

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2021-501382

(P2021-501382A)

(43) 公表日 令和3年1月14日(2021.1.14)

(51) Int.Cl.		F I			テーマコード (参考)
G06F 8/38	8/38	(2018.01)	G06F 8/38		5B376
G06T 7/00	7/00	(2017.01)	G06T 7/00	350B	5L096
G06T 7/70	7/70	(2017.01)	G06T 7/70	A	

審査請求 有 予備審査請求 未請求 (全 67 頁)

(21) 出願番号 特願2019-570927 (P2019-570927)
 (86) (22) 出願日 平成30年7月6日 (2018.7.6)
 (85) 翻訳文提出日 令和2年2月28日 (2020.2.28)
 (86) 国際出願番号 PCT/US2018/041038
 (87) 国際公開番号 W02019/156706
 (87) 国際公開日 令和1年8月15日 (2019.8.15)
 (31) 優先権主張番号 201841005130
 (32) 優先日 平成30年2月12日 (2018.2.12)
 (33) 優先権主張国・地域又は機関
 インド (IN)
 (31) 優先権主張番号 15/996,311
 (32) 優先日 平成30年6月1日 (2018.6.1)
 (33) 優先権主張国・地域又は機関
 米国 (US)

(71) 出願人 502303739
 オラクル・インターナショナル・コーポレーション
 アメリカ合衆国カリフォルニア州94065
 レッドウッド・シティー, オラクル・パークウェイ500
 (74) 代理人 110001195
 特許業務法人深見特許事務所
 (72) 発明者 クマール, サティシュ・エス・ビー
 インド、560076 バンガロール、
 イ・ティ・エム・レイアウト・セカンド・
 ステージ、サード・ディ・クロス、18・
 メイン、ナンバー・41

最終頁に続く

(54) 【発明の名称】 自動コード生成

(57) 【要約】

テキスト情報と1つ以上のグラフィックユーザーインターフェイスコンポーネントを含むGUI画面画像からのGUIの開発を自動化するための技術が開示される。GUI画面画像を解析することにより、テキスト情報を抽出し、GUI画面に含まれるUIコンポーネントを特定する。GUI画面画像内の1つ以上のテキスト領域を検出しブレースホルダに置き換える。GUI画面内の1つ以上のグラフィックユーザーインターフェイスコンポーネントをGUI画面画像から抽出し機械学習ベースの分類器を用いて分類する。分類結果、1つ以上のテキスト領域の位置、および1つ以上のグラフィックユーザーインターフェイスコンポーネントの位置に基づいて、GUIのGUIモデルを生成する。生成したモデルを用いて、場合によっては異なるプログラミング言語の各種プラットフォームのために、GUIの1つ以上のインプリメンテーション(たとえば実行可能なコード)を生成することができる。

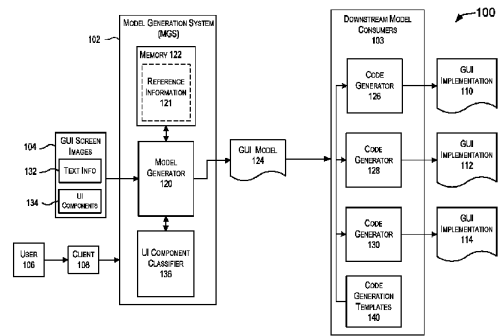


FIG. 1

【特許請求の範囲】**【請求項 1】**

方法であって、

コンピュータシステムが、入力画像から、グラフィックユーザインターフェイス（GUI）の GUI 画面を示す GUI 画面画像を検出するステップと、

前記コンピュータシステムが、第 1 のテキストコンテンツアイテムを含む前記 GUI 画面画像の第 1 の領域を検出するステップと、

前記コンピュータシステムが、前記 GUI 画面画像内の前記第 1 の領域の位置を判断するステップと、

前記コンピュータシステムが、前記第 1 の領域のコンテンツを、テキストコンテンツを含まないプレースホルダに置き換えるステップと、

前記コンピュータシステムが、前記 GUI 画面画像の第 2 の領域内に位置する第 1 のユーザインターフェイス（UI）コンポーネントを検出するステップとを含み、前記第 2 の領域は前記プレースホルダを有する前記第 1 の領域を含み、

前記コンピュータシステムが、前記 GUI 画面画像内の前記第 1 の UI コンポーネントの位置を判断するステップと、

前記コンピュータシステムが、機械学習ベースの分類器を用いて、前記第 1 の UI コンポーネントの UI コンポーネントタイプを判断するステップとを含み、前記機械学習ベースの分類器は、複数の訓練画像を含む訓練データを用いて訓練され、前記複数の訓練画像のうちの各訓練画像は UI コンポーネントを含み、前記訓練データはさらに、各訓練画像ごとに、前記訓練画像内の前記 UI コンポーネントの UI コンポーネントタイプを特定する情報を含み、

前記コンピュータシステムが、前記 GUI を実現するためのコードを生成するために使用可能な GUI モデルを生成するステップを含み、前記 GUI モデルは、前記第 1 の UI コンポーネントの情報と、前記第 1 のテキストコンテンツアイテムの情報とを含み、前記第 1 の UI コンポーネントの情報は、前記第 1 の UI コンポーネントについて判断した前記 UI コンポーネントタイプと、前記 GUI 画面画像内の前記第 1 の UI コンポーネントの位置とを示す情報を含む、方法。

【請求項 2】

前記第 1 の UI コンポーネントを検出するステップは、前記機械学習ベースの分類器が前記 GUI 画面画像内の前記第 1 の UI コンポーネントを検出するステップを含み、

前記訓練データはさらに、各訓練画像ごとに、前記訓練画像内の UI コンポーネントの位置を含む、請求項 1 に記載の方法。

【請求項 3】

前記 GUI 画面画像内の前記第 1 の領域の位置および前記 GUI 画面画像内の前記第 1 の UI コンポーネントの位置に基づいて、前記第 1 のテキストコンテンツアイテムとともに前記第 1 の UI コンポーネントをグルーピングするステップをさらに含み、

前記 GUI モデルを生成するステップはさらに、前記 GUI モデルに、前記グルーピングを示す情報を含めるステップを含む、請求項 1 または 2 に記載の方法。

【請求項 4】

前記コンピュータシステムが、第 2 のテキストコンテンツアイテムを含む前記 GUI 画面画像の第 3 の領域を検出するステップと、

前記コンピュータシステムが、前記 GUI 画面画像内の前記第 3 の領域の位置を判断するステップと、

前記コンピュータシステムが、前記 GUI 画面画像内に位置する第 2 の UI コンポーネントを検出するステップと、

前記 GUI 画面画像内の前記第 2 の UI コンポーネントの位置を判断するステップと、

前記機械学習ベースの分類器を用いて、前記第 2 の UI コンポーネントの UI コンポーネントタイプを判断するステップと、

前記第 1 の領域の前記位置、前記第 3 の領域の前記位置、前記第 1 の UI コンポーネン

10

20

30

40

50

トの前記UIコンポーネントタイプおよび位置、ならびに前記第2のUIコンポーネントの前記UIコンポーネントタイプおよび位置に基づいて、前記第1のテキストコンテンツアイテム、前記第2のテキストコンテンツアイテム、前記第1のUIコンポーネント、および前記第2のUIコンポーネントをグルーピングするステップと、

前記グルーピングに基づいて前記GUI画面のレイアウトを決定するステップとを含み、

前記GUIモデルを生成するステップはさらに、前記GUIモデルに、前記グルーピングおよび前記GUI画面の前記レイアウトを示す情報を含めるステップを含む、請求項1～3のいずれか1項に記載の方法。

【請求項5】

前記GUI画面画像内の前記第3の領域の前記位置に基づいて、前記第2のテキストコンテンツは前記GUI画面画像内のいずれのUIコンポーネントにも対応付けられていないと判断するステップと、

前記第2のテキストコンテンツアイテムはアクションを示すと判断するステップとをさらに含み、

前記GUIモデルを生成するステップは、前記GUIモデルにおいて、前記GUI画面画像の前記第3の領域内の前記第2のテキストコンテンツアイテムはクリックブルテキストであることを示すステップを含み、前記第2のテキストコンテンツのクリックは前記アクションを開始させる、請求項4に記載の方法。

【請求項6】

前記第1の領域は背景上の前記第1のテキストコンテンツアイテムを含み、前記方法はさらに、

前記GUI画面画像の前記第1の領域内の画素の強度値に基づいてヒストグラムを生成するステップと、

前記GUI画面画像の前記第1の領域内の、前記第1のテキストコンテンツアイテムに対応する画素の数と前記背景に対応する画素の数とに基づいて、前記ヒストグラムから、前記第1のテキストコンテンツアイテムの画素の強度値は前記背景の画素の強度値よりも高いと判断するステップと、

前記第1の領域内の画素の強度値を反転させるステップとを含み、前記反転により、前記第1のテキストコンテンツアイテムの画素の強度値は、前記第1の領域内の前記背景の画素の強度値よりも低くなり、

前記反転の実施後に前記第1の領域内の前記第1のテキストコンテンツアイテムを認識するステップを含む、請求項1～5のいずれか1項に記載の方法。

【請求項7】

前記ヒストグラムを生成する前に、前記第1の領域内の各画素の強度値に基づいて、前記GUI画面画像の前記第1の領域を、RGBサブ画像から2値サブ画像に変換するステップをさらに含む、請求項6に記載の方法。

【請求項8】

前記プレースホルダは画素の予め定められたパターンを含み、

前記第1のUIコンポーネントのUIコンポーネントタイプを判断するステップは、前記第2の領域内の前記プレースホルダの存在と前記プレースホルダ内の前記画素の予め定められたパターンとに基づいて、前記GUI画面画像の前記第2の領域を分類するステップを含む、請求項1～7のいずれか1項に記載の方法。

【請求項9】

前記プレースホルダは、前記第1の領域内の前記第1のテキストコンテンツアイテムの位置の境界を前記GUI画面画像から判断することを可能にする画素の予め定められたパターンである、請求項1～8のいずれか1項に記載の方法。

【請求項10】

前記画素の予め定められたパターンの画素値は、予め定められた値、または前記第1の領域の背景内の画素の画素値を含む、請求項9に記載の方法。

10

20

30

40

50

【請求項 1 1】

前記 G U I モデルに基づいて前記 G U I の 1 つ以上のインプリメンテーションを生成するステップをさらに含む、請求項 1 ~ 1 0 のいずれか 1 項に記載の方法。

【請求項 1 2】

前記 G U I モデルに基づいて前記 G U I の 1 つ以上のインプリメンテーションを生成するステップは、

前記 G U I モデルを用いて、第 1 のプラットフォームのための前記 G U I の第 1 のインプリメンテーションを生成するステップと、

前記 G U I モデルを用いて、第 2 のプラットフォームのための前記 G U I の第 2 のインプリメンテーションを生成するステップとを含み、前記第 2 のプラットフォームは前記第 1 のプラットフォームと異なる、請求項 1 1 に記載の方法。

10

【請求項 1 3】

前記 G U I モデルに基づいて前記 G U I の 1 つ以上のインプリメンテーションを生成するステップは、

前記 G U I モデルを用いて、第 1 のプログラミング言語で前記 G U I の第 1 のインプリメンテーションを生成するステップと、

前記 G U I モデルを用いて、第 2 のプログラミング言語で前記 G U I の第 2 のインプリメンテーションを生成するステップとを含み、前記第 2 のプログラミング言語は前記第 1 のプログラミング言語と異なる、請求項 1 1 または 1 2 に記載の方法。

【請求項 1 4】

20

前記 G U I モデルに基づいて前記 G U I の 1 つ以上のインプリメンテーションを生成するステップは、

前記 G U I モデルと 1 つ以上のコード生成テンプレートとを用いて前記 G U I の 1 つ以上のインプリメンテーションを生成するステップを含み、各コード生成テンプレートはプラットフォームまたはプログラミング言語に対応付けられている、請求項 1 1 に記載の方法。

【請求項 1 5】

前記 G U I モデルを生成するステップは、前記 G U I モデルの情報を、 J a v a S c r i p t オブジェクト表記 (J S O N) フォーマットで格納するステップを含む、請求項 1 ~ 1 4 のいずれか 1 項に記載の方法。

30

【請求項 1 6】

前記機械学習ベースの分類器は、画像ヒストグラムベースの非線形サポートベクターマシン分類器、または、人工ニューラルネットワークベースの分類器を含む、請求項 1 ~ 1 5 のいずれか 1 項に記載の方法。

【請求項 1 7】

前記 G U I モデルに関するユーザフィードバックを受けるステップをさらに含み、前記ユーザフィードバックは、

前記 G U I モデルにおいて前記第 1 の U I コンポーネントについて指定された前記 U I コンポーネントタイプの代わりに前記第 1 の U I コンポーネントに対応付けられる新たな U I コンポーネントタイプを特定する情報、または、

40

G U I 画面内に存在するが前記 G U I モデルに含まれない新たな U I コンポーネントを特定する情報、および前記新たな U I コンポーネントの U I コンポーネントタイプを示す情報を含み、

前記ユーザフィードバックに基づいて前記機械学習ベースの分類器を再訓練するステップをさらに含む、請求項 1 ~ 1 6 のいずれか 1 項に記載の方法。

【請求項 1 8】

前記ユーザフィードバックはさらに、前記第 1 の U I コンポーネントまたは前記新たな U I コンポーネントのサブ画像を含み、前記方法はさらに、

前記複数の訓練画像から特徴を抽出するステップと、

前記複数の訓練画像から抽出した前記特徴を、多次元空間内のデータポイントにマッピ

50

ングするステップとを含み、前記データポイントは、前記多次元空間内の一組のクラスタを形成し、

前記第1のUIコンポーネントまたは前記新たなUIコンポーネントの前記サブ画像から特徴を抽出するステップと、

前記第1のUIコンポーネントまたは前記新たなUIコンポーネントの前記サブ画像から抽出した特徴を、前記多次元空間内のデータポイントにマッピングするステップと、

前記第1のUIコンポーネントまたは前記新たなUIコンポーネントの前記サブ画像に対応する前記データポイントと、前記一組のクラスタの各クラスタの中心との間の距離を判断するステップと、

前記距離がしきい値未満であるという判断に応じて、前記第1のUIコンポーネントまたは前記新たなUIコンポーネントの前記サブ画像を前記訓練データに含めるステップとを含む、請求項17に記載の方法。

10

【請求項19】

1つ以上のプロセッサによる実行が可能な複数の命令を格納する非一時的なコンピュータ読取可能媒体であって、前記複数の命令は前記1つ以上のプロセッサによって実行されると前記1つ以上のプロセッサに以下を含む処理を実行させ、前記処理は、

入力画像から、グラフィックユーザインターフェイス(GUI)のGUI画面を示すGUI画面画像を検出することと、

第1のテキストコンテンツアイテムを含む前記GUI画面画像の第1の領域を検出することと、

20

前記GUI画面画像内の前記第1の領域の位置を判断することと、

前記第1の領域のコンテンツを、テキストコンテンツを含まないブレースホルダに置き換えることと、

前記GUI画面画像の第2の領域内に位置する第1のユーザインターフェイス(UI)コンポーネントを検出することとを含み、前記第2の領域は前記ブレースホルダを有する前記第1の領域を含み、

前記GUI画面画像内の前記第1のUIコンポーネントの位置を判断することと、

機械学習ベースの分類器を用いて、前記第1のUIコンポーネントのUIコンポーネントタイプを判断することとを含み、前記機械学習ベースの分類器は、複数の訓練画像を含む訓練データを用いて訓練され、前記複数の訓練画像のうちの各訓練画像はUIコンポーネントを含み、前記訓練データはさらに、各訓練画像ごとに、前記訓練画像内の前記UIコンポーネントのUIコンポーネントタイプを特定する情報を含み、

30

前記GUIを実現するためのコードを生成するために使用可能なGUIモデルを生成することを含み、前記GUIモデルは、前記第1のUIコンポーネントの情報と、前記第1のテキストコンテンツアイテムの情報とを含み、前記第1のUIコンポーネントの情報は、前記第1のUIコンポーネントについて判断した前記UIコンポーネントタイプと、前記GUI画面画像内の前記第1のUIコンポーネントの位置とを示す情報を含む、非一時的なコンピュータ読取可能媒体。

【請求項20】

システムであって、

40

1つ以上のプロセッサと、

前記1つ以上のプロセッサに結合されたメモリとを備え、前記メモリは、前記1つ以上のプロセッサによって実行されると前記システムに、

入力画像から、グラフィックユーザインターフェイス(GUI)のGUI画面を示すGUI画面画像を検出することと、

第1のテキストコンテンツアイテムを含む前記GUI画面画像の第1の領域を検出することと、

前記GUI画面画像内の前記第1の領域の位置を判断することと、

前記第1の領域のコンテンツを、テキストコンテンツを含まないブレースホルダに置き換えることと、

50

前記 G U I 画面画像の第 2 の領域内に位置する第 1 のユーザインターフェイス (U I) コンポーネントを検出することとを実行させ、前記第 2 の領域は前記ブレースホルダを有する前記第 1 の領域を含み、

