

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6889510号
(P6889510)

(45) 発行日 令和3年6月18日(2021.6.18)

(24) 登録日 令和3年5月25日(2021.5.25)

(51) Int.Cl. F I
G06F 11/36 (2006.01) G O 6 F 11/36 1 6 4
G06F 8/34 (2018.01) G O 6 F 8/34

請求項の数 6 (全 18 頁)

<p>(21) 出願番号 特願2020-553674 (P2020-553674)</p> <p>(86) (22) 出願日 令和1年11月22日 (2019.11.22)</p> <p>(86) 国際出願番号 PCT/JP2019/045768</p> <p>(87) 国際公開番号 W02020/110933</p> <p>(87) 国際公開日 令和2年6月4日 (2020.6.4)</p> <p>審査請求日 令和2年10月1日 (2020.10.1)</p> <p>(31) 優先権主張番号 特願2018-222247 (P2018-222247)</p> <p>(32) 優先日 平成30年11月28日 (2018.11.28)</p> <p>(33) 優先権主張国・地域又は機関 日本国 (JP)</p> <p>早期審査対象出願</p>	<p>(73) 特許権者 506381751 株式会社トラスト・テクノロジー 東京都国立市中1-9-8 第7叶ビル9 F</p> <p>(74) 代理人 100166545 弁理士 折坂 茂樹</p> <p>(72) 発明者 山本 隆一郎 東京都国立市中1-9-8 第7叶ビル9 階 株式会社トラスト・テクノロジー内</p> <p>審査官 杉浦 孝光</p>
--	---

最終頁に続く

(54) 【発明の名称】 プログラミング装置、およびプログラム

(57) 【特許請求の範囲】

【請求項1】

ユーザ操作に応じて、編集画面上に、処理単位を示すユニットのアイコンを配置して、これらユニットを連結することによりプログラムを編集するプログラミング装置であって、

前記編集画面に配置され連結されたユニットにより定義されるプログラムを実行し、各ユニットの出力を演算するプログラム実行手段と、

前記プログラム実行手段が演算して得られた各ユニットの出力を、前記編集画面に配置されたそれぞれの前記ユニットのアイコンに対応付けたデバッグビュー領域に表示するデバッグビュー表示制御手段と、

前記プログラムを構成する各ユニットの設定値の変更を受け付けるユニット編集手段とを備え、

前記プログラミング装置は、前記ユニットとして、複数種類の画像フィルタを選択可能なフレキシブルイメージフィルタユニットを利用可能であり、

前記ユニット編集手段は、

選択可能な複数種類の画像フィルタのそれぞれを入力データに対し適用した場合のプレビュー画像を、選択されている画像フィルタが適用されたプレビュー画像を識別可能な態様で一覧表示する第1プレビュー領域と、

選択されている前記画像フィルタに関する特性パラメータについて、設定可能な複数の設定値を適用した場合のプレビュー画像を、設定されている設定値が適用されたプレビ

ユー画像を識別可能な態様で一覽表示する第2プレビュー領域と、
選択されている画像フィルタを、設定されている特性パラメータの設定値にて入力データに対し適用した場合のプレビュー画像であるターゲット画像を、前記第1プレビュー領域および前記第2プレビュー領域に表示される各プレビュー画像よりも大きく表示するターゲット画像表示領域と

を備える設定画面により、前記フレキシブルイメージフィルタユニットのユーザによる設定を受け付けることを特徴とするプログラミング装置。

【請求項2】

前記プログラムを構成するユニットは、前記プログラミング装置への入力、前記プログラミング装置からの出力、条件分岐、及びディープラーニングの学習済みデータによる推論を含む複数の選択枝から選択可能とされることを特徴とする請求項1に記載のプログラミング装置。

10

【請求項3】

前記デバッグビュー表示制御手段は、前記プログラム実行手段が前記プログラムを実行する間、各ユニットの出力の変化に応じてリアルタイムで前記デバッグビュー領域の表示を更新することを特徴とする請求項1または2に記載のプログラミング装置。

【請求項4】

前記デバッグビュー表示制御手段は、ディープラーニングの学習済みデータによる推論を行う前記ユニットについて、推論において特徴としてとらえている箇所を示すヒートマップをデバッグビュー領域に表示することを特徴とする請求項1から3のいずれか1項に記載のプログラミング装置。

20

【請求項5】

前記ユニット編集手段は、前記プログラム実行手段が前記プログラムを実行している間にユニットの設定値の変更を受け付けることが可能とされ、

前記プログラム実行手段は、前記プログラムの実行中に前記ユニット編集手段が受け付けた設定値の変更を、実行する前記プログラムでの演算に随時反映することを特徴とする請求項1から4のいずれか1項に記載のプログラミング装置。

【請求項6】

コンピュータを請求項1から5のいずれか1項に記載のプログラミング装置として機能させることを特徴とするプログラム。

30

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、画像処理等に用いるプログラムを作成・編集するのに適したプログラミング装置、およびプログラムに関する。

【背景技術】

【0002】

昨今、画像を撮影するカメラ等の撮影装置、画像を処理するコンピュータ等が容易に入手可能となり、様々な現場で画像処理や画像解析を組み込んだシステムが利用されている。工場の生産現場でも生産品の寸法検査や外観検査、異物検査などにカメラとコンピュータを用いた画像検査システムが普及してきている。このようなシステムを実現するためのソリューションとして、様々な現場で利用されることを想定して汎用の装置化を行なった画像検査システムやパッケージとして販売されるソフトウェアプログラムが提供されている。しかし、多くの場合、汎用のシステムやソフトウェアプログラムをそのまま使用したのでは現場で求められる細かな仕様に対応することが困難であるため、特別仕様に対応したカスタマイズをしたり、所望の仕様に合ったプログラムを独自に開発したりといった対応がなされている。細かな仕様とは、例えば画像検査の対象物に合わせた画像処理アルゴリズムの選定や変更、画像処理に必須となる閾値（パラメータ）の調整など多岐に渡る仕様の組み合わせである。

40

【0003】

50

このように現場の要求が多岐にわたることから、システムの開発者や現場の設備を運用するユーザ自身が簡単に画像処理アルゴリズムや閾値の設定および変更ができるよう、プログラムを開発したりパラメータの調整をしたりといったことを実現できるプログラミング環境が求められている。

【0004】

プログラミングに関する専門知識が高くないユーザ等であってもプログラミングができる手法として、ファンクションブロックダイアグラム(FBD)などのグラフィカルなプログラム言語がある。

【0005】

FBDでは、各種演算処理に対応する関数からなるファンクションブロック、外部機器との間で入力・出力パラメータをやり取りするための接続子、これらファンクションブロックや接続子を接続する接続線など、各種のコンポーネントを用いて、グラフィカルにプログラミングを行うことが可能である。

【0006】

FBDをプログラミングするプログラミング装置では、このようなコンポーネントを示すシンボルをオブジェクトとして編集画面上に配置し、これらオブジェクトに設けられているポート間を接続線で結線することで処理の手順を定義し、オブジェクトについて設定可能な変数を示すプロパティを変更することで各オブジェクトでの処理を調整するという編集作業を行う(例えば、特許文献1を参照)。

【先行技術文献】

【特許文献】

【0007】

【特許文献1】特開2008-204237公報

【発明の概要】

【発明が解決しようとする課題】

【0008】

しかしながら、従来技術の手法ではプログラミング作業自体は容易になったものの、プログラムの妥当性を確認するときにプログラムを構成する複数段の処理での中間的な処理結果プログラムを容易に確認することができず、デバッグ作業や変数の細かな調整に大きな手間がかかっていた。特に様々な画像処理を多段階に行うプログラムでは、1つの処理(オブジェクト)における変数の変更が後段での出力(最終出力も当然含む)に大きな影響を与え得るため、変数調整に何度も試行錯誤を要することがあり、プログラムの編集とプログラムの妥当性の確認との反復作業は大きな負担となっているという問題があった。

