair, time}(timeIndex))$$

【 0 1 6 6 】

これらの手法に関しては、非特許文献 1 に詳細が述べられているため、これ以上の説明は省略する。

30

40

【 0 1 6 7 】

上述の式（6）及び（7）を実行するための畳込み演算方法において、第一の積和演算回路 200、第二の積和演算回路 220 及び第三の積和演算回路 1900 における演算処理は実施形態 4 と同様である。また第四の積和演算回路 2200 における演算処理も、実施形態 4 で説明した第三の積和演算回路 1900 の演算処理と同様であるため、その説明を省略する。

【 0 1 6 8 】

以上説明したように実施形態 5 によれば、少ない回路規模（レジスタサイズ等）でフィルタのカーネルサイズ単位で演算及びメモリアクセスをパイプライン化する。こうして 4 次元構造を有するカーネルを水平方向、垂直方向、特徴面方向及び時間方向の 4 つの 1 次

50

元構造を有するフィルタカーネルに分解した階層的な畳込み演算を高速に実行できる。また、参照データ及び演算結果データを同一のメモリ空間に配置するため、レジスタの設定だけで、複雑な階層的畳込み演算処理に柔軟に対応できるという効果がある。

【0169】

このようにして、階層的な畳込み演算を低成本で高速に実行でき、更に、低消費電力化を実現できる。また、同一の回路で様々な階層的結合を有する畳込み演算を高速に実行できる。

【0170】

また実施形態5及び実施形態4から容易に類推できるように、1次元構造を有するフィルタカーネルの数を更に増やした場合でも、対応する積和演算回路を直列に接続することにより、実施形態1～5で説明したのと同様にして畳込み演算を実行できる。

10

【0171】

また実施形態1～5で説明した回路構成は、本発明の畳込み演算装置の一例に過ぎず、同様の機能を実現可能なものであれば、どのような回路構成を取っても構わない。例えば、第二の積和演算回路220におけるリングバッファ212は、これまで説明した回路構成に限らない。例えば、過去に入力されたデータを繰り返し参照可能で、かつ保持したデータを順に参照可能な機能を有する記憶装置であれば、どのような回路構成を採用しても構わない。

【0172】

〔変形例1〕

20

上述の実施形態1～5に係るCNN処理部では、第一の積和演算回路200において、特許文献3と同様の演算を実行することが可能である。その場合は、例えば第二の積和演算回路220において実質的な演算を行わず、第一の積和演算回路200から入力されたデータをそのまま出力することで、同様の演算を実現することが可能である。また或いは第二の積和演算回路220をバイパスして、第一の積和演算回路200の出力を、直接、実施形態1に係るシフトレジスタ206に入力する構成を取っても良い。

【0173】

特許文献3と同様の演算を実行する場合、例えば2次元構造を有するフィルタカーネルをそのまま使用するフィルタ演算処理を実行できる。これにより、フィルタカーネルを1次元構造を有するフィルタカーネルに分解することにより生じ得る、近似による精度劣化を避けることが可能となる。

30

【0174】

〔変形例2〕

上述の実施形態1～5では、各実施形態に係るCNN処理部によってCNN演算処理を実行する例を示したが、実行可能な演算処理はこれに限るものでは無い。即ち、上記式(2)～(7)で示されるような畳込み演算を実行するものであれば、例えば一般的なフィルタ演算を実行することも可能であり、本発明は実行対象となる演算処理を特定するものでは無い。

【0175】

(その他の実施形態)

40

本発明は、上述の実施形態の1以上の機能を実現するプログラムを、ネットワーク又は記憶媒体を介してシステム又は装置に供給し、そのシステム又は装置のコンピュータにおける1つ以上のプロセッサーがプログラムを読み出し実行する処理でも実現可能である。また、1以上の機能を実現する回路(例えば、ASIC)によっても実現可能である。

【0176】

本発明は上記実施形態に制限されるものではなく、本発明の精神及び範囲から離脱することなく、様々な変更及び変形が可能である。従って、本発明の範囲を公にするために、以下の請求項を添付する。