前記 G U I 画面画像内の前記第 1 の U I コンポーネントの位置を判断することと、

機械学習ベースの分類器を用いて、前記第 1 の U I コンポーネントの U I コンポーネントタイプを判断することとを実行させ、前記機械学習ベースの分類器は、複数の訓練画像を含む訓練データを用いて訓練され、前記複数の訓練画像のうちの各訓練画像は U I コンポーネントを含み、前記訓練データはさらに、各訓練画像ごとに、前記訓練画像内の前記 U I コンポーネントの U I コンポーネントタイプを特定する情報を含み、

前記 G U I を実現するためのコードを生成するために使用可能な G U I モデルを生成することとを実行させ、前記 G U I モデルは、前記第 1 の U I コンポーネントの情報と、前記第 1 のテキストコンテンツアイテムの情報とを含み、前記第 1 の U I コンポーネントの情報は、前記第 1 の U I コンポーネントについて判断した前記 U I コンポーネントタイプと、前記 G U I 画面画像内の前記第 1 の U I コンポーネントの位置とを示す情報を含む、システム。

10

【発明の詳細な説明】

【技術分野】

【 0 0 0 1 】

関連出願の相互参照

本願は、2018年2月12日に出願され「AUTOMATED CODE GENERATION」と題されたインド仮特許出願第201841005130号に基づく優先権の利益を主張する2018年6月1日に出願され「AUTOMATED CODE GENERATION」と題された米国非仮特許出願第15/996,311号に基づく利益および優先権を主張し、あらゆる目的のために上記特許出願の内容全体を本明細書に引用により援用する。

20

【背景技術】

【 0 0 0 2 】

背景

典型的なグラフィックユーザインターフェイス (graphic user interface) (G U I) 開発プロセスにおいて、G U I は、設計者が、顧客またはクライアント調査、マーケティング調査、および、開発する G U I に含まれるべき機能および外観を伝える他の情報源に基づいて設計することができる。G U I は、アプリケーションのための各種画面のモックアップ画像、画面の設計およびルック・アンド・フィール (look and feel)、画面と画面との間の移行などのような、アプリケーションの所望のユーザインターフェイス (U I) を記述することができる。画面のモックアップ画像に加えて、G U I は、G U I に関する情報をユーザに提供するテキストコンテンツも含み得る。

30

【 0 0 0 3 】

アプリケーションのための G U I (G U I 画面の画像を含む) は、設計者が文書 (たとえば設計文書) または仕様書 (たとえば画像ファイルもしくは概略スケッチ) において文書化することができる。次に、この G U I 設計文書を用いて、アプリケーションのための G U I を実現するためのコードを作成または開発することができる。たとえば、開発段階において、アプリケーションのための G U I 画面の 1 つ以上の画像またはスケッチを含む G U I 設計文書は、G U I 設計文書に基づいて G U I および / またはアプリケーションを実現するためのコードを記述するタスクが課された技術者または開発者を含む工学技術組織に提供することができる。これらの開発者は、手作業でこのコードを記述する、または「ドラッグ & ドロップ」ベースの開発ツールを用いて所望の G U I 画面を構築し G U I 設計文書に記載されている所望の外観および機能を有する G U I を実現するコードを生成することができる。

40

【 0 0 0 4 】

このように、実行可能なアプリケーションのための G U I を設計文書から開発することは、相当な開発者の手作業の手間を要する場合がある。これは、開発者が、設計文書を検

50

討し、G U I画面の所望の機能および外観を含む要件を理解し、それからコードを記述してG U Iおよび/またはアプリケーションを実現することを必要とする場合がある。これはまた、開発者が、G U Iの開発に使用されるアプリケーション開発プラットフォームおよびプログラミング言語に関する知識を有すること、ならびに、G U Iおよびアプリケーションの開発対象である特定のターゲットシステムまたはプラットフォーム（たとえばi O S（登録商標）またはA n d r o i d（登録商標）、モバイルまたはデスクトップ）に関する知識を有することも、必要とする場合がある。このため、G U Iの開発には特定の専門技術を有する開発者が必要であることが極めて多い。これらの要因はすべて、開発段階を、冗長で、時間がかかり、大きな労働力を要し、高コストなものにする。

【発明の概要】

【課題を解決するための手段】

【0005】

簡単な概要

本開示は、アプリケーション開発に関し、より具体的には、アプリケーションのためのグラフィックユーザインターフェイス（G U I）を、当該アプリケーションの1つ以上のG U I画面用の1つ以上の画像またはスケッチ等の設計文書から開発することを自動化する技術に関する。方法、システム、1つ以上のプロセッサによる実行が可能なプログラム、コードまたは命令を格納する非一時的なコンピュータ読取可能記憶媒体などを含む、発明の各種実施形態が、本明細書に記載される。

【0006】

特定の実施形態において、アプリケーションのためのG U Iは、1つ以上のG U I画面を含み得る。各画面は、ボタン、テキスト入力ボックス、ドロップダウンリスト、ドロップダウンメニュー、アイコン、表などのような1つ以上のユーザインターフェイス（U I）コンポーネントを含む。G U I画面はまた、U Iコンポーネントに加えてテキストコンテンツを含む部分を含み得る。テキストコンテンツは、特定のU Iコンポーネントに対応付けることができ、たとえば、U Iコンポーネントに入力される情報、U Iコンポーネントの機能、U R Lリンクその他を記述することができる。

【0007】

特定の実施形態において、たとえばG U I設計者が設計したG U I画面の画像（G U I画面画像とも呼ぶ）を解析することにより、G U I画面画像からテキスト情報を抽出するとともに、G U I画面に含まれるU Iコンポーネントを特定する。各種の機械学習ベースの技術を用いることによりG U I画面画像を解析することができる。たとえば、U Iコンポーネントを、機械学習ベースの分類器（たとえばサポートベクターマシン分類器または畳み込みニューラルネットワークベースの分類器）によって検出および分類することができ、この分類器は、入力G U I画面画像を提供し、画面画像に存在するU Iコンポーネントおよび画像内の検出したU Iコンポーネントの位置を特定するように構成されている。加えて、機械学習ベースの分類器は、検出された各U Iコンポーネントのタイプおよび/または検出された各U Iコンポーネントに対応付けられた機能を特定するように構成されてもよい。G U I画面画像内のテキストコンテンツアイテムおよび対応するそれぞれの位置を検出し認識してもよい。

【0008】

次に、たとえば、検出したU Iコンポーネント、U Iコンポーネントのタイプ、U Iコンポーネントの位置、U Iコンポーネントの対応付けられたテキスト情報、および、どのU Iコンポーネントにも対応付けられていない可能性があるその他のテキスト情報に基づいて、アプリケーションのためのG U Iについて、G U Iモデルを生成することができる。G U Iモデルは、言語から独立しかつプラットフォームから独立している。アプリケーションのためのG U Iを記述する情報は、アプリケーションのための1つ以上のG U I画面を記述する情報と、各G U I画面ごとの、G U I画面上の、検出したU Iコンポーネント、テキスト部分その他に関する情報とを含み得る。

【0009】

10

20

30

40

50

アプリケーションのために生成されたモデルを格納するために各種異なるフォーマットを使用することができる。たとえば、特定の実施形態において、生成したモデルは、JavaScript（登録商標）オブジェクト表記（JavaScript Object Notation）（JSON）フォーマットのような、言語および/またはプラットフォームから独立したデータ交換フォーマットで記述することができる。いくつかの実装例において、GUIモデルは、アプリケーションに対応付けることができるメタデータとして生成してもよい。

【0010】

続いて、アプリケーションのためのGUIの解析に基づいて生成されたGUIモデルを用いて、このモデルに記述されているアプリケーションのためのGUIを実現するコードを生成することができる。生成されるこのコードは、実行または解釈されると、アプリケーションGUIを記述する設計文書（たとえばGUIを表す一組の画像を含む設計文書）に示されるルック・アンド・フィールおよび機能を有するGUIを生成するものである。生成された同じモデルを、各種の異なるプラットフォーム（たとえばiOS（登録商標）、アンドロイド（登録商標）など）のための潜在的に異なる言語でコードを生成するために使用してもよい。

10

【0011】

先に述べたように、GUI画面を表す画像を解析することにより、この画像内に存在する1つ以上のテキストコンテンツアイテムおよびUIコンポーネントを検出することができる。いくつかの実施形態において、UIコンポーネントを分類する前に、テキストコンテンツアイテムを、UIコンポーネントの画像から抽出してもよく、テキストコンテンツを含まないプレースホルダに置き換えてもよい。いくつかの実施形態において、テキストコンテンツアイテムを含む画像の部分の画素に基づいてヒストグラムを生成することができる。ヒストグラムを用いて、テキストコンテンツアイテムの強度が背景の強度よりも低いか否かを判断することができる。次に、画像のこの部分の画素を反転させることにより、この画像の部分のテキストコンテンツアイテムを、この画像の部分の背景よりも暗くすることができる。次に、反転させた画像の部分に基づいて、テキストコンテンツアイテムを認識することができる。いくつかの実施形態において、この画像の部分のテキストコンテンツアイテムを、画素の予め定められたパターンを有する、またはこの画像の部分の背景の画素値を有する画素アレイに置き換えてもよい。この画素アレイの置き換えにより、実際のおテキストコンテンツが画像部分から削除されるだけでなく、画像内のテキストコンテンツの位置および協会が識別される。いくつかの実施形態において、画素の予め定められたパターンにより、テキストコンテンツアイテムの位置および/またはUIコンポーネントのタイプを判断し易くすることができる。

20

30

【0012】

特定の実施形態において、アプリケーションのためのGUIについて生成されたGUIモデルは、このアプリケーションのための1つ以上のGUI画面に対応する情報をカプセル化することができる。各GUI画面ごとに、GUIモデルは、GUI画面に含まれる1つ以上のユーザインターフェイス（UI）コンポーネントを特定する情報を含み得る。各GUI画面ごとに、このモデルはまた、ユーザインターフェイスコンポーネントの階層的組織を特定する情報およびGUI画面上のテキストコンテンツアイテム等の、GUI画面の構造に関する情報を含み得る。たとえば、いくつかの実施形態において、UIコンポーネントを、たとえば、UIコンポーネントのタイプおよび位置に基づいてグルーピングすることにより、UIコンポーネントのサブグループ（たとえば表またはリスト）を形成することができる。サブグループをさらにクラスタ化することによって、GUI画面のより高いレベルのレイアウトを決定することができる。いくつかの実施形態において、テキストコンテンツアイテムもグルーピングすることにより、たとえば、テキストのラインまたはテキストの段落を形成することができる。いくつかの実施形態において、たとえば、テキストコンテンツアイテムおよびUIコンポーネントの位置情報に基づいて、テキストコンテンツアイテムをグルーピングするかまたはUIコンポーネントに対応付けることができる。

40

50

【0013】

特定の実施形態において、GUI画面の画像を解析することにより、テキストコンテンツアイテムおよびUIコンポーネントを含む画像の部分を特定することができる。テキストコンテンツアイテムの位置および性質に応じて、テキストコンテンツアイテムは画面上で検出されたUIコンポーネントに対応付けられていると判断することができる、または、特定の実施形態において、テキストコンテンツアイテムは独立しており検出されたどのUIコンポーネントにも対応付けられていないと判断することができる。生成されたGUIモデルはこの情報をカプセル化する。いくつかの実施形態において、GUIモデルはこのようにして、特定のテキストコンテンツアイテムは独立しておりどのUIコンポーネントにも対応付けられていないことを示すことができる。特定の実施形態において、対応付けられていないテキストコンテンツアイテムを検出し、クリックされるとアクションを開始できるクリッカブルテキスト（またはリンク）に分類することができる。この情報はGUIモデルに格納することができる。

10

【0014】

いくつかの実施形態において、GUIモデルは、設計されたGUI画面のルック・アンド・フィールを再現できるようにGUI画面の各種UIコンポーネントを如何にして表示すべきかを記述することができる。特定の実施形態において、GUIモデルはまた、GUI画面上の特定のUIコンポーネントに対応付けられる機能に関する情報を含み得る。ユーザーインターフェイスコンポーネントに対応付けられる機能は、たとえば、機械学習ベースの分類器によって分類されたユーザーインターフェイスコンポーネントのタイプ、および/または対応付けられたテキストコンテンツアイテム（もしあれば）に基づいて、判断することができる。

20

【0015】

いくつかの実施形態において、生成したGUIモデルに関するフィードバックをユーザーが編集および提供できるようにするインフラストラクチャが提供される。ユーザーフィードバックは、たとえば、自動的に生成されたモデルの部分に対して行われる訂正（たとえばUIコンポーネントに対応付けられたタイプの変更）を含み得る。フィードバックはまた、ユーザーがモデルに追加される追加情報を提供することを含み得る。次に、ユーザーフィードバックは、機械学習ベースの分類器を改善する（たとえば再訓練する）ために使用することができる。

30

【0016】

GUI設計情報に基づいてGUIについて生成されたGUIモデルは、さまざまなダウンストリームコンシューマによって使用されることができる。たとえば、ダウンストリームコンシューマは、このモデルを使用することにより、GUIを実現するためのコードを自動でかつ実質的に手作業によるコーディングなしで、生成することができる。このコードは、たとえば、GUIモデルを生成するために使用された一組の画像に示されている所望のルック・アンド・フィールおよび/または機能と実質的に同様のルック・アンド・フィールおよび/または機能を有するGUIを表示するために、ウェブブラウザによって解釈されることが可能な解釈可能プログラムまたは1つ以上のプロセッサによって実行されることが可能な実行可能プログラムであってもよい。同じGUIモデルを異なるコンシューマが使用してもよい。たとえば、第1のコンシューマが第1のプラットフォーム（たとえばiOS（登録商標））に対して実行可能なものを自動的に生成するためにこのGUIモデルを使用してもよく、第2のコンシューマが異なるプラットフォーム（たとえばAndroid（登録商標））に対して第2の実行可能なものを自動的に生成するためにこの同じGUIモデルを使用してもよい。また、GUIモデル（たとえばJSONフォーマット）を用いて、マークアップ言語（たとえばHTMLもしくはXML）またはスタイルシート言語（たとえばカスケディングスタイルシート（cascading style sheet）（CSS））のような異なるプログラミング言語でコードを生成することもできる。

40

【0017】

特定の実施形態に従うと、コンピュータによって実現される方法は、入力画像から、グ

50

ラフィックユーザインターフェイス（GUI）のGUI画面を示すGUI画面画像を検出するステップと、第1のテキストコンテンツアイテムを含むGUI画面画像の第1の領域を検出するステップと、GUI画面画像内の第1の領域の位置を判断するステップと、第1の領域のコンテンツを、テキストコンテンツを含まないプレースホルダに置き換えるステップとを含み得る。このコンピュータによって実現される方法はさらに、GUI画面画像の第2の領域内に位置する第1のユーザインターフェイス（UI）コンポーネントを検出するステップを含み得る。第2の領域はプレースホルダを有する第1の領域を含む。さらに、GUI画面画像内の第1のUIコンポーネントの位置を判断するステップと、機械学習ベースの分類器を用いて、第1のUIコンポーネントのUIコンポーネントタイプを判断するステップとを含み得る。機械学習ベースの分類器は、複数の訓練画像を含む訓練データを用いて訓練されてもよく、複数の訓練画像のうちの各訓練画像はUIコンポーネントを含み得る。訓練データはさらに、各訓練画像ごとに、訓練画像内のUIコンポーネントのUIコンポーネントタイプを特定する情報を含み得る。次に、GUIを実現するためのコードを生成するために使用可能なGUIモデルが生成されてもよい。GUIモデルは、第1のUIコンポーネントの情報と、第1のテキストコンテンツアイテムの情報とを含み得る。第1のUIコンポーネントの情報は、第1のUIコンポーネントについて判断したUIコンポーネントタイプと、GUI画面画像内の第1のUIコンポーネントの位置とを示す情報を含み得る。

10

【0018】

この方法のいくつかの実施形態において、第1のUIコンポーネントを検出するステップは、機械学習ベースの分類器がGUI画面画像内の第1のUIコンポーネントを検出するステップを含み得る。訓練データはさらに、各訓練画像ごとに、訓練画像内のUIコンポーネントの位置を含み得る。いくつかの実施形態において、この方法はさらに、GUI画面画像内の第1の領域の位置およびGUI画面画像内の第1のUIコンポーネントの位置に基づいて、第1のテキストコンテンツアイテムとともに第1のUIコンポーネントをグルーピングするステップをさらに含み、

20

GUIモデルを生成するステップはさらに、GUIモデルに、グルーピングを示す情報を含めるステップを含む。

【0019】

いくつかの実施形態において、このコンピュータによって実現される方法はさらに、第2のテキストコンテンツアイテムを含み得るGUI画面画像の第3の領域を検出するステップと、GUI画面画像内の第3の領域の位置を判断するステップと、GUI画面画像内に位置する第2のUIコンポーネントを検出するステップと、GUI画面画像内の第2のUIコンポーネントの位置を判断するステップと、機械学習ベースの分類器を用いて、第2のUIコンポーネントのUIコンポーネントタイプを判断するステップとを含み得る。このコンピュータによって実現される方法はさらに、第1の領域の位置、第3の領域の位置、第1のUIコンポーネントのUIコンポーネントタイプおよび位置、ならびに第2のUIコンポーネントのUIコンポーネントタイプおよび位置に基づいて、第1のテキストコンテンツアイテム、第2のテキストコンテンツアイテム、第1のUIコンポーネント、および第2のUIコンポーネントをグルーピングするステップと、グルーピングに基づいてGUI画面のレイアウトを決定するステップとを含み得る。GUIモデルを生成するステップはさらに、GUIモデルに、グルーピングおよびGUI画面のレイアウトを示す情報を含めるステップを含み得る。