【0009】

本発明はこのような課題を解決するためのものであり、画像処理アルゴリズムに代表されるプログラムの作成や編集時のプログラムの妥当性を容易に確認することができるプログラミング装置、およびプログラムを提供することを目的とする。

【課題を解決するための手段】

【0010】

上記の課題を解決すべく、本発明に係るプログラミング装置は、ユーザ操作に応じて、編集画面上に、処理単位を示すユニットのアイコンを配置して、これらユニットを連結することによりプログラムを編集するプログラミング装置であって、編集画面に配置され連結されたユニットにより定義されるプログラムを実行し、各ユニットの出力を演算するプログラム実行手段と、プログラム実行手段が演算して得られた各ユニットの出力を、編集画面に配置されたそれぞれのユニットのアイコンに対応付けたデバッグビュー領域に表示するデバッグビュー表示制御手段とを備えることを特徴とする。プログラムを構成するユニットは、プログラミング装置への入力、プログラミング装置からの出力、条件分岐、及びディーブラーニングの学習済みデータによる推論を含む複数の選択枝から選択可能とされるとよい。

【0011】

本発明に係るプログラミング装置は、プログラムを構成する各ユニットの設定値の変更を受け付けるユニット編集手段をさらに備えるとよい。

【0012】

本発明では、デバッグビュー表示制御手段は、プログラム実行手段がプログラムを実行する間、各ユニットの出力の変化に応じてリアルタイムでデバッグビュー領域の表示を更新するとよい。また、デバッグビュー表示制御手段は、ディープラーニングの学習済みデータによる推論を行うユニットについて、推論において特徴としてとらえている箇所を示すヒートマップをデバッグビュー領域に表示するとよい。

【0013】

本発明では、ユニット編集手段は、プログラム実行手段がプログラムを実行している間にユニットの設定値の変更を受け付けることが可能とされるとよい。また、プログラム実行手段は、プログラムの実行中にユニット編集手段が受け付けた設定値の変更を、実行するプログラムでの演算に随時反映するとよい。

【0014】

また、本発明に係るプログラミング装置は、ユニットとして、複数種類の画像フィルタを選択可能なフレキシブルイメージフィルタユニットを利用可能とされるとよい。そして、ユニット編集手段は、選択可能な複数種類の画像フィルタのそれぞれを入力データに対し適用した場合のプレビュー画像を、選択されている画像フィルタが適用されたプレビュー画像を識別可能な態様で一覧表示する第1プレビュー領域と、選択されている画像フィルタに関する特性パラメータについて、設定可能な複数の設定値を適用した場合のプレビュー画像を、設定されている設定値が適用されたプレビュー画像を識別可能な態様で一覧表示する第2プレビュー領域と、選択されている画像フィルタを、設定されている特性パラメータの設定値にて入力データに対し適用した場合のプレビュー画像であるターゲット画像を、第1プレビュー領域および第2プレビュー領域に表示される各プレビュー画像よりも大きく表示するターゲット画像表示領域とを備える設定画面により、フレキシブルイメージフィルタのユーザによる設定を受け付けるとよい。

【0015】

本発明に係るプログラムは、コンピュータを上記いずれかのプログラミング装置として機能させることを特徴とする。

【図面の簡単な説明】

【0016】

【図1】プログラミング装置の構成を示すブロック図である。

【図2】ユニットのデータ構造の一例を示す図である。

【図3】編集画面の一例を示す図である。

【図4】出力データが複数あるユニットのアイコンの一例を示す図である。

【図5】フレキシブルイメージフィルタユニットの設定画面の一例を示している。

【図6】起動直後の編集画面の一例を示す図である。

【図7】ユニット選択画面の一例を示す図である。

【図8】ユニット設定画面の一例を示す図である。

【図9】ユニットが1つ追加された状態の編集画面の一例を示す図である。

【図10】ユニットが3つ追加された状態の編集画面の一例を示す図である。

【図11】3つのユニットが連結された状態の編集画面の一例を示す図である。

【図12】実行モードでの編集画面の一例を示す図である。

【図13】編集モードでの編集画面の一例を示す図である。

【図14】ユニットボードの他の例を示す図である。

【図15】ユニットボードのさらに他の例を示す図である。

【発明を実施するための形態】

【0017】

本発明の実施の形態について図面を参照して説明する。図1は、本実施形態に係るプログラミング装置の構成を示すブロック図である。

【 0 0 1 8 】

プログラミング装置 1 0 は、全体としてサーバ装置、パーソナルコンピュータ、携帯情報端末などの情報処理装置からなり、ユーザ操作に応じて、プログラムを構成する処理ステップである各種のユニットを編集画面上に配置して、これらユニット間を接続線で結線することにより、プログラミングを作成・編集する機能を有する。

【 0 0 1 9 】

プログラミング装置 1 0 において配置できるユニットとしては、画像処理だけでなく画像検査に必要な他の様々な要素（例えば、入力、出力、条件分岐、ディープラーニングの学習済みデータによる推論等）も含まれ、プログラミングにおいては全ての処理をユニットとして統一的に扱うことができる。入力ユニットとしては、カメラや外部プログラム、共有メモリ、TCP/IPによるネットワーク入力、画像ファイル、CSVファイル、I/Oボード等が含まれる。カメラのユニットとしては、標準的なUSBカメラ(UVCドライバ)のみでなく、各社の主要な産業用カメラに対応したユニットが利用可能とされる。出力ユニットとしては、ディスプレイ表示、画像ファイル、CSVファイル、共有メモリ、TCP/IPによるネットワーク通信、I/Oボード、パトランプ等が含まれる。画像処理のユニットとしては、各種のフィルタ等の画像処理、複数の入力に対する演算（画像の加算・減算等）、RGBによる画像の分離、画像からのHSV（H：色相、S：彩度、V：明度）情報の出力等が含まれる。その他のユニットとしては、条件分岐、ディープラーニングの学習済みデータによる推論といった判断を伴う処理も含まれる。これらのユニットについては、後述するように処理内容等が定義されたユニットプレート情報 1 4 A が予め用意される。プログラミング装置 1 0 のユーザは、これらのユニットを自由に配置・連結して、所望のプログラムを作成する。

【 0 0 2 0 】

図 1 に示すように、プログラミング装置 1 0 には、主な機能部として、通信 I / F 部 1 1、操作入力部 1 2、画面表示部 1 3、記憶部 1 4、および演算処理部 1 5 が設けられる。

【 0 0 2 1 】

通信 I / F 部 1 1 は、通信用のインタフェース回路からなり、通信回線を介して外部装置（図示せず）とデータ通信を行うことにより、各種データをやり取りする機能を有する。外部装置としては、例えば、カメラ、各種のセンサ類等のプログラミング装置に入力情報を提供する機器、パトランプ、ブザー等のプログラムにより制御される出力機器、サーバ装置、パーソナルコンピュータ、携帯情報端末等、プログラムの実行結果を受け取ってさらなる処理を行う情報処理装置等が挙げられる。

【 0 0 2 2 】

操作入力部 1 2 は、キーボード、マウス、タッチパネルなどの操作入力装置からなり、ユーザ操作を示す操作を検出して演算処理部 1 5 へ出力する機能を有する。

【 0 0 2 3 】

画面表示部 1 3 は、LCDなどの画面表示装置からなり、演算処理部 1 5 から出力された編集画面、設定画面などの各種データを画面表示する機能を有する。

【 0 0 2 4 】

記憶部 1 4 は、ハードディスクや半導体メモリなどの記憶装置からなり、演算処理部 1 5 での処理に用いる各種処理情報やプログラム 1 4 P を記憶する機能を有する。