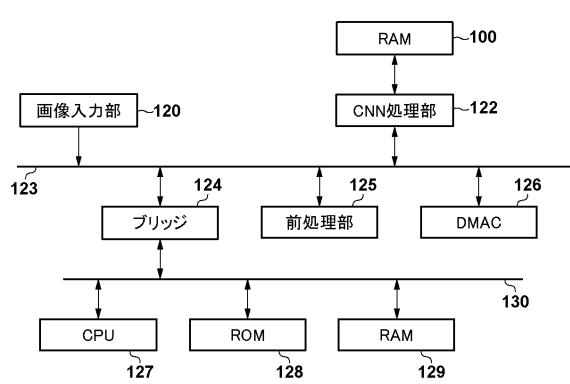
【符号の説明】

【0177】

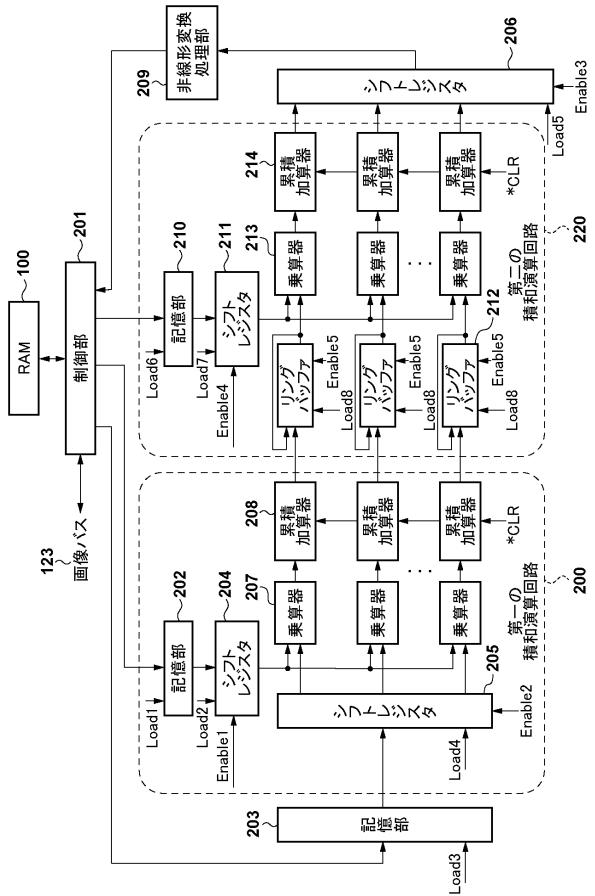
50

100...RAM、120...画像入力部、122...CNN処理部、127...CPU、128...ROM、129...RAM、202, 203, 210...記憶部、204, 205, 211, 206...シフトレジスタ、207, 213...乗算器、208, 214...累積加算器、209...非線形変換処理部、212...リングバッファ