30

40

【0020】

いくつかの実施形態において、このコンピュータによって実現される方法はさらに、GUI画面画像内の第3の領域の位置に基づいて、第2のテキストコンテンツはGUI画面画像内のいずれのUIコンポーネントにも対応付けられていないと判断するステップと、第2のテキストコンテンツアイテムはアクションを示すと判断するステップとをさらに含み得る。GUIモデルを生成するステップは、GUIモデルにおいて、GUI画面画像の第3の領域内の第2のテキストコンテンツアイテムはクリックブルテキストであることを

50

示すステップを含み得る。第2のテキストコンテンツのクリックはアクションを開始させることができる。

【0021】

いくつかの実施形態において、第1の領域は、背景上の第1のテキストコンテンツアイテムを含み得る。このコンピュータによって実現される方法はさらに、GUI画面画像の第1の領域内の画素の強度値に基づいてヒストグラムを生成するステップと、GUI画面画像の第1の領域内の、第1のテキストコンテンツアイテムに対応する画素の数と背景に対応する画素の数とに基づいて、ヒストグラムから、第1のテキストコンテンツアイテムの画素の強度値は背景の画素の強度値よりも高いと判断するステップと、第1の領域内の画素の強度値を反転させるステップとを含み得る。反転により、第1のテキストコンテンツアイテムの画素の画素値は、第1の領域内の背景の画素の強度値よりも低くなり得る。反転の実施後に第1の領域内の第1のテキストコンテンツアイテムを認識するステップを含み得る。ヒストグラムを生成する前に、第1の領域内の各画素の強度値に基づいて、GUI画面画像の第1の領域を、RGBサブ画像から2値サブ画像に変換するステップをさらに含む。

10

【0022】

この方法のいくつかの実施形態において、プレースホルダは画素の予め定められたパターンを含み得る。第1のUIコンポーネントのUIコンポーネントタイプを判断するステップは、第2の領域内のプレースホルダの存在とプレースホルダ内の画素の予め定められたパターンとに基づいて、GUI画面画像の第2の領域を分類するステップを含み得る。いくつかの実施形態において、画素の予め定められたパターンの画素値は、予め定められた値、または第1の領域の背景内の画素の画素値を含み得る。

20

【0023】

いくつかの実施形態において、GUIモデルを生成するステップは、GUIモデルの情報を、JavaScriptオブジェクト表記(JSON)フォーマットで格納するステップを含み得る。いくつかの実施形態において、この方法は、GUIモデルに基づいてGUIの1つ以上のインプリメンテーションを生成するステップをさらに含む。いくつかの実施形態において、GUIモデルに基づいてGUIの1つ以上のインプリメンテーションを生成するステップは、GUIモデルを用いて、第1のプラットフォームのためのGUIの第1のインプリメンテーションを生成するステップと、GUIモデルを用いて、第2のプラットフォームのためのGUIの第2のインプリメンテーションを生成するステップとを含み得る。第2のプラットフォームは第1のプラットフォームと異なる。いくつかの実施形態において、GUIモデルに基づいてGUIの1つ以上のインプリメンテーションを生成するステップは、GUIモデルを用いて、第1のプログラミング言語でGUIの第1のインプリメンテーションを生成するステップと、GUIモデルを用いて、第2のプログラミング言語でGUIの第2のインプリメンテーションを生成するステップとを含み得る。第2のプログラミング言語は第1のプログラミング言語と異なる。いくつかの実施形態において、GUIモデルに基づいてGUIの1つ以上のインプリメンテーションを生成するステップは、GUIモデルと1つ以上のコード生成テンプレートとを用いてGUIの1つ以上のインプリメンテーションを生成するステップを含み、各コード生成テンプレートはプラットフォームまたはプログラミング言語に対応付けられている。

30

40

【0024】

いくつかの実施形態において、機械学習ベースの分類器は、機械学習ベースの分類器は、画像ヒストグラムベースの非線形サポートベクターマシン分類器、または、人工ニューラルネットワークベースの分類器を含み得る。いくつかの実施形態において、この方法はGUIモデルに関するユーザフィードバックを受けるステップをさらに含む。ユーザフィードバックは、GUIモデルにおいて第1のUIコンポーネントについて指定されたUIコンポーネントタイプの代わりに第1のUIコンポーネントに対応付けられる新たなUIコンポーネントタイプを特定する情報、または、GUI画面内に存在するがGUIモデルに含まれない新たなUIコンポーネントを特定する情報、および新たなUIコンポー

50

ネットのUIコンポーネントタイプを示す情報を含み得る。この方法は、ユーザフィードバックに基づいて機械学習ベースの分類器を再訓練するステップをさらに含み得る。

【0025】

いくつかの実施形態において、ユーザフィードバックは、第1のUIコンポーネントまたは新たなUIコンポーネントのサブ画像を含み得る。この方法はさらに、複数の訓練画像から特徴を抽出するステップと、複数の訓練画像から抽出した特徴を、多次元空間内のデータポイントにマッピングするステップとを含み得る。データポイントは、多次元空間内の一組のクラスタを形成し、第1のUIコンポーネントまたは新たなUIコンポーネントのサブ画像から特徴を抽出するステップと、第1のUIコンポーネントまたは新たなUIコンポーネントのサブ画像から抽出した特徴を、多次元空間内のデータポイントにマッピングするステップと、第1のUIコンポーネントまたは新たなUIコンポーネントのサブ画像に対応するデータポイントと、一組のクラスタの各クラスタの中心との間の距離を判断するステップと、距離がしきい値未満であるという判断に応じて、第1のUIコンポーネントまたは新たなUIコンポーネントのサブ画像を訓練データに含めるステップとを含み得る。

10

【0026】

特定の実施形態に従うと、非一時的なコンピュータ読取可能媒体は、1つ以上のプロセッサによる実行が可能な複数の命令を格納し得る。これら複数の命令は、上記1つ以上のプロセッサによって実行されると、当該1つ以上のプロセッサに上記方法を実行させることができる。

20

【0027】

特定の実施形態に従うと、システムは、1つ以上のプロセッサと、この1つ以上のプロセッサに結合されたメモリとを備える。メモリは、当該1つ以上のプロセッサによって実行されると、このシステムに上記方法を実行させることができる。

【0028】

以下の明細書、請求項、および添付の図面を参照すると、これまでの記述はその他の特徴および実施形態とともに、より明らかになるであろう。

【図面の簡単な説明】

【0029】

【図1】特定の実施形態に係る、グラフィックユーザインターフェイス(GUI)の設計情報に基づいてGUIのGUIモデルを生成するためのシステムの一例の簡略化されたハイレベル図を示し、ダウンストリームコンシューマが、生成されたGUIモデルを使用することにより、GUIの1つ以上のインプリメンテーションを自動的に生成することができる。

30

【図2】特定の実施形態に係る、GUI画面のモックアップ画像の一例を示す図である。

【図3】特定の実施形態に係る、GUIの設計情報に基づいてGUIのGUIモデルを生成するためのシステムの一例を示す図である。

【図4】特定の実施形態に係る、機械学習ベースの分類器を用いてGUIモデルおよび/またはGUIの1つ以上のインプリメンテーションを生成するために実行されるハイレベル処理を示す簡略化されたフローチャートである。

40

【図5】特定の実施形態に係る、ユーザインターフェイスコンポーネントを検出し分類するために機械学習ベースの分類器を訓練するために実行されるハイレベル処理を示す簡略化されたフローチャートである。

【図6】特定の実施形態に係る、GUIの設計情報に基づいてGUIモデルおよび/またはグラフィックユーザインターフェイスのソースコードを生成するために実行されるハイレベル処理を示す簡略化されたフローチャートである。

【図7】特定の実施形態に係る、GUI画面の画像からテキストコンテンツアイテムを抽出するために実行されるハイレベル処理を示す簡略化されたフローチャートである。

【図8】特定の実施形態に係る、UIコンポーネントをクラスタリングすることによってグラフィックユーザインターフェイス画面に対して生成することができるUIコンポーネ

50

ント階層の一例を示す図である。

【図 9】特定の実施形態に係る、GUIモデルおよびGUI画面のためのGUIコードを自動的に生成するためのソフトウェアアプリケーションのGUI画面の一例を示す図である。

【図 10】特定の実施形態に係る、GUIモデルおよびGUI画面のためのGUIコードを自動的に生成するためのソフトウェアアプリケーションのGUI画面の一例を示す図である。

【図 11】特定の実施形態に係る、GUIモデルおよびGUI画面のためのGUIコードを自動的に生成するためのソフトウェアアプリケーションのGUI画面の一例を示す図である。

10

【図 12】特定の実施形態に係る、GUIモデルおよびGUI画面のためのGUIコードを自動的に生成するためのソフトウェアアプリケーションのGUI画面の一例を示す図である。

【図 13】特定の実施形態に係る、GUI画面について生成されたJSONフォーマットのGUIモデルの一例を示す図である。

【図 14】特定の実施形態に係る、入力GUI画面画像の一例と、本明細書に開示された技術に基づいて生成されたコードを用いて表示された対応するGUI画面画像とを示す図である。

【図 15】特定の実施形態に係る、本明細書に開示された技術に基づいて入力GUI画面画像の一例について生成されたhtmlコードの一例を示す図である。

20

【図 16 A】特定の実施形態に係る、本明細書に開示された技術に基づいて生成されたコードを用いて表示された入力GUI画面画像の別の例を示す図である。

【図 16 B】特定の実施形態に係る、本明細書に開示された技術に基づいて生成されたコードを用いて表示された対応するGUI画面画像の別の例を示す図である。

【図 17】特定の実施形態を実現するための分散型システムの簡略図を示す。

【図 18】特定の実施形態に係る、各種サービスがクラウドサービスとして提供され得るクラウドベースのシステム環境の簡略化されたブロック図である。

【図 19】特定の実施形態を実現するために使用し得るコンピュータシステムの一例を示す図である。

30

【発明を実施するための形態】

【0030】

詳細な説明

本開示は、概してアプリケーション開発に関し、より具体的には、アプリケーションのためのグラフィックユーザインターフェイス（GUI）を、GUIについての設計情報から開発することを自動化する技術に関する。方法、システム、1つ以上のプロセッサによる実行が可能なプログラム、コードまたは命令を格納する非一時的なコンピュータ読取可能記憶媒体などを含む、発明の各種実施形態が、本明細書に記載される。

【0031】

以下の記載において、説明を目的として、発明の特定の実施形態が十分に理解されるようにするために特定の詳細について述べる。しかしながら、これらの特定の詳細がなくとも各種実施形態は実施し得ることは明らかである。図面および説明は限定を意図しているのではない。本明細書において「例」または「例示的な」という単語は、ひとつの例、場合、または実例の役割を果たすことを意味するために使用されている。本明細書に記載の「例示的」なものまたは「例」として記載されるいかなる実施形態または設計も、必ずしもその他の実施形態または設計よりも好ましいまたは好都合であると解釈されることを意図している訳ではない。

40

【0032】

特定の実施形態において、アプリケーションのためのGUIは、1つ以上のGUI画面を含み得る。各画面は、ボタン、テキスト入力ボックス、ドロップダウンリスト、ドロップダウンメニュー、アイコン、表などのような1つ以上のユーザインターフェイス（UI

50

) コンポーネントを含む。アプリケーションのための GUI はまた、テキスト情報を含み得る。テキスト情報は、アプリケーションの GUI および / または各種 UI コンポーネントに対応付けられた機能および挙動を記述する、または、その他の情報もしくはユーザに対する命令を与える。

【 0 0 3 3 】

特定の実施形態において、GUI 設計者が設計した GUI 画面の画像 (GUI 画面画像とも呼ぶ) を解析することにより、テキスト情報を抽出し、UI コンポーネントを特定する。たとえば、UI コンポーネントを、機械学習ベースの分類器 (たとえばサポートベクターマシン分類器または畳み込みニューラルネットワークベースの分類器) によって検出し分類することにより、各 UI コンポーネントのタイプおよび / または対応付けられた機能を特定することができる。対応する、テキスト情報および UI コンポーネントの位置は、GUI 画面画像から判断することもできる。GUI 画面のレイアウトは、テキスト情報および UI コンポーネントの特定の属性に基づいて決定することができる。次に、たとえば、特定した UI コンポーネント、UI コンポーネントのタイプ、UI コンポーネントの位置、UI コンポーネントの対応付けられたテキスト情報、どの UI コンポーネントにも対応付けられていない可能性があるその他のテキスト情報、および、GUI 画面のレイアウトに基づいて、GUI 画面に対し、言語から独立した GUI モデルを生成することができる。生成したモデルは、データ交換フォーマットで記述してもよく、続いて、各種プログラミング言語を用いて各種プラットフォーム上に GUI 画面を実現するコードを生成することができる。いくつかの実施形態において、実行可能または解釈可能なコードを、GUI モデルに基づいて生成することにより、GUI についての設計情報に記述されているのと同じルック・アンド・フィールおよび機能を有する GUI を表示することができる。

【 0 0 3 4 】

いくつかの実施形態において、UI コンポーネントを検出または分類する前に、GUI 画面画像内のテキストコンテンツを抽出し、テキストコンテンツを含まないブレースホルダに置き換えてもよい。たとえば、いくつかの実施形態において、テキストコンテンツアイテムを含む GUI 画面画像の部分の画素の強度値に基づいてヒストグラムを生成することができる。GUI 画面画像の上記部分の画素を次に反転させることにより、この GUI 画面画像の部分内のテキストコンテンツの色を、この GUI 画面画像の部分内の背景の色よりも暗くしてもよい。次に、GUI 画面画像の反転された部分から、テキストコンテンツアイテムを認識することができる。いくつかの実施形態において、上記 GUI 画面画像の部分内のテキストコンテンツを、画素の予め定められたパターンを有する、または上記画面画像の部分の背景の画素値を有する、画素アレイに置き換えてもよい。いくつかの実施形態において、画素の予め定められたパターンは、テキストコンテンツアイテムの位置および / またはテキストコンテンツアイテムに対応付けられた UI コンポーネントのタイプを判断し易くすることができる。

【 0 0 3 5 】

特定の実施形態において、GUI について生成された GUI モデルは、アプリケーションのための 1 つ以上の GUI 画面に対応する情報をカプセル化してもよい。各 GUI 画面ごとに、GUI モデルは、GUI 画面に含まれる 1 つ以上のユーザインターフェイス (UI) コンポーネントを特定する情報を含み得る。各 GUI 画面ごとに、このモデルはまた、GUI 画面の構造に関する情報、たとえば、GUI 画面内のユーザインターフェイスコンポーネントの階層組織を特定する情報などを含み得る。たとえば、いくつかの実施形態において、UI コンポーネントを、たとえば、UI コンポーネントのタイプおよび位置に基づいてグルーピングすることにより、UI コンポーネントのサブグループ (たとえば表またはリスト) を形成することができる。サブグループをさらにクラスタリングすることにより、GUI 画面のより高いレベルのレイアウトを決定することができる。いくつかの実施形態において、テキストコンテンツアイテムもグルーピングすることにより、たとえばテキストのラインまたはテキストのパラグラフを形成することができる。いくつかの実施形態において、テキストコンテンツアイテムを、たとえば、テキストコンテンツアイテ

10

20

30

40

50

ムおよびUIコンポーネントの位置情報に基づいて、グルーピングするかまたはUIコンポーネントに対応付けることができる。

【0036】

いくつかの実施形態において、GUIモデルは、いずれのUIコンポーネントにも対応付けられない特定のテキストコンテンツアイテムを、クリックされるとアクションを開始させることができるクリック可能なテキストとして示すことができる。いくつかの実施形態において、GUIモデルは、設計されたGUI画面のルック・アンド・フィールを再現できるようにGUI画面の各種UIコンポーネントを如何にして表示すべきかを記述することができる。特定の実施形態において、GUIモデルはまた、GUI画面上でユーザインターフェイスコンポーネントに対応付ける機能に関する情報を含み得る。ユーザインターフェイスコンポーネントに対応付けられる機能は、たとえば、機械学習ベースの分類器によって分類されたユーザインターフェイスコンポーネントのタイプ、および/または対応付けられたテキストコンテンツアイテム（もしあれば）に基づいて、判断することができる。いくつかの実装例において、GUIモデルはメタデータとして生成されてもよい。いくつかの実装例において、GUIモデルはJavaScriptオブジェクト表記（JSON）フォーマットのような、言語から独立したデータ交換フォーマットで記述されてもよい。いくつかの実施形態において、ユーザは、GUIモデルに関するフィードバックを提供することができる。次に、このユーザフィードバックを用いて、機械学習ベースの分類器Qを改善（たとえば再訓練）することができる。