【 0 0 2 5 】

プログラム 1 4 P は、演算処理部 1 5 の CPU で実行されることにより、各種の処理部を実現するプログラムであり、外部装置や記録媒体（共に図示せず）から読み出されて記憶部 1 4 に予め格納される。

【 0 0 2 6 】

記憶部 1 4 で記憶する主な処理情報として、ユニットプレート情報 1 4 A 及びユニットボード情報 1 4 B がある。ユニットプレート情報 1 4 A は、プログラムを構成する処理ステップである各種のユニットについて、各ユニットのデータ構造、変数、アルゴ

10

20

30

40

50

リズム等を定義した情報である。

【0027】

図2は、ユニットのデータ構造の一例を示している。ユニットIDは、ユニットを識別するためのユニークな識別情報であり、プログラミングの過程でユニットが配置される際に自動的に採番される。ユニットIDをユーザが編集することは禁止されるとよい。ユニット名は編集画面において表示されるユニットの名称であり、ユーザが自由に付することができる。ユニット種別は、ユニットの種類を示す情報である。このユニット種別は予め定義されユーザによる編集は禁止される。ユニットの種別としては、入力（カメラ、入力画像ファイル等）、出力（パトランプ、ディスプレイ表示、出力ファイル等）、画像処理（フィルタ、画像サイズ変換等）等が挙げられる。本実施形態のプログラミング装置10では、画像に対するフィルタ処理を行うユニットの一つとして、後述するフレキシブルイメージフィルタユニットを備える。

10

【0028】

ユニット座標は、編集画面内におけるユニットの表示位置を示す情報である。ユニット座標は、プログラミングの過程でユニットを配置する際にその初期値が決まり、その後ユーザが編集画面内でユニットを移動させることに応じて適宜更新される。

【0029】

前ユニットポインタ及び次ユニットポインタはユニットの接続関係を規定する情報である。前ユニットポインタは、当該ユニットの前段に連結されたユニットのデータへのポインタ（メモリアドレス）を格納する。また、次ユニットポインタは、当該ユニットの後段に連結されたユニットのデータへのポインタ（メモリアドレス）を格納する。プログラミングの過程で配置した複数のユニットを連結する操作により、前ユニットポインタ及び次ユニットポインタが設定される。

20

【0030】

ユニット設定画面は、プログラミングの過程でユニットを配置する際に表示されるユニット固有の設定画面を規定する情報である。ユニットの設定画面は予め定義され、ユーザによる編集は禁止される。例えば、カメラ入力ユニットの場合、カメラ番号（カメラの機種等を示す）を設定する画面が定義される。なお、ユニットの種類によらず共通の設定項目であるユニット表示名等を設定する画面については、ここに定義する必要はない。

【0031】

ユニット設定データは、ユニット固有の設定データの情報である。例えば、カメラ入力のユニットの場合には選択されたカメラを示すカメラ番号がユニット設定データに該当する。また、入力される画像データのサイズ変換をするリサイズユニットの場合には、リサイズ後の画像のサイズを規定する情報（ピクセル数、入力画像に対する拡大率等）がユニット設定データに該当する。

30

【0032】

ユニット入力データは、当該ユニットに対し前段のユニットから入力される画像やファイル等のデータを表す。カメラ入力ユニットのように、前段に他のユニットが接続されないユニットについては入力データが定義されなくてよい。

【0033】

ユニットロジックは、当該ユニットで実行される処理のアルゴリズムを規定する。ユニットのロジックは予め定義され、ユーザによる編集は禁止される。なお、ユーザには、設定データを変更することにより可能な範囲でユニットでの処理の調整が許容される。ユニットのロジックとしては、例えば、カメラ入力ユニットのように外部機器からの入力を得るユニットについては、外部機器を制御して必要なデータを得て後段に出力するためのアルゴリズムが規定される。また、パトランプユニットのように入力データに基づき外部機器を制御するユニットについては、外部機器を制御して所望の動作を行わせるためのアルゴリズムが規定される。また、フィルタユニットのように入力データを処理して出力データを出力するユニットについては、入力データに基づき出力データを出力するアルゴリズムが規定される。

40

50

【0034】

ユニット出力データは、当該ユニットから次の段のユニットに出力される画像やファイル等のデータを表す。パトランプや出力ファイルのユニットのように、後段に他のユニットが接続されないユニットについては出力データが定義されなくてよい。

【0035】

記憶部14は、ユニットテンプレート情報14Aとして、各種のユニットについて上記のようなデータ構造、およびユニットの種別、設定画面、ロジック等の予め定義される情報を格納する。

【0036】

ユニットボード情報14Bは、各種のユニットを組み合わせるユーザが作成したプログラムを表す情報である。後述する編集画面を用いて配置・連結された複数のユニットについての諸情報が、先述のユニットのデータ構造に従った形式でユニットボード情報14Bとして格納される。

10

【0037】

演算処理部15は、CPUとその周辺回路を有し、記憶部14のプログラム14Pを読み込んで実行することにより、プログラミング装置の各種処理を実現する。すなわち、演算処理部15は、本発明におけるプログラム実行手段、デバッグビュー表示制御手段、及びユニット編集手段として機能する。演算処理部15で実現される主な処理として、ユニットボードの編集処理が挙げられる。

【0038】

20

編集処理は、ユーザ操作に応じて編集画面上に各種のユニットを配置する機能、配置されたユニット間をユーザ操作に応じて連結する機能、および配置されたユニットの設定値をユーザ操作に応じて変更する機能等を含んでいる。

【0039】

図3は、編集処理を行う際に用いられる編集画面の一例を示している。図3に示されるように、編集画面は、ユニットの編集や制御に用いるコントロールボタンが配置されたコントロール領域R1と、ユーザがユニットを配置・接続してグラフィカルにプログラミングを行うための領域であるユニットボード編集領域R2とを備える。

【0040】

30

コントロール領域R1は、編集画面上部に設けられ、ユニットを配置、移動、実行等させるためのコントロールボタン、チェックボックス等のコントロールオブジェクトが複数配置される。これらのコントロールオブジェクトは、ユニットボード情報14Bの制御に関するもの、ユニットボード編集領域R2でのプログラミングに関するもの、ユニットボード編集領域R2にてプログラミングされたプログラムの実行制御に関するもの、ユニットボード編集領域R2の標示制御に関するものの4種類に大別される。

【0041】

ユニットボード情報14Bの制御に関するコントロールオブジェクトとしては、編集集中のユニットボードの名称表示T1、ユニットボード編集領域R2でプログラミングしたユニットボード(プログラム)を保存するための保存ボタンB1、保存済みのユニットボードを読み込んでユニットボード編集領域R2に表示させるための読込ボタンB2、ユニットボード編集領域R2で編集集中のユニットボードの編集を終了するための終了ボタンB3、後述するデバッグビューを有効にするか否かを選択するためのチェックボックスCB1等が含まれる。

40

【0042】

ユニットボード編集領域R2でのプログラミングに関するコントロールオブジェクトとしては、ユニット追加ボタンB4、ユニット編集ボタンB5、ユニット連結ボタンB6、ユニット移動ボタンB7、ユニット削除ボタンB8、及びユニット複製ボタンB9が含まれる。