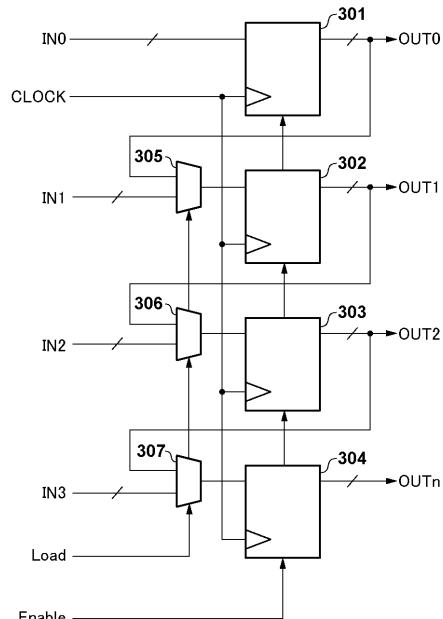
【図1】



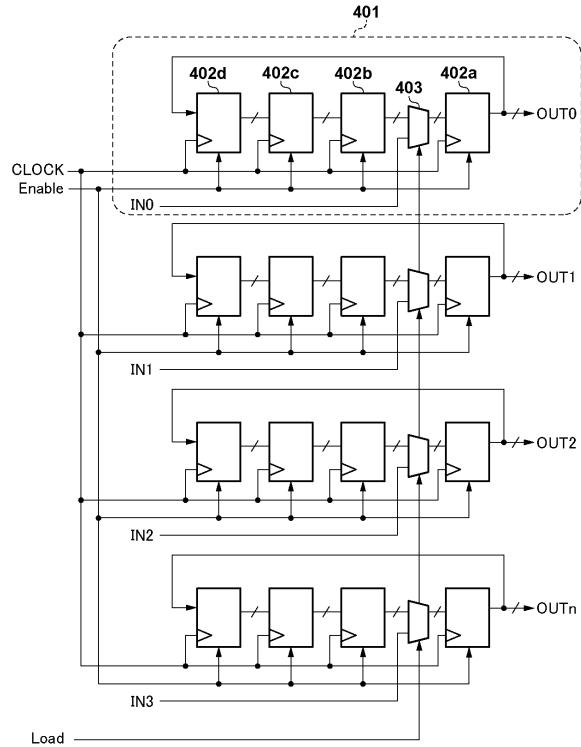
【図2】



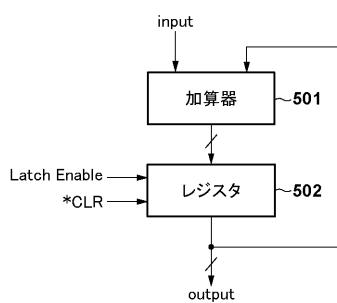
【図3】



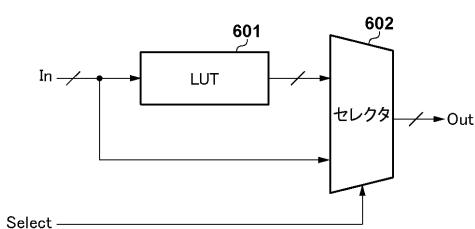
【 図 4 】



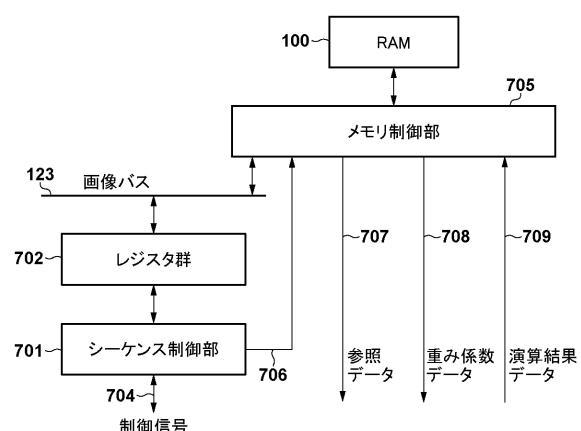
【 図 5 】



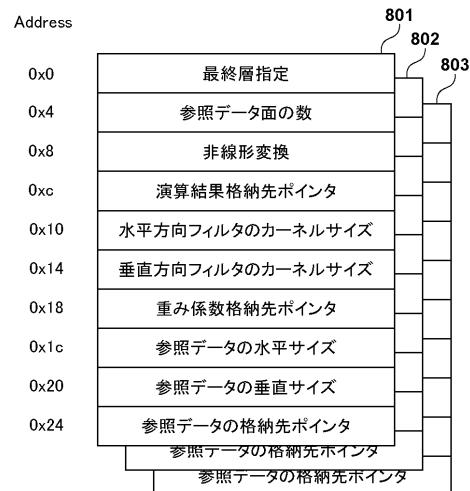
【図6】



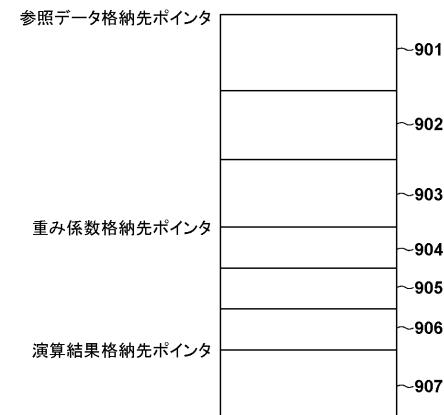
【 図 7 】



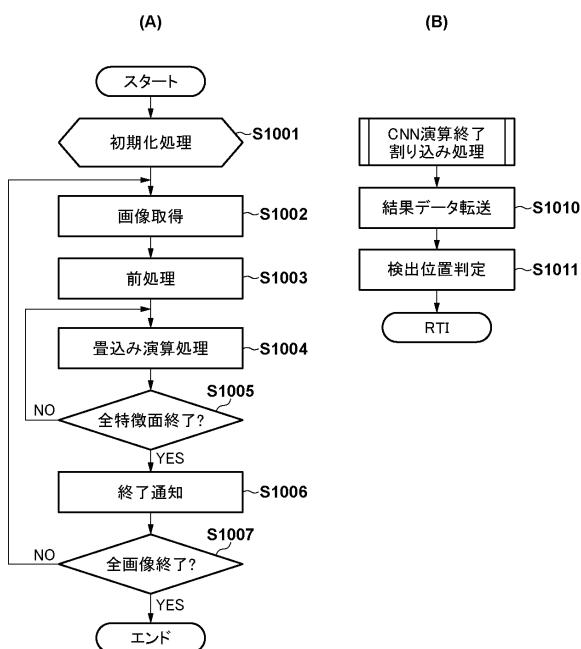
【図 8】



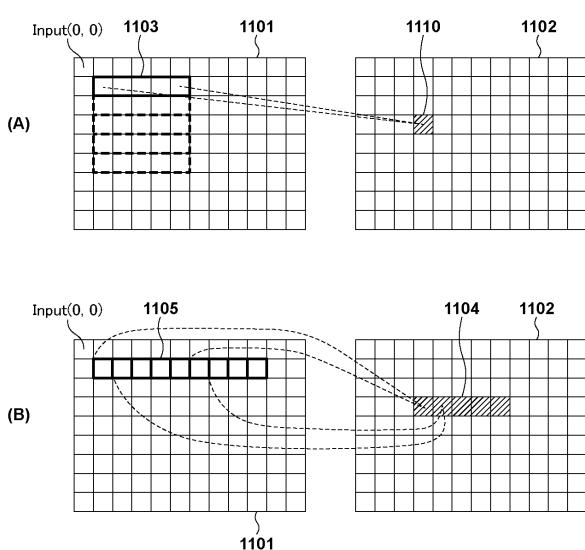
【図 9】



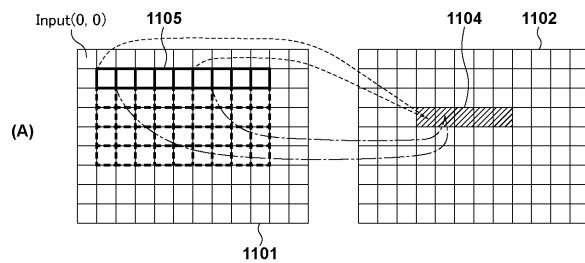
【図 10】



【図 11】



【図12】



(B)

【図13】

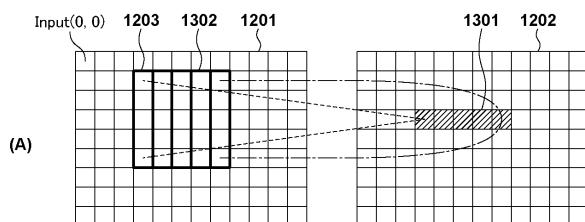