10

20

【0037】

設計情報に基づいて生成されたGUIモデルは、各種ダウンストリームコンシューマが使用できる。たとえば、ダウンストリームコンシューマは、このモデルを使用することにより、GUIを実現するためのコードを、自動的にかつ実質的に手作業のコーディングなしで、生成することができる。このコードは、GUIを表示するために、1つ以上のプロセッサが実行可能な実行可能プログラムであってもよく、または、たとえばウェブブラウザが解釈可能な解釈可能プログラムであってもよい。同じGUIモデルを異なるコンシューマが使用することが可能である。たとえば、第1のコンシューマが第1のプラットフォーム（たとえばiOS（登録商標））に対して実行可能なものを自動的に生成するためにこのGUIモデルを使用してもよく、第2のコンシューマが異なるプラットフォーム（たとえばAndroid（登録商標））に対して第2の実行可能なものを自動的に生成するためにこの同じGUIモデルを使用してもよい。また、GUIモデル（たとえばJSONフォーマット）を用いて、マークアップ言語（たとえばHTMLもしくはXML）またはスタイルシート言語（たとえばカスケーディングスタイルシート（CSS））のような異なるプログラミング言語でコードを生成することもできる。

30

【0038】

本明細書で使用するUIコンポーネントは、GUI画面のグラフィカルコンポーネントを意味し得る。UIコンポーネントは、ボタン、テキスト入力ボックス、ドロップダウンリスト、ドロップダウンメニュー、チェックボックス、ラジオボタン、スイッチボタン、アイコン、表、（人々または物体の）写真、ラインディバイダ、コンテナなどのような、さまざまなタイプのUIコンポーネントを含み得る。GUI画面は1つ以上のテキスト領域を含み得る。各テキスト領域はテキストコンテンツを含み得る。各テキスト領域内のテキストコンテンツは、1つ以上のテキストコンテンツアイテムを含み得る。本明細書で使用するテキストコンテンツアイテムは、任意の言語の特定のテキスト文字を含むGUI画面の非グラフィカルコンポーネントを意味し得る。たとえば、テキストコンテンツアイテムは、単語、特殊文字、または短い語句を含み得る。本明細書で使用する、UIコンポーネントは、UIコンポーネントまたはテキストコンテンツアイテムを含む、GUI画面に示される何等かの要素を含み得る。

40

【0039】

説明のために、本明細書において特定の例について述べる。しかしながら、これらの例は、限定ではなく例示を意図している。本明細書に開示される教示は、モバイルアプリケ

50

ーション、非モバイルアプリケーション、デスクトップアプリケーション、ウェブアプリケーション、企業アプリケーションなどのような、さまざまな種類のアプリケーションにも適用可能である。さらに、本開示の教示は、特定の動作環境（たとえばオペレーティングシステム、デバイス、プラットフォームなど）に限定されるのではなく、複数の異なる動作環境に適用することが可能である。

【0040】

図1は、特定の実施形態に係る、グラフィックユーザインターフェイス（GUI）についての設計情報に基づいてGUIのGUIモデルを生成するためのシステム100の一例の簡略化されたハイレベル図を示し、生成されたGUIモデルは、ダウンストリームコンシューマが使用して、GUIの1つ以上のインプリメンテーションを自動的に生成することが可能である。図1に示されるシステム100は、一例にすぎず、本開示の範囲を不当に制限することを意図しているのではない。当業者は多数の可能な変形、代替形、および修正を認識するであろう。たとえば、いくつかの実装例において、システム100は、図1に示されているものよりも多くのまたは少ないサブシステムまたはコンポーネントを有していてもよく、2つ以上のサブシステムを組み合わせてもよく、または、サブシステムの異なる構成または配置を有していてもよい。

10

【0041】

図1に示されるように、システム100は、GUIの1つ以上のGUI画面画像104を入力として受け、1つ以上のGUI画面画像104に基づいてGUIのGUIモデル124を生成するように構成された、モデル生成システム（model generation system）（MGS）102を含み得る。次に、GUIモデル124は、1つ以上のダウンストリームコンシューマ103によって使用されることができる。1つ以上のダウンストリームコンシューマ103は、GUIモデル124に基づいて、実質的に手作業によるコーディングなしで、GUIの1つ以上のGUIインプリメンテーション110、112、および114を生成することができる。GUIインプリメンテーション110、112、および114は、1つ以上のプロセッサによって実行可能であり、それにより、異なるプラットフォーム上にGUIを表示することができる。

20

【0042】

先に示したように、GUIのためのGUIモデルまたはインプリメンテーションの作成前に、GUIについての情報を定義または収集してもよい。1つ以上のGUI画面画像104を、顧客またはクライアント調査、マーケティング調査、ならびに、GUIのルック・アンド・フィールおよびGUIに含めるべき機能を伝えるその他の情報源に基づいて、集約してもよい。したがって、1つ以上のGUI画面画像104は、GUIの所望のルック・アンド・フィールおよび機能を記述することができる。機能コンサルタント、ユーザー体験（UX）設計者などのようなさまざまな異なる人員がGUI画面画像104の準備に関わり得る。GUI画面画像104は、コンピュータ支援設計ツールを用いて生成しデジタルフォーマットで保存してもよく、または紙の上にスケッチとして手作業で作成してからスキャンしてデジタル画像にしてもよい。

30

【0043】

図1に示される実施形態のような特定の実施形態において、GUI画面画像104は、1つ以上の画像を含み得る。各画像は、1つのGUI画面のためのものであってもよく、複数のGUI画面のためのものであってもよい。特定の実施形態において、GUI画面画像104は、シーケンスとして受けてもよく、順序を有していてもよく、このシーケンスまたは順序は、画像によって示される画面と画面との間の流れを特定する。GUI画面画像104は、画面の設計およびルック・アンド・フィール、画面と画面との間の移行などを示し得る。各GUI画面画像104は、1つ以上のUIコンポーネント134とテキスト情報132とを含み得る。GUI画面画像104は、ビットマップファイル、JPEGファイル、PNG（ポータブルネットワークグラフィックス）ファイル、GIFファイル、PDFファイルなどのような、多種多様なフォーマットのうちの1つで受けることができる。多種多様な技術を用いてGUI画面画像104を生成することができる。たとえば

40

50

、G U I画面画像104は、カメラ、スキャナなどのような画像取込デバイスを用いて取り込まれた写真である画像を含み得る。別の例として、G U I画面画像104は、スクリーンショット、たとえば既存のアプリケーションの画面のスクリーンショットである画像を含み得る。この場合、これから開発されるアプリケーションが、既存のアプリケーションと同様のG U I画面を有することになる(たとえば、既存のアプリケーションは、アプリケーションの新たなバージョンを開発しようとしているアプリケーションの前のバージョンの可能性がある)。G U I画面画像104はまた、画像編集アプリケーション(たとえばAdobe Corporation(登録商標)が提供する各種画像編集アプリケーション)のようなアプリケーションを用いて生成された画像を含み得る。G U I画面画像104はまた、各種ワードプロセッサ(たとえばM S W O R D(登録商標))、図作成アプリケーション(たとえばV i s i o(登録商標))、およびその他のアプリケーションのような画像を作成または編集可能なソフトウェアアプリケーションを用いて生成された画像を含み得る。

10

【0044】

G U I画面画像104を用いて開発しようとしているアプリケーションは、モバイルアプリケーション(たとえばモバイルデバイスが実行可能なアプリケーション)、デスクトップアプリケーション、ウェブアプリケーション、企業アプリケーションなどを含むがこれらに限定されない各種のアプリケーションのうちの一つであってもよい。アプリケーションは、多種多様なデバイス(たとえばスマートフォン、タブレット、ラップトップ、デスクトップコンピュータなど)およびプラットフォーム(たとえばi O S(登録商標)プラットフォーム、A n d r o i d(登録商標)プラットフォーム、W i n d o w(登録商標)プラットフォームなど)を対象とし得る。たとえば、G U I画面画像104は、A p p l e(登録商標)i P h o n e(登録商標)上で実行されるように設計されたモバイルアプリケーションのための1つ以上のG U I画面の1つ以上の画像を含み得る。この場合、画面は、モバイルデバイスの画面の面積全体またはその一部にわたっている。

20

【0045】

各G U I画面画像104は、ボタン、テキスト入力ボックス、ドロップダウンリスト、ドロップダウンメニュー、チェックボックス、アイコン、表、(人々または物体の)写真などのような、1つ以上のU Iコンポーネント134を含み得る。いくつかの実施形態において、いくつかのU Iコンポーネント134は、対応付けられたテキストを含み得る。たとえば、あるボタンは、このボタンの上の「次へ」、「取消」、「確認」、または「O K」のような単語を含み得る。また、U Iコンポーネント134は、サイズ、位置、または対応付けられたアクションもしくは機能のような、対応付けられた属性を有し得る。たとえば、U Iコンポーネント134は、G U I画面画像104内のいずれかの位置にあればよい。いくつかの実施形態において、U Iコンポーネント134は、表、リスト、ツリー構造、フローチャート、組織図などのような、レイアウトまたは階層構造に従って配置されてもよい。いくつかのU Iコンポーネント134は、クリック可能であってもよく、選択可能であってもよく、そうでなければユーザ入力(たとえばユーザエントリ)を取り込んでもよく、その他いくつかのU Iコンポーネントは、静的であってもよく、またはユーザ入力を取り込まなくてもよい。

30

40

【0046】

図1に示されるように、各G U I画面画像104はテキスト情報132を含み得る。テキスト情報132は、画面のルック・アンド・フィール(たとえば、画面の構造、画面のユーザインターフェイスコンポーネント、使用されるフォント、画面上で使用される色(たとえば前景および背景色)、画面の機能およびそのユーザインターフェイスコンポーネント、画面およびそのユーザインターフェイスコンポーネントが表示するデータなどを含むが、これらに限定されない、アプリケーションの機能のテキスト記述を提供し得る。また、テキスト情報132は、その他有益な材料またはユーザに対する命令を含み得る。いくつかの実施形態において、テキスト情報132はまた、複数の画面間の移行を記述し得る。

50

【0047】

従来のアプリケーション開発環境において、アプリケーションは、GUI画面画像104に基づいて手作業で構築することができる。たとえば、GUI画面画像104は、アプリケーションのためのGUI画面画像104を実現するためのコードおよびアプリケーションのその他のコンポーネントを記述するタスクが課された技術者または開発者を含む工学技術組織に提供することができる。これらの開発者は、GUI画面画像104を理解してから、GUI画面画像104に定められているアプリケーションのGUIおよび対応付けられた機能を実現するコードを手作業で書き込む必要がある場合がある。

【0048】

特定の実施形態に従うと、モデル生成システム102は、GUI画面画像104を入力として取り込み、たとえば、モデル生成器120、UIコンポーネント分類器136、および/またはメモリ122に格納されている参照情報121を用いて、GUIモデル124を自動的に生成するように構成することができる。次に、GUIモデル124を用いて、GUIの1つ以上のインプリメンテーションを自動的に生成することができる。これらのインプリメンテーションは、GUIを実現するコードおよびロジックを含み得る。これらのインプリメンテーションは、1つ以上のプロセッサが実行することができる実行可能なインプリメンテーションを含み得る。

10

【0049】

図1に示されるように、モデル生成システム102は、協働することによりGUIモデル124を生成するように構成された1つ以上のサブシステムを含み得る。これらのサブシステムは、ハードウェア、コンピュータシステムの1つ以上の処理ユニット（たとえばプロセッサもしくはコア）によって実行されるソフトウェア（たとえばコード、命令、プログラム）、またはその組み合わせで実現されてもよい。ソフトウェアは、メモリ122のような非一時的な記憶媒体（たとえばメモリデバイス）に格納されてもよい。図1に示される実施形態において、モデル生成システム102は、1つのモデル生成器120を含む。代替の実施形態において、モデル生成システム102は複数のモデル生成器120を含み得る。

20

【0050】

モデル生成システム102は、さまざまな異なる技術を用いて、GUI画面画像104からGUIモデル124を構築することができる。たとえば、モデル生成システム102は、GUI画面画像104を処理し解析することにより、GUIに対して指定される1つ以上のGUI画面を決定することができ、各GUI画面ごとに、この画面上に含まれる一組のユーザインターフェイスコンポーネントおよびこれらのユーザインターフェイスコンポーネントの物理的配置を決定することができる。いくつかの実施形態において、このGUIモデル生成処理は、たとえば、GUI画面について、一組のユーザインターフェイスコンポーネント（たとえばボタン、ドロップダウンリスト、セグメントなど）およびそれらの属性（たとえばラベル、サイズ、位置）を決定することと、GUI画面内のUIコンポーネントの物理的レイアウトを決定すること（たとえばUIコンポーネントまたはUIコンポーネントのグループの階層的包含関係を決定すること）と、UIコンポーネントのうち1つ以上に対応付けられる機能を決定することを含み得る。

30

40

【0051】

特定の実施形態において、モデル生成システム102は、GUI画面画像104を処理し解析し、たとえば、UIコンポーネントの各々の境界を検出可能な輪郭検出技術を用いて、画面に含まれる1つ以上のユーザインターフェイスコンポーネントを特定するように、構成することができる。UIコンポーネントの各々の境界に基づいて、各UIコンポーネントのサイズおよび位置を決定することができる。各UIコンポーネントの境界の内部の画像を、抽出し、サポートベクターマシン分類器、畳み込みニューラルネットワークベースの分類器のような機械学習ベースの分類器を用いて（たとえばソフトマックス（Softmax）分類器を用いて）、または、その他の深層ニューラルネットワークベースの分類器を用いて、分類することができる。分類器は、UIコンポーネントを、ボタン、テキスト

50

入力ボックス、ドロップダウンリスト、ドロップダウンメニュー、チェックボックス、アイコン、表、（人々または物体の）写真などのような、多数の可能なUIコンポーネントのタイプのうちの、あるUIコンポーネントのタイプに分類することができる。GUI画面のコンポーネントおよびGUI画面のレイアウトを決定するためにモデル生成システム102が実行し得る処理のその他の説明および例を以下に示す。

【0052】

特定の実施形態において、モデル生成システム102は、GUI画面画像104からテキスト情報132を抽出することができる。たとえば、モデル生成システム102のいくつかの実装例において、テキスト検出ツールを用いることにより、テキストコンテンツを含み得るGUI画面画像104内の領域の位置（たとえば座標）を判断することができ、次に、光学文字認識（OCR）ツールを用いて、GUI画面画像104内のこれらの領域からテキストコンテンツアイテムを抽出する（たとえば認識する）ことができる。いくつかの実施形態において、各UIコンポーネントのサイズおよび位置ならびにテキストコンテンツアイテムの位置および情報に基づいて、いくつかのテキストコンテンツアイテム（たとえばクリック可能なボタン上のテキスト）を特定のUIコンポーネント（たとえばクリック可能なボタン）に対応付けることができる。

10

【0053】

特定の実施形態において、モデル生成システム102が実行する処理は、モデル生成器120およびUIコンポーネント分類器136を含むモデル生成システム102にアクセス可能な参照情報121によってガイドされてもよい。参照情報121はさまざまな種類の情報を含み得る。たとえば、特定の実施形態において、参照情報121は、モデル生成システム102が実行する胥吏をガイドするさまざまなルールを含み得る。特定の実施形態において、参照情報121は、GUIに対して指定される1つ以上のGUI画面、および/または各GUI画面ごとの、この画面に含まれる一組のユーザインターフェイスコンポーネント、およびGUI画面の物理的レイアウトを決定するためにモデル生成システム102が使用し得るルールを含み得る（たとえば、UIコンポーネントおよびテキストコンテンツアイテムクラスタリングのためのルール）。図1に示される実施形態において、参照情報121はメモリ122に格納されてもよい。いくつかの他の実施形態において、参照情報121は、モデル生成システム102から遠隔にある場所に格納されてもよい。さらに他の実施形態において、参照情報121の一部がモデル生成システム102において局所的にメモリ122に格納されてもよく、その他の部分がモデル生成システム102から遠隔に格納されてもよい。

20

30

【0054】

特定の実施形態において、モデル生成システム102が実行する処理または処理の一部は、各種機械学習技術を用いて実行してもよい。たとえば、機械学習技術を用いて、UIコンポーネントの検出および分類、テキスト検出、テキストコンテンツ（たとえば手書きのテキストコンテンツ）認識、またはUIコンポーネントおよびテキストコンテンツアイテムのクラスタリングのために、使用してもよい。これらの機械学習技術に関連する情報（たとえばニューラルネットワークの重み、畳み込みニューラルネットワークのフィルタ、機械学習に使用されるモデル、モデルを訓練するためのデータ、モデルのアプリケーション、モデルの精度を改善するためのフィードバック情報など）も、参照情報121に含まれていてもよい。特定の実施形態において、ルールベースの処理と機械学習ベースの技術との組み合わせを、モデル生成システム102が使用してもよい。よって、参照情報121は、機械学習ベースのモデルに関連する情報およびルールを含み得る。

40

【0055】

特定の実施形態において、参照情報121はまた、さまざまなタイプのUIコンポーネントに関する情報を含み得る。UIコンポーネントのタイプについての情報は、このタイプ（たとえばボタン、チェックボックス、入力ボックスなど）の名称（たとえば識別）、UIコンポーネントのタイプについてのパラメータ（たとえば属性、要素、またはフィールド名）、UIコンポーネントのタイプについての機能またはアクション、接続性、およ