【0043】

ユニット追加ボタンB4は、ユニットボード編集領域R2に新たなユニットを配置する

50

追加モードを開始するためのコントロールボタンである。このユニット追加ボタン B 4 を押下すると、配置するユニットを選択するためのユニット選択画面が表示される。ユニット選択画面で所望のユニットを選択し、画面表示に従いユニットの設定を行った後、ユニットボード編集領域 R 2 内の所望の位置をクリックすると、そのクリックした位置にユニットのアイコンが配置される。図 3 に示すように、ユニットのアイコンは、略矩形のオブジェクトであり、ユニット種別とユニット名称が表示される。

【 0 0 4 4 】

ユニット編集ボタン B 5 は、ユニットボード編集領域 R 2 に配置されているユニットの設定を編集する編集モードを開始するためのコントロールボタンである。このユニット編集ボタン B 5 を押下して、ユニットボード編集領域 R 2 内に配置されているユニットのアイコンをクリックすると、当該ユニットの設定値を編集するための設定画面が表示される。ユーザは設定画面にてユニットの各種設定値を編集することができる。

10

【 0 0 4 5 】

ユニット連結ボタン B 6 は、ユニットボード編集領域 R 2 に配置されているユニット間の接続関係を設定する連結モードを開始するためのコントロールボタンである。ここで、ユニットの連結とは、連結元のユニットの出力が連結先のユニットの入力となるようにデータの受け渡しを定義づけることである。

【 0 0 4 6 】

図 4 は、ユニットの一例として、画像からの H S V (H : 色相、 S : 彩度、 V : 明度) 情報を出力するユニットの表示例を示している。図 4 に例示されるように、各ユニットには、入力及び出力の数に応じたジョイント図形 J (例えば小さな四角形) が表示される。また、ジョイント図形 J には入出力の内容を示す文字等が表示される。ユニットのアイコンでは、左辺が入力端、右辺が出力端を意味するため、ユニットの左辺には入力の数に応じたジョイント図形 J が表示され、ユニットの右辺には出力の数に応じたジョイント図形 J が表示される。連結関係を可視化すべく、ユニットボード編集領域 R 2 において、連結されたユニット間には連結されていることを示す接続線 (ジョイント図形 J 同士を結ぶ線) が表示される。

20

【 0 0 4 7 】

ユニット連結ボタン B 6 を押下すると、連結モードが開始され、連結元のユニットの選択を受け付ける状態となる。所望のユニットのアイコン (複数の出力を有するユニットについてはいずれかの出力のジョイント図形 J) をクリックすると、そのユニットの出力 (複数ある場合にはクリックされたジョイント図形 J に対応するもの) が連結元として特定される。続いて連結先のユニットのアイコン (複数の入力を有するユニットについてはいずれかの入力のジョイント図形 J) をクリックすると、そのユニットに入力 (複数ある場合にはクリックされたジョイント図形 J に対応するもの) が連結先として特定され、2つのユニットが連結される。このような操作により、1組のユニットを連結することができる。ユニットボード編集領域 R 2 において、連結されていることを示すべく、連結されたユニットのジョイント図形 J 間を結ぶ線が表示される。ユニットのアイコンでは、左辺が入力端、右辺が出力端を意味する。したがって、第 1 のユニットの出力が第 2 のユニットに入力される場合、第 1 のユニットのアイコンの右辺と第 2 のユニットのアイコンの左辺とを結ぶ線が表示されることになる。

30

40

【 0 0 4 8 】

ユニット移動ボタン B 7 は、ユニットボード編集領域 R 2 に配置されているユニット間の接続関係を設定する移動モードを開始するためのコントロールボタンである。このユニット移動ボタン B 7 が押下された移動モードにおいて、ユーザはユニットボード編集領域 R 2 内に配置されているユニットをドラッグアンドドロップ操作により自由に移動させることができる。

【 0 0 4 9 】

ユニット削除ボタン B 8 は、ユニットボード編集領域 R 2 に配置されているユニットを削除する削除モードを開始するためのコントロールボタンである。このユニット削除ボタ

50

ンB 8 が押下された削除モードにおいて、ユニットボード編集領域 R 2 内に配置されているユニットをクリックすると、当該ユニットを削除することができる。なお、削除を実行する前に意思確認を要求する画面を表示し、削除の意思が確認できた場合にのみ削除を実行するように構成してもよい。

【 0 0 5 0 】

ユニット複製ボタン B 9 は、ユニットボード編集領域 R 2 に配置されているユニットを複製する複製モードを開始するためのコントロールボタンである。このユニット複製ボタン B 9 が押下された複製モードにおいて、ユニットボード編集領域 R 2 内に配置されているユニットをクリックすると、当該ユニットが複製対象として選択され、さらにユニットボード編集領域 R 2 内をクリックすると、そのクリックした位置に選択したユニットの複製（ユニット ID が異なり、未連結のもの）が追加される。

10

【 0 0 5 1 】

ユニットボード編集領域 R 2 にてプログラミングされたプログラムの実行制御に関するコントロールオブジェクトとしては、ユニット実行ボタン B 1 0、ステップ実行ボタン B 1 1、及びユニット停止ボタン B 1 2 が含まれる。

【 0 0 5 2 】

ユニット実行ボタン B 1 0 は、ユニットボード編集領域 R 2 で定義されたプログラムを実行する実行モードを開始するためのコントロールボタンである。このユニット実行ボタン B 1 0 が押下された実行モードでは、ユニットボード編集領域 R 2 に配置され連結されたユニット群により定義される一連のプログラムが、演算処理部 1 5 により最初から最後まで通して実行される。

20

【 0 0 5 3 】

ステップ実行ボタン群 B 1 1 は、ユニットボード編集領域 R 2 で定義されたプログラムを、段階を分けて順に実行するステップ実行モードを制御するコントロールボタン群である。ステップ実行ボタン群 B 1 1 は、主にプログラムの動作確認等のデバッグに用いられるものであり、ブレークポイント設定、ステップイン、ステップオーバー、ステップアウト等の制御を行うためのボタンが設けられる。

【 0 0 5 4 】

ユニット停止ボタン B 1 2 は、ユニット実行モード及びステップ実行モードを終了するためのコントロールボタンである。

30

【 0 0 5 5 】

ユニットボード編集領域 R 2 の標示制御に関するコントロールオブジェクトとしては、ユニットボード編集領域 R 2 での表示倍率を設定するためのテキストボックス T 2 及びスライダバー S B、及びユニットボード編集領域 R 2 内のユニットを透過表示するか否かを設定するためのチェックボックス C B 2 が含まれる。

【 0 0 5 6 】

ユニットボード編集領域 R 2 は、ユニットのアイコンやユニット間の結線等が表示される領域である。ユーザは、ユニットボード編集領域 R 2 に配置したユニットを、ドラッグアンドドロップ操作によって自由に移動させたり、ユニットをダブルクリックして設定画面を表示させ、設定値を調整したりすることができる。

40

【 0 0 5 7 】

また、ユニットボード編集領域 R 2 には、配置された個々のユニットでの処理結果（出力データ）を示すデバッグビュー D V を表示することができる。デバッグビュー D V は、例えば対応するユニットからの吹き出しのように表示されるとよい。デバッグビュー D V を表示するか否かは、ユニットボード編集領域 R 2 に設けられたチェックボックス C B 1 の入力状態（ON / OFF）によって定められる。チェックボックス C B 1 が ON であれば、デバッグビュー D V はユニット実行モード及びステップ実行モードだけでなく、追加モード、編集モード等のその他のモードでユニットボード編集領域 R 2 内のユニットを編集している際にもデバッグビュー D V が表示される。なお、所定の静止画像を出力するユニットのように、実行モードでなくても出力が確定するユニットでは、配置するだけでデ