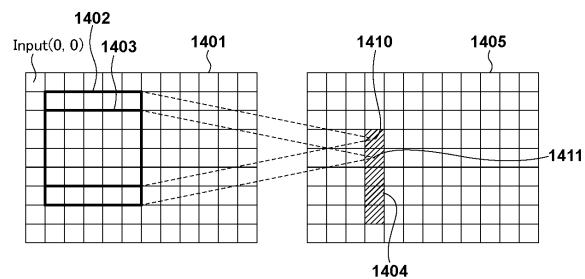
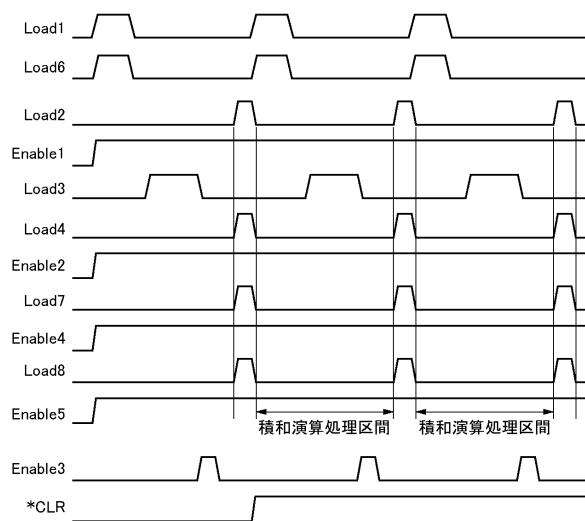


Diagram (B) illustrates a 10x10 grid with various shaded regions and labels. The grid is divided into several sections by dashed lines. Labels include 'Input(0, 0)' at the top left, '1302' and '1201' at the top center, '1301' and '1202' at the top right, and a label '(B)' on the left side. Shaded regions include a central 4x4 square, a 2x2 square in the top-left corner, and a 2x2 square in the bottom-right corner. The grid is bounded by a dashed outer line.