50

び記憶情報、および/またはUIコンポーネントのタイプについてのデータソースなどのような、情報を含み得る。モデル生成システム102は、このような情報を用いることにより、GUIモデル124において、特定したUIコンポーネントを記述し特定のアクションをUIコンポーネントに対応付ける。

【0056】

特定の実施形態において、参照情報121は、ユーザ（たとえばプログラマー）またはモデル生成システム102の管理者によって構成されてもよい。いくつかの他の実施形態において、参照情報121は、1つ以上の機械学習技術を用いて構築されてもよい。たとえば、参照情報121は、訓練データおよび教師あり機械学習技術を用いて構築された1つ以上の機械学習ベースのモデルを含み得る。1つ以上の機械学習ベースのモデルを用いることにより、GUIモデル124を生成することができる。

10

【0057】

特定の実施形態において、GUIモデル124を、モデル生成器120が生成した1つ以上のファイルに存続させてもよい。GUIモデル124はさまざまなフォーマットで記述し得る。たとえば、いくつかの他の実施例において、GUIモデル124は、JavaScriptオブジェクト表記(JSON)フォーマットのような、言語から独立したデータ交換フォーマットで記述することができる。特定の実施形態において、モデル情報は、拡張可能マークアップ言語(XML)またはjQueryのようなマークアップ言語で符号化されてもよい。たとえば、モデル生成システム102は、ともにGUIモデル124を表す1つ以上のXMLファイルを生成してもよい。生成したファイルは、メモリ122に、または、モデル生成システム102がアクセス可能な何等かのその他のメモリ場所に格納してもよい。特定の実施形態において、GUIモデル124を、モデル生成システム102により、最初にファイルに存続させることなく、1つ以上のダウンストリームコンシューマ、たとえばコード生成器126、128、および130に送ってもよい。

20

【0058】

次に、GUIモデル124を、1つ以上のダウンストリームモデルコンシューマ103が使用することができる。たとえば、モデルコンシューマ103は、GUIモデル124に基づいて1つ以上のGUIインプリメンテーション110、112、および114を生成するように構成し得る。GUIインプリメンテーション110、112、および114は各々、GUIモデル124において特定された情報に基づき得る。GUIモデル124は設計されたGUI画面画像104に基づいて生成されるので、GUIモデル124に基づいて生成されるGUIインプリメンテーションは、GUI画面画像104に記述されているルック・アンド・フィールおよび機能を有し得る。たとえば、GUIモデル124は、特定の一組のUIコンポーネントを含み特定の一組の機能またはアクションにマッピングされた特定のGUIウィンドウまたは画面を指定する情報を含み得る。GUIモデル124に基づいて生成されるGUIインプリメンテーション（たとえばGUIを実現するコードまたは命令）は、特定の一組のUIコンポーネントを含み特定の一組の機能またはアクションにマッピングされた特定のGUI画面をインスタンス化するためのコードおよびロジックを含み得る。

30

【0059】

したがって、GUIインプリメンテーションは、GUI画面画像104に基づいて生成されたGUIモデル124によって記述されたGUI画面および対応付けられたアクションまたは機能を実現することができる。たとえば、GUIモデル124が特定の物理的レイアウトで配置された一組のユーザインターフェイスコンポーネントを含む特定の画面を指定している場合、この画面および特定の物理的レイアウトをGUIインプリメンテーションによって実現することができる。GUIモデル124が特定のユーザインターフェイスコンポーネントの特定の機能を指定している場合、このモデルに基づいて生成されたGUIインプリメンテーションは、この特定の機能を実現しこの機能を特定のユーザインターフェイスコンポーネントに対応付けるためのロジックを含み得る。特定の実施形態において、GUIインプリメンテーションは、特定のユーザインターフェイスコンポーネント

40

50

を、この特定のユーザインターフェイスコンポーネントに対応付けられる特定の機能を実現するコードにリンクさせることを可能にするフックを提供し得る。特定の実施形態において、GUIインプリメンテーションは、開発者が、追加機能を実現する追加コードをGUIインプリメンテーションにフックさせるまたは追加することを可能にするフックを提供するコードを含み得る。

【0060】

特定の実施形態において、ダウンストリームモデルコンシューマ103は1つ以上のコード生成器126、128、および130を含み得る。これらのコード生成器は、GUIモデル124を入力として取り込み、場合によっては、たとえば、異なるプログラミング言語および/または異なるプラットフォームのためのコード生成テンプレート140に基づいて、異なるプログラミング言語および/または異なるプラットフォームでGUIのコードインプリメンテーションを生成するように構成されている。コード生成器は、GUIモデル124を入力として取り込み、GUIをそのコード生成器に固有の言語で生成してもよい。インプリメンテーションは、1つ以上のプロセッサが実行可能なGUIの実行可能インプリメンテーションであってもよい。たとえば、コード生成器126は、モデル124を入力として取り込み、第1のプラットフォーム（たとえばiOS（登録商標）プラットフォーム）のための第1の言語でGUIインプリメンテーション110を生成してもよい。コード生成器128は、第1のプラットフォームのためのGUIモデル124を用いて第2の言語でGUIインプリメンテーション112を生成してもよい。コード生成器130は、Android（登録商標）プラットフォームのためにGUIモデル124を用いてGUIインプリメンテーション114を生成してもよい。GUIインプリメンテーションをコンパイルする（または解釈する、またはそれに対して実行される何等かのその他の処理）ことにより、GUIの実行可能バージョンを生成してもよい。

10

20

【0061】

特定の実施形態において、GUIインプリメンテーション110、112、および114の各々は、GUIを実現するために使用可能なコード生成テンプレートに対応する。コード生成テンプレートは、コンピュータシステムの1つ以上のプロセッサが実行するために実行可能なGUIを生成するためにコンパイルまたは解釈することが可能なハイレベルコード（方法、機能、クラス、イベントハンドラなどを含み得る）を含む1つ以上のソースコードファイルを含み得る。このようにして、GUIの実行可能なインプリメンテーションを、GUIモデル124に基づいて自動的に生成することが可能であり、実行可能なインプリメンテーションは、GUI設計情報に記述されているGUIのルック・アンド・フィールならびにGUIおよびUIコンポーネントの機能をカプセル化する。たとえば、コード生成器126を、GUIモデル124に対応するマークアップコードを含む1つ以上のファイルを受け、マークアップコード（たとえばXML）を（ハイレベル）ソースコード（たとえばJava（登録商標）、C++、またはその他のプログラミング言語）に変換することによって1つ以上のソースコードファイルを含むGUIインプリメンテーション110を出力するように、構成してもよい。

30

【0062】

いくつかの実装例において、モデル生成システム102は、ユーザ106が、直接またはクライアントデバイス108を介してモデル生成システム102とやり取りすることを可能にする1つ以上のインターフェイスを提供し得る。これらのユーザインターフェイスは、たとえば、各種GUI、コマンドラインインターフェイス（CLI）、およびその他のインターフェイスを含み得る。特定の実施形態において、モデル生成システム102は、統合開発環境（integrated development environment）（IDE）の一部であってもよい。

40

【0063】

クライアントデバイス108は、パーソナルコンピュータ、デスクトップコンピュータ、モバイルもしくはハンドヘルドデバイス（たとえばラップトップ、スマートフォン、タブレットなど）、またはその他の種類のデバイスを含むがこれらに限定されない、異なる

50

種類のものであってもよい。特定の実施形態において、クライアントデバイス108は、直接または通信ネットワークを介してモデル生成システム102に通信可能に結合されてもよい。通信ネットワークはさまざまな種類のものであってもよく、1つ以上の通信ネットワークを含み得る。通信ネットワークの例は、インターネット、ワイドエリアネットワーク(WAN)、ローカルエリアネットワーク(LAN)、イーサネット(登録商標)ネットワーク、パブリックまたはプライベートネットワーク、有線ネットワーク、無線ネットワークなど、およびその組み合わせを含み得るがこれらに限定されない。異なる通信プロトコルを用いることにより、クライアントデバイス108とモデル生成システム102との間の通信を容易にしてもよく、これは、IEEE802.XXプロトコルスイート、TCP/IP、IPX、SAN、AppleTalk(登録商標)、Bluetooth(登録商標)、およびその他のプロトコルなどの無線プロトコルおよび有線プロトコル双方を含む。

10

【0064】

ユーザとモデル生成システム102とのやり取りは各種形態を取ることができる。ユーザは、GUI画面画像104を、モデル生成システム102が提供する1つ以上のインターフェイスを用い、上記やり取りを介してモデル生成システム102に提供することができる。特定の実施形態において、モデル生成システム102の出力(たとえばGUIモデル124)または処理の状態に関する情報を、モデル生成システム102が、クライアントデバイス108に伝達しクライアントデバイス108を介してユーザ106に示してもよい。たとえば、GUIモデル124に関連する情報をユーザ106に対して提示してもよい。そうすると、ユーザ106は、GUIモデル124の精度をレビューし、必要であればGUIモデル124に対して変更を加えることができる。このようにして、ユーザ106は、モデル生成システム102が生成したGUIモデル124に関するフィードバックを提供することが可能である。そうすると、モデル生成システム102はこのフィードバックを用いて参照情報121(たとえば訓練サンプル)をアップデートすることができる。アップデートされた参照情報121は、1つ以上の機械学習ベースのモデルを再訓練するためおよび/または次世代のGUIモデル124を生成するために使用することができる。たとえば、ユーザ106が、GUIモデル124をレビューしながら、GUIモデルにおいて示されているUIコンポーネントのうちの1つが間違っ て分類されたと判断した場合、ユーザは、フィードバック(たとえば誤って分類されたUIコンポーネントの画像およびUIコンポーネントの実際のクラスまたはタイプ)をクライアントデバイス108を介してモデル生成システム102に与えることができる。モデル生成システム102は、ユーザフィードバックに基づいて、UIコンポーネントを適切なアクションに対応付ける、UIコンポーネントを再度グルーピングする、またはレイアウトを再度生成するなどして、GUIモデル124のアップデートを行うことができる。このようにして、ユーザからのフィードバックループにより、モデル生成システム102の精度を経時的に改善することが可能である。次に、注釈付きの画像を参照情報として保存し、何等かの機械学習ベースのモデル(たとえばUIコンポーネント分類器)を再訓練するための訓練サンプルとして使用してもよい。

20

30

【0065】

先に述べたように、モデル生成システム102は、GUI画面画像104に基づいて、自動的にかつ実質的にユーザの手作業による介入なしで、GUIモデル124を生成するように構成されている。さらに、同じGUIモデル124を用いて、異なるデバイス、プラットフォーム、および/または言語のためのGUIインプリメンテーションを生成することができる。多くの場合、GUIモデル124をダウンストリームモデルコンシューマが用いてGUIインプリメンテーションを自動的に生成することができる。たとえば、GUIインプリメンテーションは、開発者による実現のために手作業でコードを記述することなく、GUIモデル124に基づいて生成することができる。このようにして、実行可能なGUIインプリメンテーションを、実質的にユーザの手作業による介入なしで、またはアプリケーションのためのコードまたはロジックを手作業で記述することなく、GUI

40

50

画面画像 104 から自動的に生成することができる。この自動化レベルは、実質的にアプリケーション開発サイクルを高速化し開発コストを低減することが可能である。いくつかの実施形態において、GUIモデル 124 を用いて、GUIインプリメンテーションのテストを自動化するためのテストを生成することもできる。

【0066】

図2は、特定の実施形態に係るGUI画面200のモックアップ画像の一例を示す。GUI画面200の画像は、GUI設計者から、開発者がアプリケーションのためのGUIを実現するために使用するようになるために提供することができる。GUI画面200は、1つ以上のUIコンポーネントと1つ以上のテキストコンテンツアイテムとを含み得る。UIコンポーネントは、GUI画面200上のいずれかの位置にあればよい。UIコンポーネントは、たとえば、1つ以上のボタン、テキスト入力ボックス、ドロップダウンリスト、ドロップダウンメニュー、チェックボックス、アイコン、表、(人々または物体の)写真、ラインディバイダ、コンテナなどを含み得る。たとえば、図2に示されるように、GUI画面200は、ボタン214、テキスト入力ボックス206、ラジオボタン210、ドロップダウンリスト212、およびドロップダウン表8(たとえばカレンダー)208を含み得る。いくつかのUIコンポーネントは、対応付けられたテキストコンテンツアイテムを含み得る。たとえば、ボタン214は、ボタン上の「次へ」、「取消」、または「OK」ボタンのようなテキストコンテンツアイテム218を含み得る。UIコンポーネントは、サイズ、色、位置のような対応付けられた属性、または、対応付けられたアクションもしくは機能を有し得る。

10

20

【0067】

いくつかの実施形態において、UIコンポーネントは、表、リスト、ツリー構造、フローチャート、組織図などのようなレイアウトまたは階層構造に従って、GUI画面200上に配置することができる。たとえば、図2に示される例において、テキスト入力ボタン206は縦方向に並べてもよく、表を形成してもよい。

【0068】

いくつかのUIコンポーネントは、クリック可能、選択可能、またはその他のユーザ入力を取り込んでもよく、その他いくつかのUIコンポーネントは、静的でユーザ入力を取り込まないものであってもよい。たとえば、テキスト入力ボックス206は、キーボードからのユーザ入力を取り込むことができ、ラジオボタン210およびドロップダウンは(たとえばマウスを用いて)手作業で選択可能であり、ボタン214はクリックされることにより新たなGUI画面を表示させることができる。GUI画面200のUIコンポーネントは、静的コンポーネントであっても動的コンポーネントであってもよい。静的コンポーネントは、GUI画面上に表示されたその値がインスタントが異なっても不変のコンポーネントである。動的コンポーネントは、GUI画面に表示されたその値が、インスタントが異なると変化し得るコンポーネントである。たとえば、図2において、いくつかのテキスト入力ボックスは静的コンポーネントである。ボタン214は、各々が異なるタイプのUIコンポーネントであれば、静的コンポーネントであり得る。ボタン214は、同じタイプのUIコンポーネントに対応付けられているのであれば、動的コンポーネントであり得るものであり、ボタン各々についてテキストがUIコンポーネントの一部である場合と比較して、対応するテキストを動的に追加することにより、同じUIコンポーネント画像を用いて生成し得るものである。

30

40

【0069】

図2に示されるように、GUI画面200は、タイトル220、入力ボックスの説明204、デフォルト入力216、およびボタン214に対応付けられたテキストコンテンツアイテム218のような同一タイプの異なるUIコンポーネントを区別するテキスト情報といった、さまざまなテキスト情報を含み得る。タイトル220のようないくつかのテキスト情報は、アプリケーションの機能のテキスト記述を提供し得る。入力ボックスの説明2045のようないくつかのテキスト情報は、いくつかのUIコンポーネントのテキスト記述を提供し得る。テキストコンテンツアイテム218のようないくつかのテキスト情報

50

の各々は、UIコンポーネントの一部であってもよく、ボタン214のような、対応付けられたUIコンポーネントの機能を記述してもよい。デフォルト入力216のようなくつかのテキスト情報は、その値が入力、選択、そうでなければ変更され得るいくつかのUIコンポーネントのデフォルト値を提供し得る。テキスト情報は、フォント、サイズ、色、位置などのような、対応付けられた属性を有し得る。いくつかの実施形態において、テキスト情報はユーザに対する命令を含み得る。テキスト情報は静的であっても動的であってもよい。図2における静的テキストの例は、ヘッダセクションにおけるヘッダタイトル「アカウント作成」、本体セクションにおける属性タイトル「名」、「姓」、「Eメール」、「ユーザ名」などを含み得る。図2における動的テキストの例は、本体セクションにおける属性値「英語」、「Peter」など、その値があるインスタンスの第1の人物に関する情報から、別のインスタンスの第2の人物に関する情報に変更する可能性があるものを含み得る。

10

【0070】

各種情報をGUI画面画像から抽出し、画面のUIコンポーネントおよびテキストコンテンツを識別するために使用することができる。識別したコンポーネントの特定の属性を、たとえば、機械学習ベースのモデルを用いて判断することができる。次に、識別したコンポーネントの属性を用いて、GUIモデルおよび/またはGUIを実現するためのコードを生成することができ、詳細は以下に述べる通りである。

【0071】

図3は、特定の実施形態に係る、GUIについての設計情報に基づいてGUIのためのGUIモデルを生成するためのシステム300の一例を示す。この例において、システム300は、クライアントサブシステム310とサーバサブシステム320とを含むクライアント-サーバアーキテクチャを有する。クライアントサブシステム310は、GUIプロトタイプを画像またはスケッチの形態でアップロードするためのプロビジョンと、GUIインプリメンテーションを生成するためのコード生成ロジックとを含み得る。いくつかの実施形態において、コード生成ロジックは、クライアントサブシステム310ではなくサーバサブシステム320上に実現し得る。サーバサブシステム320は、モデル生成システム330と、クライアントサブシステム310とサーバサブシステム320との間の相互運用性および分離を提供するレプレゼンテーション・ステート・トランスファー(Representational State Transfer)(REST)サービス340とを含み得る。いくつかの実施形態において、システム300は、サーバ等の同一のコンピューティングシステム上で実現されてもよい。