50

デバッグビューＤＶにその出力が表示される。一方、カメラや画像処理など、処理を実行しなければ出力データが出力されないユニットでは、配置しただけではデバッグビューＤＶの枠が表示されるのみで出力データ（画像）は表示されず空欄であり、実行時にはじめて出力データが表示される。そして、実行中は、ユニットの出力の変化に応じてリアルタイムでデバッグビューＤＶの表示が更新される。

【 0 0 5 8 】

デバッグビューＤＶに表示するユニットの出力データは、ユニットの種類によってさまざまであってよい。カメラや画像処理のように画像を出力するユニットの場合には、デバッグビューＤＶに出力データとして画像を表示するとよい。また、ディープラーニングの学習済みデータによる推論のユニットの場合には、ユニットに対する入力に基づく推論によりクラス分類して、順位（該当する可能性の高い順にクラスを順位付けした数値）や各クラスの推定確率（該当する可能性）をデバッグビューＤＶに表示してもよい。あるいは、ディープラーニングの学習済みデータによる推論のユニットにおいて、ニューラルネットワークが特徴としてとらえている箇所を示すヒートマップをデバッグビューＤＶに表示してもよい。このヒートマップは、入力される画像に重畳して、半透過表示されるとよい。

10

【 0 0 5 9 】

このようなデバッグビューＤＶをユニットボード編集領域Ｒ２に配置されている各ユニットに対して表示させることにより、最終的な出力だけでなく各ユニットでの中間的な出力データをも一目で確認することができる。様々な画像処理を多段階に行うプログラムでは、各段階の処理（ユニット）を経ることによる処理結果の変遷を最終出力まで一覧できる形で確認できるので、所望の出力結果を得るための設定の試行錯誤を、デバッグビューＤＶを見ながら容易に行うことが可能となる。

20

【 0 0 6 0 】

なお、実行モードの間もユニット編集ボタンＢ５は有効とされ、実行モードの最中にユニット編集ボタンＢ５をクリックし、さらに編集対象のユニットのアイコンをクリックすると、当該ユニットの設定画面が表示される。そして、設定画面にて設定値を変更すると、その設定値の変更はリアルタイムで実行中のプログラムに反映される。その結果、デバッグビューＤＶの表示にも、設定値の変更の影響が即座に反映される。このような構成により、ユーザは、設定値を変更した影響を確認しながら設定値の調整を行うことができる。

30

【 0 0 6 1 】

続いて、以上のような操作画面を用いてユニットボード編集領域Ｒ２に配置されるユニットの１つとして本実施形態のプログラミング装置１０で利用可能とされるフレキシブルイメージフィルタユニットについて説明する。画像に対するフィルタとして、ハフ変換、二値化、反転、エッジ抽出、色抽出、彩度抽出等、様々な種類のアルゴリズムが従来から用いられている。フレキシブルイメージフィルタユニットでは、フィルタの種類（アルゴリズム）及びフィルタの特性を決める特性パラメータを設定値とするユニットである。フレキシブルイメージフィルタユニットは、画像に対するフィルタリングを行うユニットであるため、入力データ及び出力データはいずれも画像のデータである。

40

【 0 0 6 2 】

フレキシブルイメージフィルタユニットは、他のユニットと同様にユニットボード編集領域Ｒ２に追加し、連結することで使用できるが、その設定画面に特徴を有する。図５はフレキシブルイメージフィルタユニットの設定を受け付ける設定画面の一例を示している。図５に示されるように、フレキシブルイメージフィルタユニットの設定画面は、ターゲット画像表示領域Ｒ３、アルゴリズムプレビュー領域Ｒ４（本発明の第１プレビュー領域に相当）、パラメータプレビュー領域Ｒ５（本発明の第２プレビュー領域に相当）を備えている。ターゲット画像表示領域Ｒ３は画面の中央部を含む画面内の主要領域を占める。ターゲット画像表示領域Ｒ３には、その時点で設定されているフィルタ種類及び特性パラメータのフィルタを入力データに対し適用した場合のプレビュー画像であるターゲット画

50

像が表示される。ターゲット画像表示領域 R 3 に表示されるターゲット画像は、アルゴリズムプレビュー領域 R 4 やパラメータプレビュー領域 R 5 に表示されるプレビュー画像より大きく表示されるとよい。各領域 (R 3 ~ R 5) に表示されるプレビュー画像は、入力に対して予め各条件のフィルタを適用したものを閲覧して選択できるように構成してもよいし、選択される毎に逐次処理して更新するように構成してもよい。

【 0 0 6 3 】

アルゴリズムプレビュー領域 R 4 は、ターゲット画像表示領域 R 3 の左側に設けられた領域である。アルゴリズムプレビュー領域 R 4 には、選択可能なアルゴリズムのフィルタ (特性パラメータは各フィルタの所定の初期値とするとよい) を入力データに対し適用した場合のプレビュー画像が、縦に並べて表示される。複数のプレビュー画像のうち、その時点で選択されているフィルタ種類に対応するものは、太枠で囲む等により識別可能とされる。図 5 の例では二値化のフィルタが選択されている。

10

【 0 0 6 4 】

フレキシブルイメージフィルタユニットの設定画面において、フィルタの種類を選択する操作は、アルゴリズムプレビュー領域 R 4 の表示と対応した操作とするとよい。すなわち、キーボードで「上」又は「下」キーを押下する操作、マウスでスクロールホイールを回転する操作、マウスでアルゴリズムプレビュー領域 R 4 内の所望のフィルタに対応するプレビュー画像をクリックする操作等によりフィルタの種類を選択できるように構成するとよい。これらの直感的な操作に加え、リストボックス等によりフィルタ種類の選択を受け付けられるようにしてもよい。

20

【 0 0 6 5 】

パラメータプレビュー領域 R 5 は、ターゲット画像表示領域 R 3 の上側に設けられた領域である。パラメータプレビュー領域 R 5 には、その時点で選択されているアルゴリズムのフィルタにおいて特性パラメータを異ならせた複数のプレビュー画像が、横に並べて表示される。複数のプレビュー画像のうち、その時点で設定されている特性パラメータに対応するものは、太枠で囲む等により識別可能とされる。プレビュー画像は、設定可能な最小値から最大値までを所定のステップに分割した特性パラメータの値で生成し、特性パラメータの値の昇順又は降順に並べて表示するとよい。図 5 の例では、選択されている二値化のフィルタにおいて、二値化の閾値を異ならせた場合のプレビュー画像が表示されている。

30

【 0 0 6 6 】

フレキシブルイメージフィルタユニットの設定画面において、特性パラメータを設定する操作は、パラメータプレビュー領域 R 5 の表示と対応した操作とするとよい。すなわち、キーボードで「右」又は「左」キーを押下する操作、マウスでパラメータプレビュー領域 R 5 内の所望の特性パラメータに対応するプレビュー画像をクリックする操作、マウスホイールを左右にスイッチする操作、特定のキー (例えば、Shift キーや Ctrl キー) を押しながらマウスの左右ボタンをクリックする操作等により特性パラメータを設定できるように構成するとよい。これらの直感的な操作に加え、特性パラメータの値を入力する入力エリアを設けて直接入力を受け付けられるようにしてもよい。

【 0 0 6 7 】

フレキシブルイメージフィルタユニットは、ユニットを追加した段階では具体的な画像処理アルゴリズムは決定されていない。フレキシブルイメージフィルタユニットを含むユニットボード (プログラム) を試験的に実行しつつ、アルゴリズムの選択と各アルゴリズムの閾値の設定を 2 次元的に俯瞰して行うことで、試行錯誤の時間を節約して素早く適切なフィルタ条件を設定することができる。