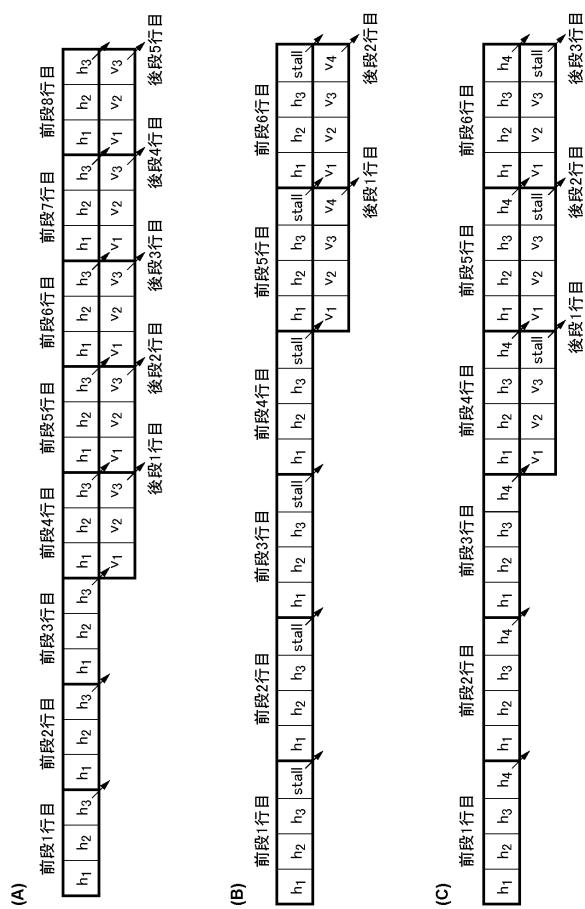
【図14】



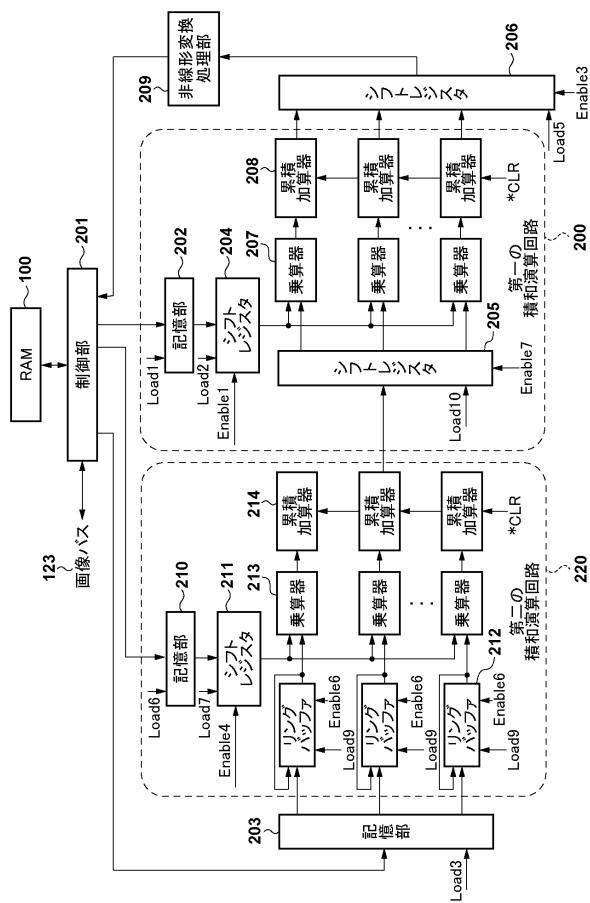
【図15】



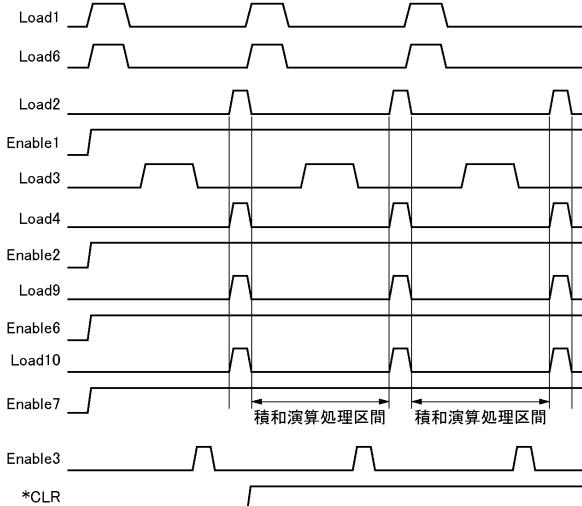
【図16】



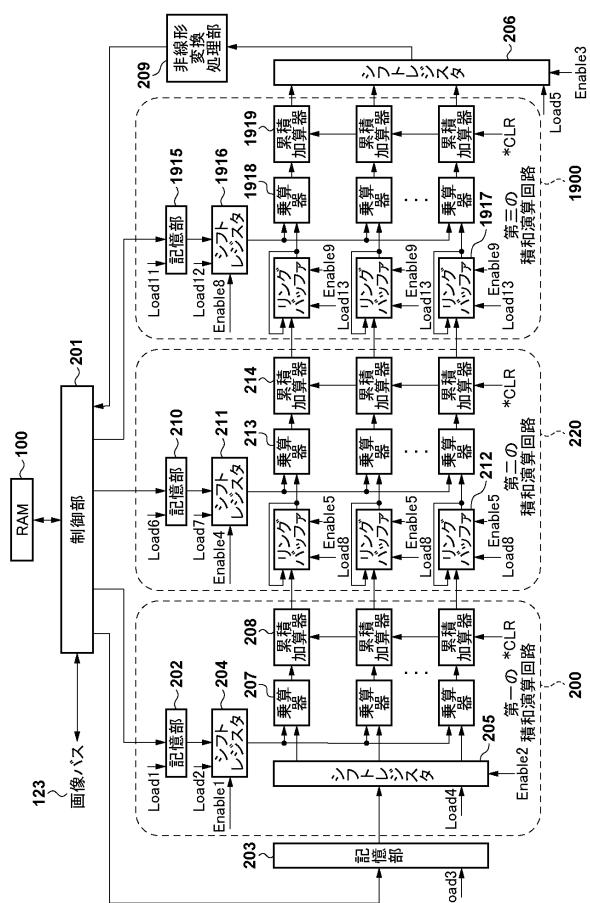
【図17】



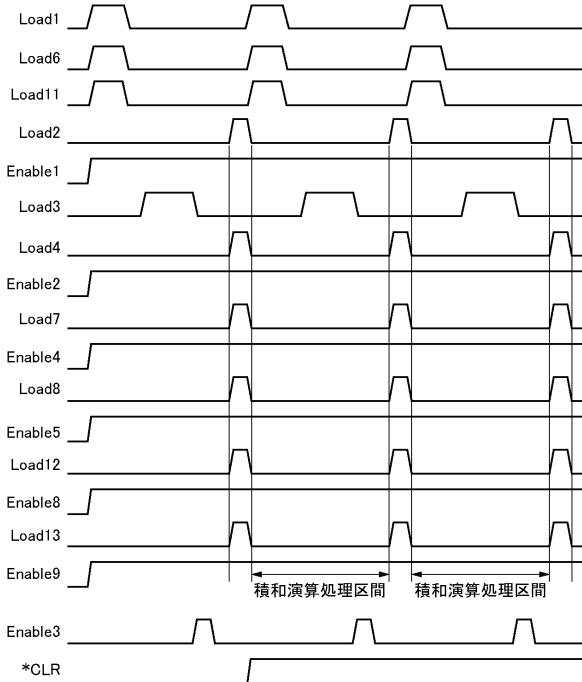
【図18】



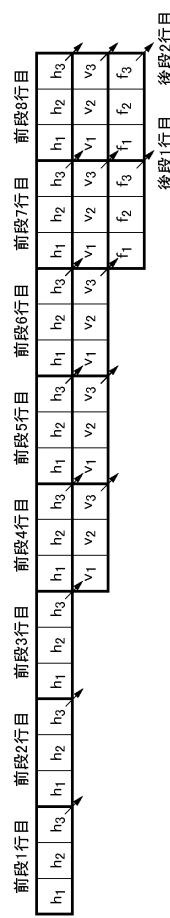
【図19】