20

30

【0072】

1つ以上のGUI画面のための1つ以上の画像(またはスケッチ)等のGUI設計文書を受けると、GUI画面の各画像は、クライアントサブシステム310上のアップローダ302によってサーバサブシステム320にアップロードすることができる。GUI画面の画像は、RESTサービス340を介してサーバサブシステム320に送信することができ、RESTサービス340は、GUI画面の画像をモデル生成システム330に送ることにより、サーバサブシステム320上でモデル生成プロセスを開始させることができる。モデル生成システム330は、GUI画面の画像、および、機械学習サブシステム360が生成した機械学習ベースのモデルに基づいて、GUIモデルを生成するように構成されている、モデル生成器350を含み得る。

40

【0073】

機械学習サブシステム360は、GUI画面の画像のような、GUI設計文書に基づいて、GUIモデルを生成するために使用される、機械学習ベースのモデルを生成し得る。機械学習ベースのモデルは、たとえば、UIコンポーネント検出器および/または分類器およびテキスト検出器および/または文字分類器を含み得る。いくつかの実施形態において、機械学習ベースのモデルは、画像ファイル内のオブジェクトを識別し分類することができる畳み込みニューラルネットワーク(CNN)等のニューラルネットワークベースの分類器を含み得る。

50

【 0 0 7 4 】

機械学習ベースのモデルは、教師あり学習技術および注釈付き訓練サンプル 3 6 8 を用いて訓練することができる。注釈付き訓練サンプル 3 6 8 は、ボタン、テキストフィールド、テキスト入力ボックス、ドロップダウンリスト、ドロップダウンメニュー、アイコン、表、円グラフ、棒グラフなどのような、各種 UI コンポーネントを含む画像を含み得る。注釈は、画像内の各 UI コンポーネントの位置、UI コンポーネントのタイプもしくはクラス、または UI コンポーネントに対応付けられたアクションのような、各 UI コンポーネントを一意的に識別するラベルまたはタグを含み得る。いくつかの実施形態において、注釈は画像ファイルにメタデータとして格納されてもよい。いくつかの実施形態において、注釈は画像ファイルとは別に格納されてもよい。いくつかの実施形態において、訓練サンプル内の画像はテキストコンテンツを含まなくてもよい。たとえば、訓練サンプルを生成するために使用される画像内のテキストコンテンツアイテムを、識別し、ブラックボックスまたはホワイトボックスのような予め定められたパターンに置き換えてもよい。

10

【 0 0 7 5 】

注釈付き訓練サンプル 3 6 8 は、特徴抽出エンジン 3 6 6 に送ることができる。特徴抽出エンジン 3 6 6 は、たとえば、深層 CNN を実現することにより、さまざまなコンポーネントの画像から異なるレベルの特徴を取り出すことができる。これに対し、従来のコンピュータビジョンベースの技術は、特徴工学の面倒なタスクを要し得るものであり、特定の UI コンポーネントについて明確な特徴を手作業で識別し、サポートベクターマシン (support vector machine) (SVM) または浅層ニューラルネットワークのような分類器に送ることができる。

20

【 0 0 7 6 】

特徴抽出エンジン 3 6 6 は、畳み込み、非線形 (または活性化) 関数 (たとえば ReLU)、および / またはプーリング (またはサブサンプリング) 等の動作を、深層 CNN の異なるレイヤの入力画像に対して実行することができる。CNN における畳み込み演算を用いることにより、入力画像 (たとえば訓練画像または GUI 画面画像) から特徴を抽出することができる。畳み込みにおいて、(フィルタ、カーネル、または特徴検出器と呼ばれる) 行列を、入力画像 (または処理された画素アレイもしくは特徴マップ) の上で、(ストライドと呼ばれる) 特定のステップでスライドさせることができる。すべての位置 (またはステップ) で、入力画像内のフィルタ行列と重複した行列との間の要素ごとの操作を計算し合計することにより、畳み込み特徴または特徴マップと呼ぶことができる、出力行列の 1 要素を表す最終値を得ることができる。

30

【 0 0 7 7 】

活性化関数 (たとえば正規化線形ユニット (ReLU)) を用いる、追加の非線形演算を、すべての畳み込み演算後に用いてもよい。ReLU の目的は非線形性を CNN に導入することである。tanh またはシグモイド関数のようなその他の非線形関数も使用できるが、ReLU がほとんどの状況においてより性能がよいことがわかっている。

【 0 0 7 8 】

(サブサンプリングまたはダウンサンプリングとも呼ばれる) 空間プーリングは、各特徴マップの次元を低減する一方で、最も重要な情報を保持することができる。特に、プーリングは、特徴次元をより小さくかつより管理し易くすることができ、ネットワーク内のパラメータおよび計算の数を減じることができる。また、プーリングは、ネットワークを、入力画像内の小さな変形、歪み、および移動に対して不変にすることができ、入力画像内の小さな歪みはプーリングの出力を変化させないであろう。なぜなら、ローカルな近隣の最大値または平均値を使用するからである。よって、プーリングは、入力画像の同変表現を得るのに役立つものであり、画像内の物体は、どこに位置していても検出できる。空間プーリングは、最大プーリング、平均プーリング、合計プーリングなどのような異なるやり方で実行してもよい。

40

【 0 0 7 9 】

入力画像 (たとえば訓練画像または GUI 画面画像) から特徴を抽出するために、先ず

50

畳み込みレイヤを用いて、1つ以上のフィルタで入力画像を畳み込むことにより、画像内のエッジまたは傾斜の検出のような、入力画像の1つ以上の特徴マップを生成することができる。活性化関数は非線形性を特徴マップに導入することができる。任意で、プーリングレイヤが特徴マップをダウンサンプリングすることにより、特徴マップの次元を低減してもよい。畳み込み、非線形（または活性化）関数、および任意のプーリングまたはサブサンプリング動作を再び、前のレイヤが生成した特徴マップに対して実行することにより、異なるレベルで特徴マップを抽出することができ、前のレイヤからの特徴マップを畳み込みの入力として使用することができる。このようにして、複数レベルの特徴マップを抽出して使用することにより、GUI画面画像または訓練画像内のUIコンポーネントを検出し分類することができる。

10

【0080】

次に、機械学習ベースのモデルを、抽出された特徴および注釈付き訓練サンプルの注釈を用いて訓練することができる。いくつかの実施形態において、SSD技術（たとえば、Liu et al., “SSD: Single Shot MultiBox Detector,” arXiv:1512.02325v5, Dec. 29, 2016参照）またはYOLO技術（たとえば、e.g., Redmon et al., “YOLO9000: Better, Faster, Stronger,” arXiv:1612.08242v1, Dec. 25, 2016参照）のようなオブジェクト検出技術を用い、オブジェクトを訓練画像のさまざまな異なる位置に、抽出した特徴を用いてローカライズすることができる。たとえば、深層CNNにおける分類レイヤ（全結合レイヤなど）は、抽出した特徴を用いることにより、訓練画像内に存在するUIコンポーネントを検出し分類することができる。分類結果は、訓練サンプルに対応付けられた注釈と比較することができる。分類結果が注釈に一致しない場合、フィードバックを分類レイヤおよび/または特徴抽出レイヤ（たとえば畳み込みレイヤ）に与えることにより、たとえば、バックプロパゲーション技術を用いて、深層CNNのパラメータを調整することができる。上記訓練プロセスは、各訓練サンプルごとに繰り返してもよい。深層CNNは、注釈付き訓練サンプルに基づいて、自身でUIコンポーネントのさまざまなタイプを区別するために使用できる一組の特徴を学習できるので、特徴抽出の面倒なタスクは回避できる。訓練プロセス後に、訓練されたモデル364を生成することができ、これは訓練サンプル内にあるUIコンポーネントを正確に検出し分類することができるであろう。機械学習ベースのモデルを生成するためのより詳細な処理は、以下でたとえば図5に関連して説明する。

20

30

【0081】

モデル生成器350は、光学特徴認識（OCR）モジュール352を含み得る。OCRモジュール352は、GUI画面の画像内のテキストコンテンツを検出し認識する、および/またはテキストコンテンツをGUI画面画像から分離することができる。いくつかの実施形態において、OCRモジュール352は、機械学習ベースのモデルを使用することにより、テキストコンテンツを検出し認識することができる。OCRモジュール352の出力は、テキストコンテンツアイテムと、それらの位置と、テキストコンテンツなしの画像とを含み得る。以下で詳細に述べるように、OCRモジュール352からの出力画像において、GUI画面画像内にテキストコンテンツアイテムが存在する領域は、ブラックボックス等の何等かの予め定められたパターンを含み得る。OCRモジュール352の出力はオブジェクト検出モジュールに送ればよい。OCRモジュール352のより詳細な動作は以下でたとえば図7との関連で説明する。

40

【0082】

オブジェクト検出モジュール354は、機械学習サブシステム360が生成した機械学習ベースのモデルを用いてOCRモジュール352からの出力画像内のUIコンポーネントを検出し分類することができる。たとえば、機械学習ベースのモデルは、UIコンポーネント検出器および/または分類器を含み得る。いくつかの実施形態において、機械学習ベースのモデルは、画像内のオブジェクトを識別し分類することができる、畳み込みニューラルネットワークのようなニューラルネットワークベースの分類器を含み得る。オブジェクト検出モジュール354は、機械学習ベースのモデルを用いることにより、GUI画

50

面画像内のUIコンポーネントを識別し、各UIコンポーネントを分類することにより、UIコンポーネントのタイプと対応付けられたアクションまたは機能を実現することができる。出力検出モジュール354の出力は、テキストコンテンツアイテム（およびそれらの位置情報）、ならびに位置、タイプ（もしくはクラス）、対応付けられたアクションまたは機能を含み得る。

【0083】

いくつかの実施形態において、オブジェクト検出モジュール354の出力はクラスタリングモジュール356に送ることができる。クラスタリングモジュール356は、検出したUIコンポーネントおよびテキストコンテンツアイテムを、たとえば、UIコンポーネントおよび/またはテキストコンテンツアイテムローカル情報、UIコンポーネントのタイプなどに基づいて、グルーピングすることができる。たとえば、グルーピングは、コンポーネント間の距離および/または類似性に基づいていてもよい。いくつかの実施形態において、クラスタリングモジュール356は、一組のルールを用いてグルーピングを実行し得る。たとえば、テキストコンテンツアイテムを、テキストボタンではなくテキスト入力UIコンポーネントとともにグルーピングしてもよい。別の例として、ボタンおよびデートピッカー（date picker）コンポーネント（たとえばカレンダー）はともにグルーピングしなくてもよく、レイアウトにおいて個別の要素とみなしてもよい。いくつかの実施形態において、グルーピングに使用される一組のルールは、メモリ122内の参照情報121の一部として格納されるなど、パーシステントメモリデバイス内のデータ構造（たとえば表またはリスト）に格納してもよい。

10

20

【0084】

クラスタリングモジュール356は、ボトムアップでグルーピングを再帰的に実行することができる。たとえば、個々のコンポーネント（たとえばUIコンポーネントおよび/またはテキストコンテンツアイテム）をグルーピングしてサブグループにすることができ、次に、このサブグループをグルーピングして高レベルのサブグループにし、1つのグループが画像内のすべてのコンポーネントをカバーするまで、続けることが判明した。

【0085】

いくつかの実施形態において、テキストコンテンツアイテム、UIコンポーネント、テキストコンテンツアイテムおよびUIコンポーネントに対応付けられた情報（たとえば位置、タイプなど）、およびクラスタマップ（またはレイアウト）を、テキスト解析モジュール358に送ることができる。テキスト解析モジュール358は、テキストコンテンツアイテムを解析することにより、GUI画面画像内のクリック可能なテキストコンテンツアイテムを識別できる。クリック可能なテキストコンテンツアイテムは、何等かのアクションまたは機能を示すことができ、通常は、少なくとも1つの動詞（たとえば、取消、保存、クリアなど）を含み得るものであり、どのUIコンポーネントにも対応付けられないであろうものである。たとえば、他のテキストコンテンツアイテムから分離された「取消」というテキストを有するテキストコンテンツアイテムは、クリック可能なテキストであることを意図し得る。いくつかの実施形態において、このようなクリック可能な単語の辞書を管理しテキスト解析に使用してもよい。

30

【0086】

いくつかの実施形態において、OCRモジュール352の出力は、個々の単語をそれぞれの位置情報とともに含み得る。クラスタリングモジュール356によるグルーピングの後に、個々の単語は、ラインまたはパラグラフの一部になり得る、または、GUI画面画像上の独立したアイテムになり得る。クラスタリングモジュール356は、個々の単語を、これらの単語がラインまたはパラグラフの一部である場合に、グルーピングすることができる。クリック可能なテキストコンテンツアイテムは、独立したアイテムであるかまたは特定数までの単語を有するラインに含まれるがパラグラフの一部である単語からのものではない単語から、識別することができる。いくつかの実施形態において、クリック可能と識別されたテキストコンテンツアイテムのタイプを、テキストから、「クリック可能なテキスト」に変更することができる。

40

50

【0087】

クリック可能なテキストコンテンツアイテムを識別しそれに対応付けられたタイプを変更するためのテキスト解析後に、クラスタマップをアップデートしメタデータ生成器362に与えてもよい。メタデータ生成器362は、1つ以上のGUI画面を含み得るGUIのGUIモデルを生成し得る。GUIモデルは、サーバサブシステム320に提供されるGUI画面画像の最適表現であってもよい。いくつかの実施形態において、メタデータ生成器362は、GUIモデルを、JavaScriptオブジェクト表記(JSON)フォーマットのような言語から独立したデータ交換フォーマットで、生成することができる。次に、GUIモデル(たとえばJSONメタデータで記述されている)を、クライアントサブシステム310からの要求に対する応答として、RESTサービス340を介してクライアントサブシステム310に送ることができる。

10

【0088】

GUIモデル306の受信後に、クライアントサブシステム310は、GUIモデル(たとえばJSONメタデータ)をページ生成器308に送ることができる。ページ生成器308は、図1に関連して先に述べたようにコード生成器(たとえばコード生成器126、128、または130)を含み得る。ページ生成器308は、GUIモデル306を入力として取り込み、ターゲットプラットフォーム、たとえばiOS(登録商標)またはAndroid(登録商標)を用いて動作するモバイルデバイスや、iOS(登録商標)、Windows(登録商標)、またはLinux(登録商標)を用いて動作するワイド画面を有するシステムに対し、対象言語でGUIを実現するコードを生成することができる。たとえば、コードはAngularJSまたはBootstrapに対して生成することができる。GUIインプリメンテーションは、1つ以上のプロセッサが実行可能なGUIの実行可能なインプリメンテーションであってもよい。いくつかの実施形態において、GUIインプリメンテーションをコンパイルする(または解釈する、またはそれに対して実行される解釈されたまたはその他何等かの処理が実行された)ことにより、GUIの事項可能バージョンを生成することができる。GUIに対して生成されるページアーティファクト312を次にエンドユーザが使用できるようにする。

20

【0089】

いくつかの場合において、モデル生成システム330は、GUI画面画像内のUIコンポーネントを正確に検出または分類しなくてもよい。フィードバックプロセスを用いることにより、モデル生成システム330にフィードバックを与え、生成されたGUIモデルを訂正することができる。たとえば、開発者は、GUIモデル306をレビューし、未検出のまたは誤分類されたUIコンポーネントを特定し、特定した未検出のまたは誤分類されたUIコンポーネントの画像と正しいラベルとを、任意のフィードバックモジュール304を介して提供することができる。たとえば、モデル生成システム330(より具体的にはオブジェクト検出モジュール354)によって誤分類されたもしくはまだ検出されていないUIコンポーネントがある場合、誤分類されたもしくは未検出のUIコンポーネントの画像または誤分類されたもしくは未検出のUIコンポーネントを含むGUI画面画像を、誤分類されたまたは未検出のUIコンポーネントの正しいラベルとともに、RESTサービス340を介してモデル生成システム330に与えることができる。フィードバック情報を、特徴抽出エンジン366が使用して、上述のように誤分類されたもしくは未検出のUIコンポーネントまたはGUI画面画像から特徴(たとえば特徴マップまたは特徴ベクトル)を抽出してもよい。いくつかの実施形態において、ユーザフィードバックから抽出される特徴を、特徴クラスタリングモジュール372に送ってもよい。

30

40

【0090】

特徴クラスタリングモジュール372は、訓練サンプル内のUIコンポーネントの各々の特徴を、多次元空間であってもよい特徴空間内のデータポイントにマッピングすることができる。一組のクラスタを特徴空間内に形成してもよく、各クラスタはUIコンポーネントのタイプに対応し得る。クラスタ間変化(または距離)は、UIコンポーネントをこれらの特徴を用いて適切に分類できる、重要であり得る。誤分類されたまたは未検出のU

50

UIコンポーネントのフィードバック画像から抽出された特徴も特徴空間内のデータポイントにマッピングできる。誤分類されたまたは未検出のUIコンポーネントの画像から抽出された特徴を表すデータポイントと、一組のクラスタのクラスタ中心との距離を計算することにより、誤分類されたまたは未検出のUIコンポーネントと訓練サンプルに既に含まれているUIコンポーネントとの間の類似性または相違性を求めることができる。これらの距離のうちの1つがしきい値未満である場合、誤分類されたまたは未検出のUIコンポーネントの画像を訓練サンプルに追加し、機械学習ベースのモデルの再訓練をトリガしてもよい。いくつかの実施形態において、このしきい値は実験結果に基づいて決定してもよい。