40

【 0 0 6 8 】

以上で説明した、フレキシブルイメージフィルタユニットでは、アルゴリズムプレビュー領域 R 4 に表示される縦方向の一覧で各アルゴリズムの処理結果を確認しながら切り替えることができる。また、選択中のアルゴリズムの閾値毎の処理結果をパラメータプレビュー領域 R 5 に表示される横方向の一覧でパラメータ変更の影響を俯瞰的に確認しながら

50

、パラメータを変更することができる。その結果、従来の手法ではひとつひとつ設定と確認に時間のかかっていたフィルタ選択や特性パラメータ設定の試行錯誤を、設定変更の影響を確認しながら設定画面内で効率的に直感的な操作で行うことができる。

【 0 0 6 9 】

[本実施の形態の動作]

続いて、上記のように構成されるプログラミング装置 1 0 を用いたプログラミングの典型的な流れを説明する。

【 0 0 7 0 】

プログラミング装置 1 0 においてプログラム 1 4 P を起動すると、図 6 に示すように編集画面が表示される。この起動直後の編集画面では、ユニットボード編集領域 R 2 は空欄となっている。

10

【 0 0 7 1 】

続いて、ユニット追加ボタン B 4 をクリックして追加モードとすると、図 7 に示すようにユニット選択画面が表示される。このユニット選択画面では、ユニットテンプレート情報 1 4 A に登録されている各種のユニットのうちいずれかを選択する。各ユニットは、ユニット種別により分類されている（データ構造におけるユニット種別）。そこで、ユニット選択画面では、リストメニュー L 1 から所望のユニット種別を選択し、さらにユニット表示領域 R 6 に一覧表示される当該ユニット種別に属するユニットの中からいずれかを選ぶことにより、追加するユニットを特定する。ここでは、入力（Input）の種別に属する U S B カメラのユニットを選択したものとす。また、ユニット選択画面では、ユニット名称をテキストボックス T 3 で適宜編集することで設定する。ここではユニット名称として「USB Camera (UVC Universal Video Class)_4」という名称が設定されたものとする。

20

【 0 0 7 2 】

ユニットの選択及びユニット名の設定が済んだ後、OK ボタンをクリックすると、ユニット選択画面が消え、ユニット設定画面が表示される。このユニット設定画面は、ユニットのデータ構造において説明したように、ユニット固有の設定画面である。ここで追加しようとしている U S B カメラのユニット設定画面の例を図 8 (a) に示す。この設定画面では、カメラ番号を設定する。図 8 (b) は、ユニット設定画面の他の例であり、画像サイズを変更するリサイズユニットのユニット設定画面を示している。これらのような追加しようとするユニットの固有のユニット設定画面に対して所望の設定を行った後、OK ボタンをクリックするとユニット設定画面の標示が消え、入力待ちの状態となる。そこで、ユニットボード編集領域 R 2 内におけるユニットのアイコンを配置したい位置をクリックすると、当該クリックした位置に、図 9 に示すようにユニット選択画面で選択したユニットのアイコン U 1 が追加されるとともに、追加モードが終了となる。アイコン U 1 の脇にはチェックボックス C B 2 の設定に応じてデバッグビュー D V 1 が表示される。同様の操作を繰り返して、必要な数・種類のユニットをユニットボード編集領域 R 2 に追加する。

30

【 0 0 7 3 】

ユニット編集ボタン B 5 を押して編集モードにすれば、一旦追加したユニットについて、ユニットのアイコンをクリックすることでユニット設定画面を立ち上げ、ユニットの各種パラメータ等を変更することができる。また、ユニット削除ボタン B 8 を用いたユニットの削除、ユニット複製ボタン B 9 を用いたユニットの複製等も行うことができる。

40

【 0 0 7 4 】

ユニットボード編集領域 R 2 で編集したユニットボード（ユニットの接続により定義されるプログラム）は、保存ボタン B 1 をクリックすることにより表示される画面にて、ユニットボード名、保存場所を指定することに保存しておくことができる。また、読みボタン B 2 をクリックすることにより表示される画面にて、過去に作成したユニットボードを選択し、当該選択したユニットボードを読み込んでユニットボード編集領域 R 2 に表示させることもできる。

【 0 0 7 5 】

図 1 0 は、U S B カメラ、二値化、及び B L O B 検出の 3 つのユニット（U 1 ~ U 3 ）

50

を追加した状態の画面例を示している。図10の状態では、各ユニットは連結されておらず、入出力データの受け渡しが行われない状態である。そこで、ユニット連結ボタンB6をクリックして連結モードとし、ユニット間の接続を設定する。連結モードが開始されると、まず連結元のユニットの選択を受け付ける状態となる。所望のユニットの出力のジョイント図形をクリックするとそのユニットが連結元として特定される。続いて連結先のユニットのアイコンの入力のジョイント図形をクリックすると、2つのユニットが連結されるとともに、連結モードは終了となる。以上の一連の操作により、1組のユニットを連結することができる。

【0076】

図11は、ユニットボード編集領域R2に追加した3つのユニット(U1~U3)を、USBカメラ、二値化、BLOB検出の順に接続した状態を示している。この状態でユニット実行ボタンB10をクリックすると実行モードとなり、プログラムが実行される。すなわち、USBカメラでの撮像が開始され、撮像した画像に対するグレースケール変換、及びリサイズが行われる。実行モードにおいてデバッグビューDVが有効にされていると、図12に示すように、各ユニット(U1~U3)に対応するデバッグビュー(DV1~DV3)に各ユニットでの出力がリアルタイムで表示される。実行モードは、ユニット停止ボタンB12がクリックされるまで継続される。

【0077】

プログラミング装置10では、実行モードが継続中に各ユニットの設定値を調整することが許容される。すなわち、実行モード中にもユニット編集ボタンB5は有効とされ、これをクリックした後、設定値を調整したいユニットをクリックすると、当該ユニットの設定画面が表示される。表示された設定画面で設定値を変更すると、変更がリアルタイムでデバッグビューに反映される。例えば図13に示した例では、デバッグビュー(DV1~DV3)で各段階の画像をリアルタイムで確認しながら二値化のユニットの設定画面で設定値(閾値)を調整し、画像中の検出対象物が白い塊として画像に現れるよう調整している。このような構成により、ユーザは様々なアルゴリズムの積み重ねの集大成である画像処理プログラムにおける個々レイヤー(ユニット)の調整を、その影響をリアルタイムで確認しながら効率的に行うことができる。

【0078】

図14は、プログラミング装置10を用いてプログラミングしたユニットボードの他の例を示している。具体的には、図14は、複数の入力を有するユニットや複数の出力を有するユニットを用いて、より複雑な処理を実現したユニットボードを示している。

【0079】

すなわち、図14に示したユニットボードは、入力ユニットとしてUSBカメラのユニットU4と静止画のユニットU5を備え、減算ユニットU6によりUSBカメラで撮影した画像から静止画を減算することにより背景を除いた着目被写体の画像を抽出する処理を実現している。なお、減算ユニットは、2つの入力(A入力とB入力)を有し、A入力に入力される画像からB入力に入力される画像を減算してえられる画像を出力するユニットである。

【0080】

ユニットボード編集領域R2にユニット追加し連結する手法は図10を参照して説明した手法と同様である。ユニットを連結する際に、減算ユニットのいずれの入力に連結するかは、ジョイント図形JA及びJBのいずれをクリックするかにより設定される。

【0081】

図14に示した例のように、前段に他のユニットが接続されないユニットが複数ある場合、これらの複数のユニットから処理が開始されるものとしてプログラムが実現される。このような処理の開始点となるユニットとして、前段に他のユニットが接続されないユニットが自動的に設定されるように構成してもよいし、開始点となるユニットをユーザが明示的に設定するように構成してもよい。開始点となるユニットをユーザが明示的に設定するように構成する場合、編集画面に開始点となるユニットを設定するためのインター