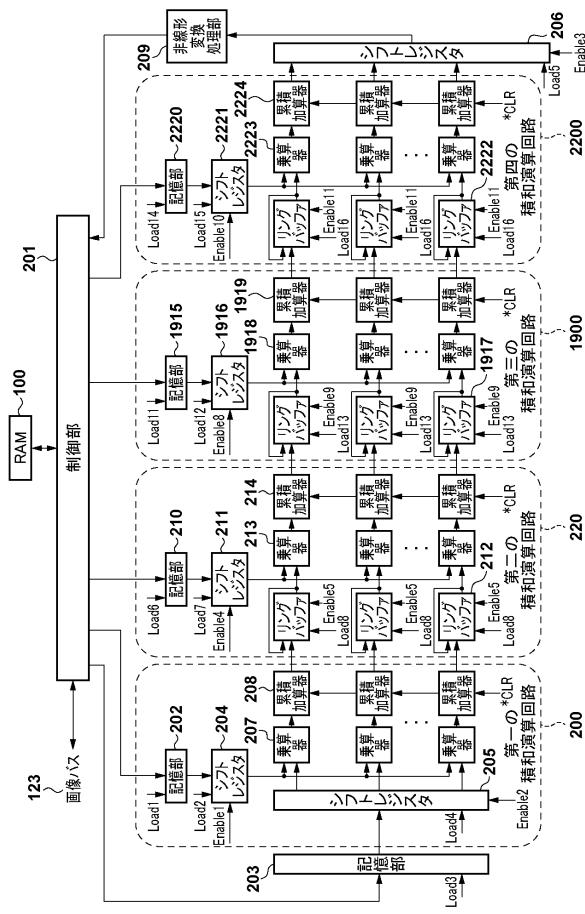
【図20】



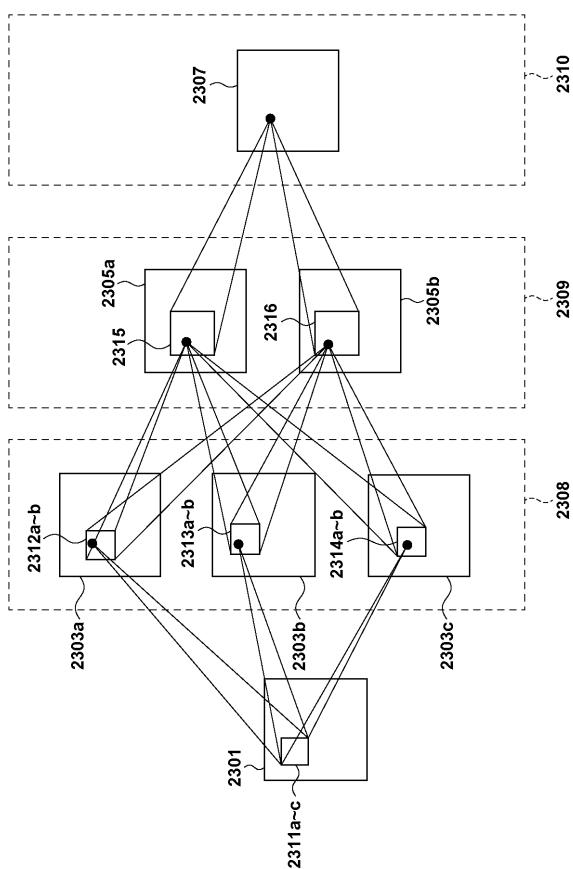
【図 2 1】



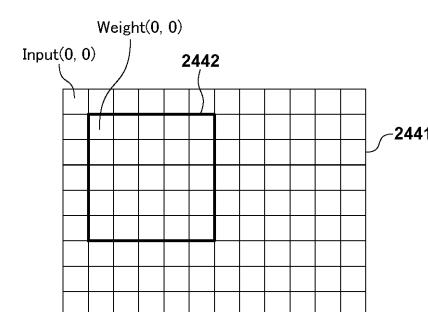
【図 2 2】



【図 2 3】



【図 2 4】



フロントページの続き

(72)発明者 野村 修
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(72)発明者 山本 貴久
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(72)発明者 加藤 政美
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(72)発明者 伊藤 嘉則
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(72)発明者 森 克彦
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

審査官 漆原 孝治

(56)参考文献 特開平09-212477(JP, A)
特開2010-134697(JP, A)

(58)調査した分野(Int.Cl., DB名)

G 06 F 17 / 10