【0091】

しきい値は、注釈付き訓練サンプル368の特定のセキュリティレベルを保証するのに役立ち得る。たとえば、いくつかの場合において、開発者が「円グラフ」を誤って「棒グラフ」としてラベル付けすることがある。上述のようにフィードバックにおいて開発者が提供する注釈付きサンプルをフィルタリングするための距離が計算されていない場合、機械学習ベースのモデルを、間違っラベル付けされたサンプルで再訓練してもよい。これは、再訓練されたモデルを、前のモデルよりも悪化させる可能性がある、または、訓練プロセスを収束させない可能性がある。よって、特徴クラスタリングモジュール372および距離比較ブロック370は、ユーザが提供したサンプルのUIコンポーネントが既に訓練サンプル内にあるいくつかのUIコンポーネントと似ている場合は専らユーザが提供したサンプルを用いてモデルを再訓練することを保証するのに役立ち得る。いくつかの実施形態において、計算した距離がしきい値を上回る場合、ユーザが提供したサンプルは無視すればよい。いくつかの実施形態において、人間による介入は、計算した距離がしきい値を上回る場合に要求すればよい。

【0092】

次に、ユーザが提供したサンプルを用いて再訓練されたモデルを用いて、現在のGUI画面画像およびその後受けるGUI画面画像内のUIコンポーネントを検出し分類することができる。GUIモデルを、再訓練されたモデルを用いた検出および分類の結果に基づいてアップデートし、GUIインプリメンテーションを生成するためにクライアントサブシステム310に送り返すことができる。

【0093】

図1または図3に記載のシステムを用いることにより、機械学習ベースのモデルを訓練し、このモデルを用いてGUIモデルおよび/またはGUIインプリメンテーションを自動的に生成することができる。

【0094】

図4は、特定の実施形態に係る、機械学習ベースの分類器を用いてGUIモデルおよび/またはGUIの1つ以上のインプリメンテーションを生成するために実行されるハイレベル処理を示す簡略化されたフローチャート400である。図4に示される処理は、それぞれのシステムの1つ以上の処理ユニット（たとえばプロセッサ、コア）によって実行されるソフトウェア（たとえばコード、命令、プログラム）、ハードウェア、またはその組み合わせで実現することができる。ソフトウェアは非一時的な記憶媒体（たとえばメモリデバイス）に格納されていてもよい。図4に示され以下で説明される方法は、限定ではなく例示を意図している。図4は特定のシーケンスまたは順序で発生する各種処理ステップを示すが、これは限定を意図したものではない。特定の代替実施形態において、ステップは異なる順序で実行されてもよく、いくつかのステップは並列に実行されてもよい。特定の実施形態において、図4に示される処理は、上記モデル生成システム102または300を含むサーバ、クライアントコンピュータシステム、プロセッサ（たとえばグラフィック処理ユニット（graphic processing unit）（GPU）またはテンソル処理ユニット（tensor processing unit）（TPU））、ASIC、またはFPGAによって実行されてもよい。

【0095】

10

20

30

40

50

図4に示されるように、この処理は、モデル生成システム、より具体的には機械学習サブシステム（たとえばモデル生成システム330内の機械学習サブシステム360）が、GUI画面画像内のUIコンポーネントを検出し分類するための分類器のような機械学習ベースのモデルを構築することができる402から始めることができる。機械学習ベースの分類器を構築するために、サポートベクターマシン（SVM）分類器または畳み込みニューラルネットワーク（CNN）分類器のようなタイプの分類器を先ず選択することができる。分類器のいくつかのパラメータ、たとえばSVMのカーネルのタイプまたはCNN分類器のレイヤの数ならびにフィルタの数およびサイズを決定することができる。次に、（既知のタイプのUIコンポーネントの訓練画像、または既知のタイプのUIコンポーネントを有するGUI画面の訓練画像のような）訓練サンプルを、選択した分類器タイプおよび/または分類器のパラメータに基づいて、処理することができる。いくつかの実装例において、訓練および推論プロセスの精度および速度を改善するために、訓練および/または推論プロセスの開始前に、テキスト情報を抽出して取り除いてもよい。既知のタイプのUIコンポーネント（注釈が示される）を有する処理後の訓練画像を用いて分類器を訓練することができる。

10

20

30

40

50

【0096】

たとえば、分類器のパラメータ（たとえば分類器内の畳み込みレイヤの各種重みまたはフィルタ）を（たとえば何等かのランダムまたはデフォルト値）に初期化し、各訓練画像ごとに、分類器が訓練画像から特徴を抽出し、抽出した特徴に基づいて訓練画像を分類することができる。損失関数値（たとえばエラー）を、訓練画像および分類結果における既知のタイプのUIコンポーネントに基づいて求めることができる。分類器のパラメータを調整することにより、損失関数値を低減することができる。いくつかの実装例において、訓練画像を分類器を用いて分類することは、訓練画像から特徴を抽出すること、たとえばフィルタを用いて訓練画像を畳み込むことにより特徴マップを生成すること、または画像の特定のヒストグラム（たとえばカラーヒストグラムまたは傾斜のヒストグラム）から特徴を抽出すること、および、訓練画像の抽出した特徴に基づいて訓練画像内のUIコンポーネントのタイプを判断することを、含み得る。機械学習ベースのモデルの構築のさらなる詳細は、以下、図5との関連で説明する。

【0097】

404で、モデル生成システムは、機械学習ベースのモデルを用いてGUIのためのGUIモデルを生成することができる。たとえば、いくつかの実装例において、入力されたGUI画面画像の各々を処理することにより、テキストコンテンツアイテムおよび個々のUIコンポーネントを含むGUIコンポーネントを特定して抽出し、テキストコンテンツアイテムおよびUIコンポーネントのパラメータ（たとえばサイズ、位置、色など）を判断し、機械学習ベースのモデルを用いてUIコンポーネントを分類することにより、UIコンポーネントのタイプを判断することができる。次に、分類したUIコンポーネントおよびテキストコンテンツアイテムをグルーピングすることにより、GUIコンポーネントの階層を形成することができる。GUI画面のレイアウト（またはクラスタマップ）をこの階層に基づいて判断することができる。いくつかの実施形態において、クリック可能なテキストコンテンツアイテムを特定してもよい。次に、GUIモデルを生成することにより、GUI画面、GUI画面内のUIコンポーネント、GUI画面内のテキストコンテンツアイテムなどのレイアウトを記述することができる。いくつかの実装例において、GUIモデルは、JavaScriptオブジェクト表記（JSON）フォーマットのような、言語から独立したデータ交換フォーマットで記述されてもよい。次に、404で生成されたGUIモデルを、さまざまなダウンストリームの目的のために使用することが可能である。

【0098】

たとえば、406で、GUIモデルに基づいて、GUIの1つ以上のGUI画面の1つ以上のインプリメンテーションを自動的に生成することができる。いくつかの実施形態において、その上でGUIを使用し得るプラットフォームおよびこのインプリメンテーショ

ンのターゲット言語に基づいて、コード生成テンプレートを、利用できるコード生成テンプレート（たとえば図1に関して説明したコード生成テンプレート140）から選択することができる。コード生成器（たとえばコード生成器126、128、または130）は、選択したコード生成テンプレートおよび図1に関して先に述べたGUIモデルに基づいて、GUIを実現するためのコードを生成することができる。特定の実施形態において、このインプリメンテーションは、1つ以上のプロセッサが実行可能なGUIの実行可能なインプリメンテーションであってもよい。他の実施形態において、このインプリメンテーションは、GUIを実現するためのコードおよびロジックを含み得る。次に、このインプリメンテーションをコンパイルまたは解釈することにより、実行可能なGUIを生成することができる。

10

【0099】

いくつかの実施形態において、408で、モデル生成システムは、404で生成したGUIモデルに関するユーザフィードバックを受けることができる。たとえば、GUIモデル内で誤分類されたまたは未検出のUIコンポーネントがある場合、誤分類されたまたは未検出のUIコンポーネントの画像、または、誤分類されたまたは未検出のUIコンポーネントを含むGUI画面画像（および画像内の誤分類されたまたは未検出のUIコンポーネントの位置）を、ユーザから提供された誤分類されたまたは未検出のUIコンポーネントについての注釈とともに、モデル生成システムに提供することができる。誤分類されたまたは未検出のUIコンポーネントの画像および注釈を用いることにより、機械学習ベースのモデルを再訓練するおよび/またはGUIモデルを再度生成することができる。

20

【0100】

図5は、特定の実施形態に係る、ユーザインターフェイスコンポーネントを検出し分類するための機械学習ベースの分類器を訓練するために実行されるハイレベル処理を示す簡略化されたフローチャート500である。特定の実施形態において、図5に示される処理は、図4の402で実行される処理の一部として実行することができる。図5に示される処理は、それぞれのシステムの1つ以上の処理ユニット（たとえばプロセッサ、コア）によって実行されるソフトウェア（たとえばコード、命令、プログラム）、ハードウェア、またはその組み合わせで、実現し得る。ソフトウェアは（たとえばメモリデバイス上の）非一時的な記憶媒体に格納されていてもよい。図5に示され以下で説明される方法は、例示を意図しているの限定しているのではない。図5は特定のシーケンスまたは順序で発生する各種処理ステップを示しているが、これは限定を意図している訳ではない。特定の代替実施形態において、これらのステップは何等かの異なる順序で実行されてもよく、または、いくつかのステップが並列に実行されてもよい。特定の実施形態において、図5に示される処理は、モデル生成システム102、機械学習サブシステム360、プロセッサ（たとえばGPUもしくはTPU）、ASIC、またはFPGAによって実行されることができる。

30

【0101】

502で、1つ以上の注釈付き訓練画像を受け、たとえばメモリ122に格納することができる。訓練画像は、ボタン、テキスト入力ボックス、ドロップダウンリスト、ドロップダウンメニュー、チェックボックス、ラジオボタン、スイッチボタン、アイコン、表、（人々または物体の）写真、ラインディバイダ、コンテナなどのような、既知のタイプの各種UIコンポーネントを含み得る。訓練画像には、UIコンポーネントのタイプ、および、訓練画像内のUIコンポーネントの位置等の、UIコンポーネントに対応付けられたその他の情報を用いて、注釈を付けることができる。いくつかの実施形態において、各訓練画像は1つのUIコンポーネントの画像を含むだけであってもよい。いくつかの実施形態において、各訓練画像は複数のUIコンポーネントを含んでいてもよい。いくつかの実施形態において、各訓練画像は、テキストコンテンツを含むまたは含まない場合があるGUI画面の画像に対応していてもよい。いくつかの実施形態において、訓練GUI画面画像は、GUIを設計するために使用し得る各種フロントエンド技術にまたがる各種ソースからのものであってもよい。

40

50

【0102】

各訓練画像を処理することにより、この訓練画像内の1つ以上のUIコンポーネントを抽出し分類することができる。訓練画像がUIコンポーネント以外の特徴を含むいくつかの実施形態において、訓練プロセスの精度および速度を改善するために、訓練のためのUIコンポーネントの検出および分類に干渉する可能性が高い視覚的特徴を一時的に消去してもよい。たとえば、下記の504および506における動作について説明するように、1つ以上のUIコンポーネントを検出し分類する前に、GUI画面に対応する各訓練画像ごとに、テキスト情報を抽出し訓練画像から削除してもよい。各訓練画像ごとに、画像内の特徴を抽出し、抽出した特徴に基づいて訓練された機械学習ベースの分類器を用いて訓練画像内のUIコンポーネントを検出し分類することができる。分類器を、分類結果と訓練画像の注釈との比較に基づいて調整してもよい。

10

【0103】

任意で、504で、(もしあれば)テキスト情報を訓練画像から抽出してもよい。テキスト情報の抽出は、訓練画像内のテキストコンテンツアイテムを検出することと、抽出したテキストコンテンツアイテムを認識することとを含み得る。たとえば、いくつかの実施形態において、全層畳み込みネットワーク(fully convolutional network)(FCN)モデル(たとえば、Zhou, et al., "EAST: An Efficient and Accurate Scene Text Detector," arXiv:1704.03155v2, July 10, 2017参照)を用いて、訓練画像内のテキストコンテンツアイテムを検出し、テキストコンテンツアイテムの位置(たとえばバウンディングボックス(bounding box)の座標)および/またはテキストコンテンツアイテムの寸法を求めることができる。いくつかの実施形態において、設計ツールの拡張(たとえばGIMPまたはPhotoshop)を用いて、訓練画像のレイヤからテキスト領域を抽出してもよい。次に、求めた位置におけるサブ画像を抽出してもよく、光学文字認識(OCR)プロセスを抽出したサブ画像各々に対して実行することにより、テキストコンテンツアイテムに対応付けられたテキスト情報を抽出することができる。いくつかの実施形態において、OCRプロセスはまた、機械学習ベースのモデルを使用することにより、テキストコンテンツアイテム内の文字を認識することができる。たとえば、いくつかの場合において、テキストコンテンツアイテムは従来のOCR技術では正確に認識されないかもしれない手書き文字を含み得る。機械学習ベースの分類器を用いて、各文字を一組の既知の文字のうちの1つに分類することができる。

20

30

【0104】

506で、訓練画像内の(もしあれば)テキスト情報を、予め定められたパターンのようなプレースホルダに置き換えてもよい。たとえば、サブ画像内の画素値を、背景画素値または予め定められた画素値に置き換えて、続くUIコンポーネント検出および分類にテキストコンテンツアイテムが干渉しないようにすることができる。いくつかの実装例において、プレースホルダは訓練のための特徴の役割も果たし得る。たとえば、プレースホルダは、元のUIコンポーネントが特定のテキスト情報を含み得るものでありしたがってクリック可能ボタンまたはテキスト入力ボックスのような特定のタイプのUIコンポーネントのうちの1つである可能性があることを、示すことができる。加えて、UIコンポーネント内のテキストコンテンツアイテムの配列はプレースホルダによって保存することができる。

40

【0105】

508で、訓練画像から特定の特徴を抽出することができる。たとえば、特徴抽出エンジン366について先に述べたように、深層CNNを用いて、訓練画像から異なるレベルにある各種特徴(たとえば特徴ベクトルまたは特徴マップ)を抽出してもよい。異なる特徴は異なるフィルタを用いて抽出することができる。たとえば、フィルタを用いることにより、訓練画像を、訓練画像内のオブジェクトのエッジまたは輪郭を含む特徴マップに畳み込むことができる。抽出すべき特徴のタイプは訓練プロセス中に自動的に決定することができ、よって、異なるタイプのUIコンポーネントを、抽出した特徴に基づいて、正確に検出し分類することができる。

50

【0106】

510で、SVM分類器またはCNN分類器のような機械学習ベースの分類器は、508で抽出した特徴に基づいて、訓練画像内のUIコンポーネントを検出し分類することができる。各種実施形態において、UIコンポーネントの検出およびUIコンポーネントの分類は、異なる機械学習ベースのモデルを用いて別々に実行してもよく、または、同一の分類器を用いて合同で実行してもよい。いくつかの実施形態において、まず、UIコンポーネントの各々の境界を検出できる各種輪郭検出技術を用いて、訓練画像内のUIコンポーネントを特定し訓練画像から抽出し、次に、サブ画像内の抽出したUIコンポーネントを分類することができる。たとえば、ニューラルネットワークは、オブジェクト検出技術（たとえば上記SSDまたはYOLO技術）を実現することにより、抽出した特徴を用いて、1つ以上のUIコンポーネントを訓練画像の1つ以上の異なる位置にローカライズすることができる。各UIコンポーネントの位置または境界に基づいて、各UIコンポーネントの境界内のサブ画像を抽出してもよい。次に、UIコンポーネントを含む訓練画像の、判断した位置から抽出したサブ画像を、分類器によるUIコンポーネントの分類に使用してもよい。いくつかの実施形態において、同じ深層ニューラルネットワークは、合同UIコンポーネント検出および分類を実行してもよく、抽出された特徴を用いて、UIコンポーネントと訓練画像の1つ以上の異なる位置にローカライズするとともに1つ以上のUIコンポーネントをパイプラインまたは並列処理で分類してもよい。いくつかの実施形態において、機械学習ベースのモデルを訓練することによりGUI画面画像内の複数のUIコンポーネントを検出するために、少なくともいくつかの訓練画像は異なる位置の複数のUIコンポーネントを含み得る。

10

20

【0107】

UIコンポーネントの分類中に、訓練画像から抽出した特定の特徴を組み合わせて使用することにより、あるタイプのUIコンポーネントを他のタイプのUIコンポーネントから区別することができる。先に述べたように、テキスト情報のような干渉する特徴を、潜在的に区別がつく特徴（たとえば予め定められたパターン）と置き換えることができるので、訓練処理をより速くすることができ、より正確な分類器をもたらしすることができる。UIコンポーネントを分類するために使用できる数種類の機械学習ベースの分類器が存在し得る。以下、UIコンポーネントを分類するために使用し得る分類器のいくつかの例について説明する。