10

20

30

40

50

スを設定するとともに、ユニットのデータ構造には、図 2 に示した項目に加え、開始点であることを示す情報を含めるとよい。

【 0 0 8 2 】

図 1 5 は、プログラミング装置 1 0 を用いてプログラミングしたユニットボードのさらに他の例を示している。図 1 5 (a) および (b) は、それぞれディープラーニングの学習済みデータによる推論のユニットにおけるデバッグビュー D V の表示例を示している。

【 0 0 8 3 】

図 1 5 (a) に示した例では、ディープラーニングの学習済みデータによる推論のユニットであるユニット U 8 が、ユニット U 7 から与えられる検査用画像（本例では正常と異常の形成品が混在した餃子の静止画）を入力画像として、外観検査用にクラス分類する。ユニット U 8 のデバッグビュー D V 8 には、順位と推定確率を表示に基づく推論によりクラス分類して、順位（該当する可能性の高い順にクラスを順位付けした数値）や各クラスの推定確率（該当する可能性）がリスト表示される。このようなデバッグビューの表示によれば、ユーザはディープラーニングの推論の結果を、最も該当可能性が高いクラスだけでなく、該当可能性が低い（順位が下位の）クラスも含め、その該当可能性を確認しながらアルゴリズムを構成する各ユニットの調整を行うことができる。更に、その推論結果は後段のユニットに連結することによって、さらなる画像処理や判定結果の利用が可能となる。

10

【 0 0 8 4 】

図 1 5 (b) に示した例では、ディープラーニングの学習済みデータによる推論のユニット U 1 0 が、ユニット U 9 から与えられる餃子の静止画を入力画像として、ニューラルネットワークが特徴としてとらえている箇所を示すヒートマップをデバッグビュー D V 1 0 に表示する。このヒートマップは、入力される画像に重畳して、半透過表示される。なお、図面はモノクロとなっているが、実際にはカラーで特徴量の増減を表現するとよい。具体的には、特徴のある箇所が特徴量に応じてカラーで青から赤方向にグラデーション状に表示されるとよい。このようなデバッグビューの表示によれば、ユーザはディープラーニングの推論において特徴（本例では開いた餃子の皮と、はみ出した肉の不良品としての特徴）として捉えられている箇所を視覚的に確認することができ、ディープラーニングの学習が目的通りに達成したかをリアルタイムに素早く確認し、アルゴリズムを構成する各ユニットの調整を行うことができる。また、ヒートマップの表示結果で学習がうまくいっていないと判断できる場合は、デバッグビューの結果をもとに速やかに再学習を行うことができる。

20

30

【 0 0 8 5 】

このように、プログラミング装置 1 0 によれば、画像の加算や減算といった複数の入力を伴う処理や、条件分岐、分離といった複数の出力を伴う処理を含む複雑なプログラムを、ユニットを配置して連結するというグラフィカルなプログラミング手法により、容易に作成することが可能となる。

【 0 0 8 6 】

以上で説明したようにプログラミング装置 1 0 によれば、ユニットボード編集領域 R 2 に配置された各ユニットについて、デバッグビューにて出力を一目で、且つ、リアルタイムで確認することができる。このため、プログラムの妥当性の確認や、変数調整の試行錯誤を効率的に行うことができる。

40

【 0 0 8 7 】

[実施の形態の拡張]

以上、実施形態を参照して本発明を説明したが、本発明は上記実施形態に限定されるものではない。本発明の構成や詳細には、本発明のスコープ内で当業者が理解しうる様々な変更をすることができる。

【 符号の説明 】

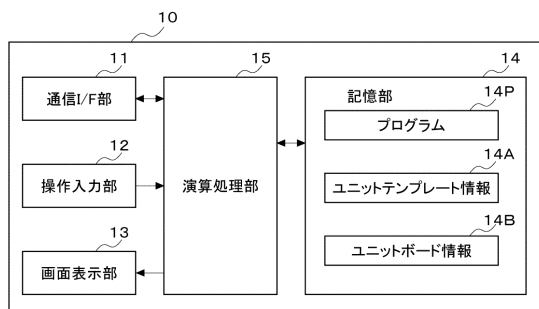
【 0 0 8 8 】

1 0 ... プログラミング装置、 1 1 ... 通信 I / F 部、 1 2 ... 操作入力部、 1 3 ... 画面表示

50

部、14...記憶部、15...演算処理部

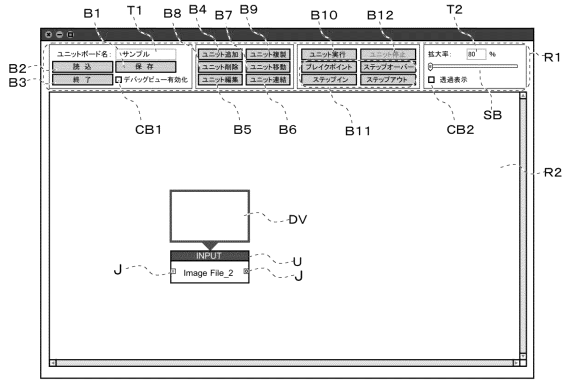
【図1】



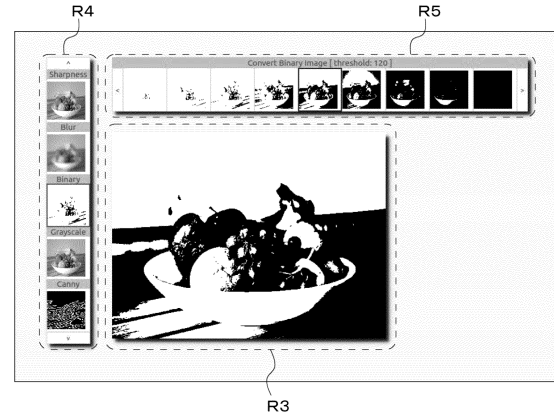
【図2】

ユニットID
ユニット種別
ユニット座標
前ユニットポインタ
次ユニットポインタ
ユニット設定画面
ユニット設定データ
ユニット入力データ
ユニットロジック
ユニット出力データ

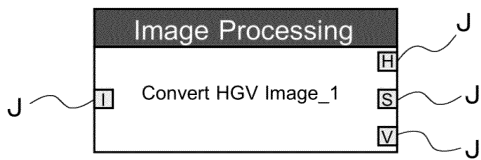
【図3】



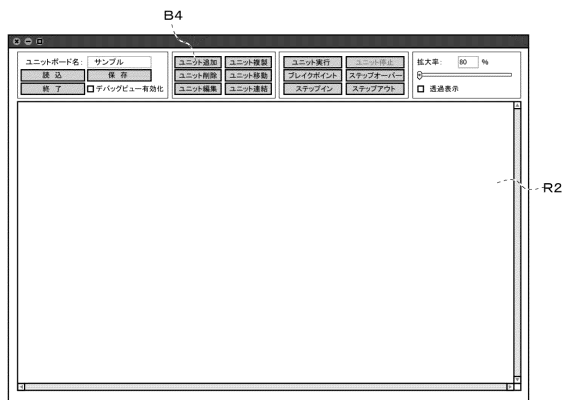
【図5】



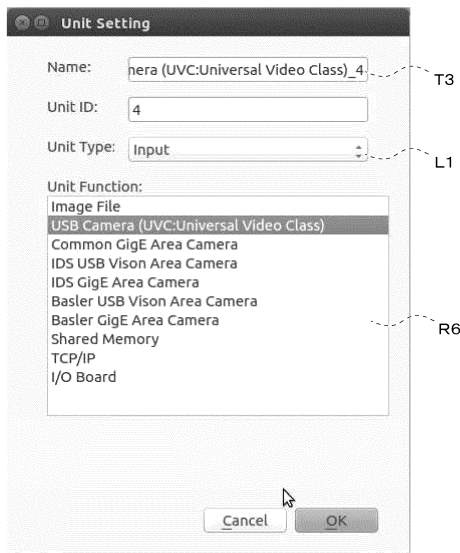
【図4】



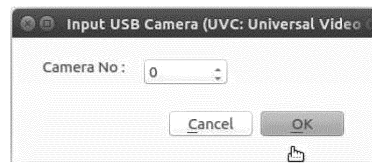
【図6】



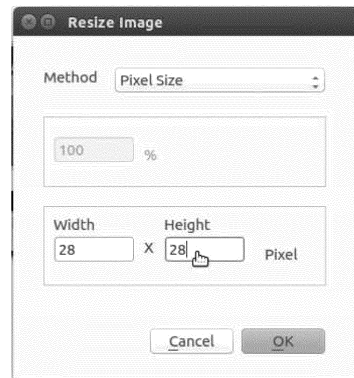
【図7】



【図8】

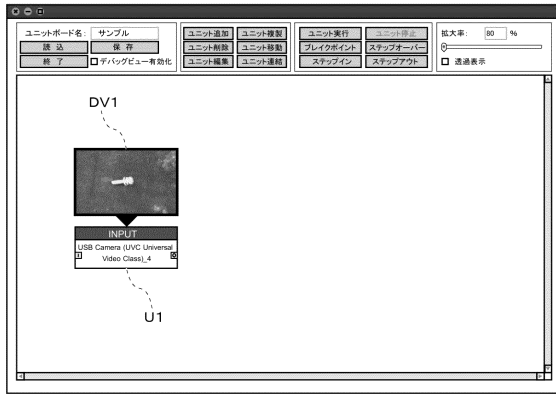


(a)

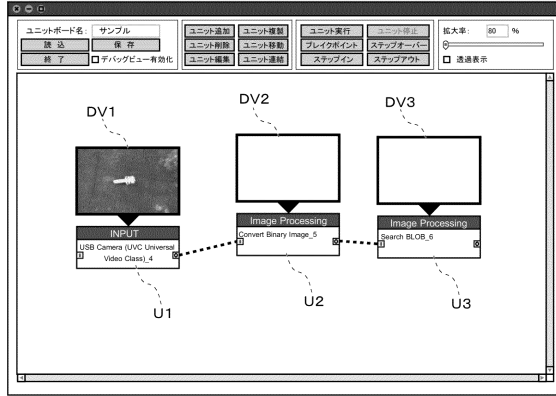


(b)

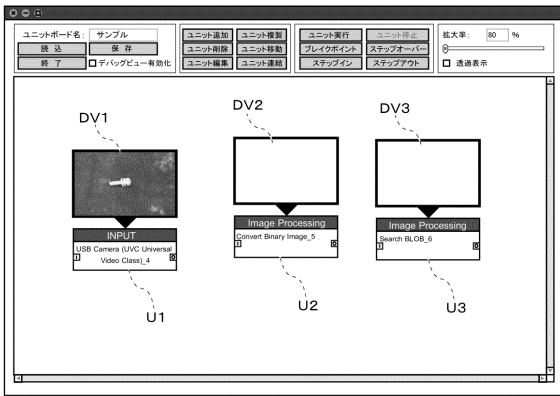
【図 9】



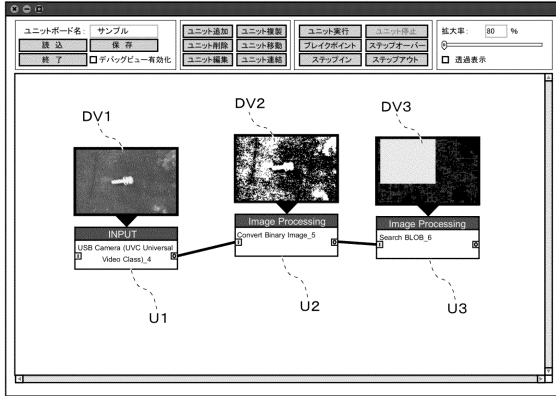
【図 11】



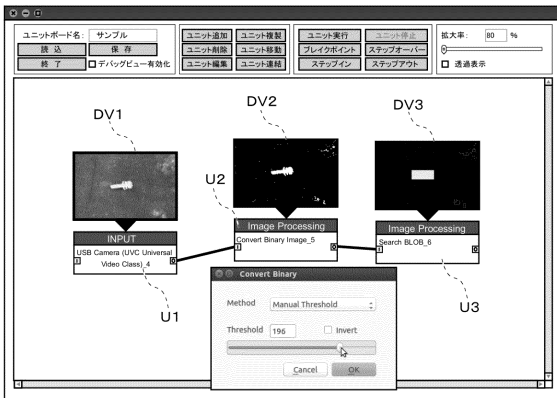
【図 10】



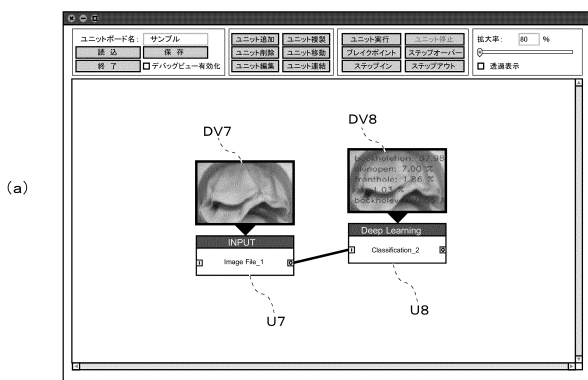
【図 12】



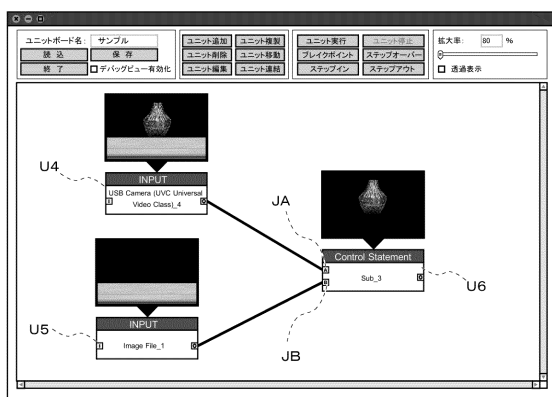
【図 13】



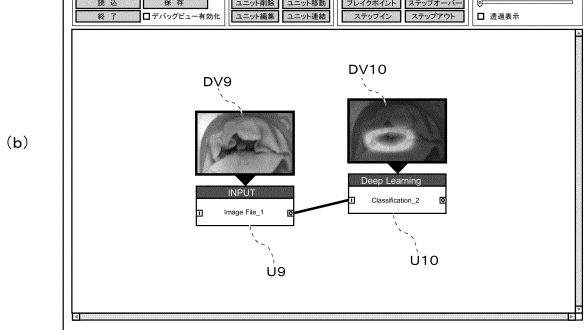
【図 15】



【図 14】



【図 15】



フロントページの続き

- (56)参考文献 特開2011-118467(JP,A)
特開2013-140572(JP,A)
国際公開第2018/061605(WO,A1)
特開平04-365179(JP,A)
特開2003-067766(JP,A)
特表2015-519625(JP,A)
安田国裕, My人工知能の育て方, Interface, 日本, CQ出版株式会社, 2018年10月, Vol.44, No.12, pp.20-37

(58)調査した分野(Int.Cl., DB名)

G06F 8/00 - 8/77
G06F 11/28 - 11/36
G06T 1/00