30

【0108】

いくつかの実装例において、サポートベクターマシン（SVM）分類器を用いてUIコンポーネントを分類することができる。SVM分類器はデシジョン境界を定めるデシジョン面に基づく。SVM分類器は、線形分類器（すなわちラインを用いて一組のオブジェクトをそれぞれのクラスまたはグループに分離する分類器）であってもよく、または、非線形カーネル関数に基づくより複雑な構造（超平面）を用いて異なるクラスのオブジェクトを分離する非線形分類器であってもよい。いくつかの実施形態において、画像の画素ではなく、画像のヒストグラム特徴（たとえばカラーヒストグラム、傾斜のヒストグラム）を、SVM分類器を用いる分類に使用してもよい。

【0109】

先に述べたように、各種ニューラルネットワークベースの分類器を用いることによりUIコンポーネントを分類することもできる。たとえば、ニューラルネットワークベースの分類器は、特徴抽出用のいくつかのレイヤと分類用のいくつかのレイヤ（たとえば全結合レイヤ）とを含み得る。特徴抽出用のレイヤは、特徴抽出エンジン366について先に述べた特徴抽出レイヤと同様であってもよい。全結合レイヤでは、あるレイヤ内のすべてのノードが隣接するレイヤ上のすべてのノードに接続される。全結合レイヤは、入力画像の抽出された特徴を、入力として用いて、入力画像を出力レイヤの各種クラスに分類ことができ、出力レイヤ上の各ノードは、UIコンポーネントの可能なクラス（またはタイプ）に対応し得る。たとえば、全結合レイヤは、出力レイヤ上でソフトマックス活性化関数を使用してもよい。ソフトマックス関数は、任意の実数値スコアのベクトルを得て、合

40

50

計 1 になる 0 と 1 との間の値 (確率) を有するベクトルに変換することができる。次に、入力画像を、出力レイヤにおける可能な各クラスに対応付けられた確率値に基づいて分類することができる。

【 0 1 1 0 】

5 1 2 で、分類器 (たとえば CNN ベースの分類器) を、分類結果と訓練画像の注釈との比較に基づいて調整することができる。一般的に、分類器のパラメータ (たとえばフィルタの重みおよび全結合レイヤの重み) は、まず、何等かのランダム値、デフォルト値、または同様の分類器で使用される値で、初期化することができる。重みのような分類器のパラメータはランダム値を有し得るものであり訓練例について最適化されないため、出力確率はランダムであろう。いくつかの実施形態において、可能な各クラスについての分類エラーは、分類結果と訓練画像の注釈との比較に基づいて判断することができ、出力レイヤにおける総エラーは、可能なすべてのクラスの確率エラーを合計することによって計算できる。次に、バックプロパゲーション技術のような技術を用いて、すべてのパラメータを基準とするエラーの傾斜と、ネットワークで訓練される重みとを計算することができ、出力エラーを最小にするために、傾斜降下を用いてネットワーク内で訓練するすべてのパラメータおよび重みをアップデートすることができる。重みは総エラーに対するその貢献度に従って調整することができる。同じ訓練画像が再び入力にある場合、出力確率はターゲット確率により近くなるであろう。このことは、ネットワークがこの特定の画像の分類を学習したことを示す。上記訓練プロセスは、訓練サンプル内のおすべての訓練画像に対して繰り返すことができる。先に述べたように、フィルタの数、フィルタサイズのような畳み込みニューラルネットワークのいくつかのパラメータ、ニューラルネットワークのアーキテクチャなどは、予め選択されており訓練プロセス中は不変であってもよい。

10

20

【 0 1 1 1 】

訓練プロセスの最後において、分類器のすべてのパラメータは最適化されており訓練サンプルからの訓練画像を正確に分類することができる。まだ見られていない画像 (たとえばテスト画像または新たな場増) が分類器に入力されると、分類器は、フォワードプロパゲーションステップを経て、訓練された重みおよびパラメータを用いて各クラスの確率を出力することができ、これは、訓練プロセスとの対比で推論プロセスと呼ぶことができる。訓練サンプルが十分である場合、訓練された分類器は、まだ見られていないサンプルを正しいクラスに分類することができる。

30

【 0 1 1 2 】

先に述べたように、いくつかの実施形態において、分類器はユーザフィードバックに基づいて再訓練することができる。たとえば、ユーザは、機械学習ベースの分類器を用いてモデル生成システムが生成した GUI モデルをレビューしてもよい。分類器が誤分類したまたはまだ検出していない UI コンポーネントがある場合、誤分類されたもしくは未検出の UI コンポーネントの画像、または、誤分類されたもしくは未検出の UI コンポーネントを含む GUI 画像を、ユーザが提供する誤分類されたまたは未検出の UI コンポーネントについての正しい注釈とともに、モデル生成システムにフィードバックしてもよい。

【 0 1 1 3 】

5 1 4 で、機械学習サブシステム 3 6 0 のような機械学習サブシステムは、ユーザフィードバックに基づく分類器の再訓練が必要か否かを判断してもよい。たとえば、フィードバックにおける画像の特徴を、5 0 8 で述べたように抽出し、多次元特徴空間内のデータポイントにマッピングしてもよい。この特徴空間は、既に訓練サンプル内にある異なるタイプの UI コンポーネントに対応する一組のクラスタを含み得る。誤分類されたまたは未検出の UI コンポーネントから抽出された特徴を表すデータポイントと、一組のクラスタのクラスタ中心との間の距離を計算することにより、誤分類されたまたは未検出の UI コンポーネントと、訓練サンプルに含まれる UI コンポーネントとの類似点または相違点を判断することができる。距離のうちの 1 つがしきい値未満の場合、誤分類されたまたは未検出の UI コンポーネントの画像と、ユーザが提供した注釈とを、信頼できるとみなしてもよく、訓練サンプルに追加して機械学習ベースのモデルの再訓練をトリガしてもよい。

40

50

計算した距離がしきい値を上回っている場合、ユーザが提供する画像および注釈は無視してもよい、または、人間による介入を要求してもよい。

【0114】

514で、ユーザフィードバックにおいて提供された画像および注釈が信頼できると判断された場合、分類器をユーザフィードバックの基づいて再訓練してもよい。ユーザフィードバックに基づいて分類器を再訓練するための動作は、508、510、および512に関して先に述べた動作と同様であってもよい。機械学習ベースのモデルが訓練された後に、モデル生成システムは、機械学習ベースのモデルを用い、ユーザが提供するGUI画面画像に基づいてGUIモデルを生成することができる。GUIを実現するためのソースコードも、人間による介入を最小にして自動的に生成することができる。

10

【0115】

図6は、特定の実施形態に係る、GUIの設計情報に基づいてグラフィックユーザインターフェイスのためのGUIモデルおよび/またはソースコードを生成するために実行されるハイレベル処理を示す、簡略化されたフローチャート600である。図6に示される処理は、1つ以上の処理ユニット（たとえばプロセッサ、コア）によって実行されるソフトウェア（たとえばコード、命令、プログラム）、ハードウェア、またはその組み合わせで実現することができる。ソフトウェアはメモリ（たとえばメモリデバイスのような非一時的なコンピュータ読取可能記憶媒体）に格納されていてもよい。図6に示される特定の一連の処理ステップは限定を意図しているのではない。これらの処理ステップは図6に示される順序と異なる順序で実行されてもよく図6に示されるすべてのステップを実行する必要のある訳ではないことがわかる。いくつかの実施形態において、図6に示される処理のうちの少なくとも一部を、図1に示されるモデル生成システム102または図3に示されるモデル生成システム330が実行してもよい。

20

【0116】

簡潔にするために、図6および以下の関連する説明に示されるフローチャート600は、GUIのための1つのGUI画面画像に対する処理について説明している。しかしながらこれは限定を意図しているのではない。代替実施形態において、複数のGUI画面画像を用いてGUIのための複数のGUI画面について説明することができる。いくつかのこのようなシナリオにおいて、図6に示される処理（またはこの処理の一部）を、GUIのための各GUI画面ごとに繰り返してもよい。

30

【0117】

602で、1つ以上のGUI画面画像を含む入力画像を受け取ることができる。各GUI画面画像は、1つ以上のUIコンポーネントと、テキスト情報を含む1つ以上のテキスト領域とを含み得る。まず、各GUI画面画像を検出し入力画像から抽出することができる。先に述べたように、GUI画面画像は、ユーザインターフェイスの設計者が、顧客またはクライアント調査、マーケティング調査、ならびに、GUIのルック・アンド・フィールおよび開発するGUIに含まれる機能を伝える情報のその他のソースに基づいて生成し得るものである。GUI画面画像は、コンピュータ支援設計ツールを用いて生成されてデジタルフォーマットで保存されてもよく、または、紙の上のスケッチとして手作業で生成されスキャンされてデジタル画像にされてもよい。いくつかの実施形態において、GUI画面画像は、いくつかの修正が施されたGUIの前のバージョンのスナップショットであってもよい。特定の実施形態において、クライアントデバイス108（またはクライアントサブシステム310）によって実行されるプログラムは、ユーザがモデル生成システム102（またはモデル生成システム330）に対してインターフェイスしやり取りすることを可能にし得る。たとえば、プログラムは、ユーザ（たとえばプログラマー）がソフトウェアアプリケーションを構築するための設備およびツールを提供する統合開発環境（IDE）の一部であってもよい。このような環境において、IDEは、UI開発者が設計したGUI画面画像をユーザがインポート（たとえばドラッグ・アンド・プレース）することを可能にするツールを提供することができる。

40

【0118】

50

先に述べたように、モデル生成システムが、G U I画面の機能およびレイアウト、G U I画面のコンポーネント、U Iコンポーネントのタイプおよび機能、U Iコンポーネント間のマッピングなどを決定できるようにするために、G U IのためのG U I画面画像が、各種解析を受けてもよい。

【0119】

604で、テキストコンテンツを含むテキスト領域を検出しG U I画面画像から抽出することができる。各テキスト領域内のテキストコンテンツは1つ以上のテキストコンテンツアイテムを含み得る。いくつかの実施形態において、深層畳み込みネットワークモデルを用いて、G U I画面画像内のテキスト領域を検出し、テキスト領域の位置（たとえばバウンディングボックスの座標）および/または寸法を求めることができる。いくつかの実施形態において、設計ツールの拡張（たとえばG I M PまたはP h o t o s h o p）を用いて、訓練画像のレイヤからテキスト領域を抽出してもよい。次に、求めた位置におけるサブ画像を抽出しテキストコンテンツアイテムの抽出に使用してもよい。

10

【0120】

606で、サブ画像から、テキストコンテンツアイテムおよびテキスト領域内の対応する位置を抽出することができる。たとえば、いくつかの実施形態において、光学文字認識（O C R）プロセスを、抽出した各サブ画像に対して実行することにより、各テキストコンテンツアイテムに対応付けられたテキスト情報を抽出することができる。いくつかの実施形態において、O C Rプロセスはまた、機械学習ベースのモデルを用いることにより、テキストコンテンツアイテム内の文字を認識することもできる。たとえば、いくつかの場合において、テキストコンテンツアイテムは、従来のO C R技術では正しく認識されない可能性がある手書き文字を含むことがある。機械学習ベースの分類器を用いることにより、各文字を、一組の既知の文字のうちの1つに分類することができる。テキストコンテンツアイテムの検出および認識についてのさらなる詳細については以下で図7との関連で説明する。

20

【0121】

608で、G U I画面画像内のテキスト領域をプレースホルダに置き換えることができる。たとえば、サブ画像内の画素値を、背景画素値または予め定められた画素値に置き換えて、続くU Iコンポーネントの抽出および分類にテキスト領域が干渉しないようにすることができる。先に述べたように、いくつかの実装例において、プレースホルダは分類のための特徴の役割も果たし得る。たとえば、プレースホルダは、元のU Iコンポーネントが特定のテキスト情報を含み得るものでありしたがってクリック可能ボタンまたはテキスト入力ボックスのような特定のタイプのU Iコンポーネントのうちの1つである可能性があることを、示すことができる。加えて、U Iコンポーネント内のテキストコンテンツアイテムの配列はプレースホルダによって保存することができる。いくつかの実装例において、U Iコンポーネントに重なるまたはU Iコンポーネントに隣接するテキストのみをプレースホルダに置き換えてもよい。

30

【0122】

610で、510に関して先に述べた処理と同様のやり方で、U Iコンポーネントを検出することができる。G U I画面画像内の対応する位置を求めることができる。U Iコンポーネントは、たとえば、ボタン、チェックボックス、リスト、テキスト入力ボックス、アイコン、コンテナ、ラジオボタン、スイッチボタンなどを含み得る。各種輪郭検出技術を用いることにより、各U Iコンポーネントの境界を検出することができる。たとえば、先に述べたように、ニューラルネットワークは、G U I画面画像から特徴を抽出することができ、オブジェクト検出技術（たとえば上記S S DまたはY O L O技術）を実現することにより、抽出された特徴を用いて、1つ以上のU IコンポーネントをG U I画面画像の1つ以上の異なる位置にローカライズすることができる。各U Iコンポーネントの位置または境界に基づいて、各U Iコンポーネントの境界内のサブ画像をG U I画面画像から抽出することができる。

40

【0123】

50

612で、610で検出したUIコンポーネントを、機械学習ベースの分類器およびサブ画像を用いて分類することができる。先に述べたように、機械学習ベースの分類器は、たとえば、既知のタイプのUIコンポーネントを含む訓練画像を用いて訓練することができるSVMまたはCNN分類器を含み得る。分類器は、UIコンポーネントが属しているであろうUIコンポーネントのタイプを判断することができる。いくつかの実施形態において、各タイプのUIコンポーネントを、このタイプのUIコンポーネントの、たとえば画像または画像のソース、機能、対応するアクション、対応するコード、他のアプリケーションまたはGUI画面へのリンクなどを記述し得るデータモデルに対応付けることができる。先に述べたように、いくつかの実施形態において、610および612における動作は、同じ深層CNNベースの分類器によって実行されてもよい。

10

【0124】

614で、610で検出され612で分類されたUIコンポーネント、および604で検出され606で抽出されたテキストコンテンツアイテムをグルーピングすることにより、GUI画面のレイアウトを生成することができる。先に述べたように、クラスタリングモジュール（たとえばクラスタリングモジュール356）は、特定の予め定められたルールに基づいてボトムアップでグルーピングを再帰的に実行することができる。テキストコンテンツアイテムおよびUIコンポーネントの属性、たとえば位置、サイズ、タイプ、機能等を、グルーピングに使用することができる。たとえば、個々のコンポーネント（たとえばUIコンポーネントおよび/またはテキストコンテンツアイテム）をサブグループ（たとえば表、リスト、組織図、パラグラフなどを表す）にグルーピングすることができ、次にサブグループをより高いレベルのサブグループにグルーピングすることができ、これは、1つのグループが画像内のすべてのコンポーネントをカバーするまで行われる。たとえば、606で抽出されたテキストコンテンツアイテムは、個々の単語をその位置情報とともに含み得る。クラスタリングモジュールは、個々の単語を、これらの単語がラインまたはパラグラフの一部であれば、グルーピングすることができる。異なるレベルでのグルーピングに基づいて、クラスタリングモジュールは、階層および位置情報を用い、画像内のコンポーネントの階層、および、GUI画面の最適なレイアウトを決定することができる。

20

【0125】

616で、どのUIコンポーネントにも対応付けられていないクリックブルテキストコンテンツアイテムを特定することができる。クリックブルテキストコンテンツアイテムは、何等かのアクションまたは機能を示し得るものであり、通常は少なくとも1つの動詞（たとえば取消、保存、クリアなど）を含むが、どのUIコンポーネントにも対応付けられていない場合がある。たとえば、他のテキストコンテンツアイテムから分離されている「取消」というテキストを有するテキストコンテンツアイテムは、クリックブルテキストを意図し得るものであり、このテキストコンテンツアイテムがクリックされるとアクションが実行され得る。よって、GUIモデルは、テキストコンテンツアイテムをクリック可能であると説明しクリックブルテキストコンテンツアイテムに対応付けられたリンクまたはコードを提供することができる。いくつかの実施形態において、クリックブルテキストコンテンツアイテムは、独立したアイテムであるかまたは特定数までの単語を有するラインに含まれるがパラグラフの一部である単語からのものではない単語から、識別することができる。いくつかの実施形態において、このようなクリックブルテキストコンテンツアイテムのタイプを、「テキスト」から「クリック可能テキスト」に変更してもよい。

30

40

【0126】

618で、テキストコンテンツアイテムおよび対応する位置、分類されたUIコンポーネントおよび対応する位置、ならびにGUI画面のレイアウトに基づいて、GUIのためのGUIモデルを生成することができる。特定の実施形態において、GUIモデルは、604、606、610、612、614、および616で実行された処理に関する情報を格納することができる。たとえば、GUIモデルは、604および606で決定されたGUI画面内のテキストコンテンツアイテム、610で決定されたUIコンポーネントの位

50

置、612で決定されたUIコンポーネント（およびUIコンポーネントに対応付けられたデータモデル）のタイプ、614で決定されたGUI画面のレイアウトなどを格納することができる。GUIモデルに格納された情報は、ダウンストリームコンシューマが使用して、GUIのインプリメンテーションを生成することができる。いくつかの実装例において、GUIモデルはGUIのためのメタデータとして生成されてもよい。いくつかの実装例において、GUIモデルは、JavaScriptオブジェクト表記（JSON）フォーマットのような言語から独立したデータ交換フォーマットで記述されてもよい。

【0127】

620で、GUIモデルに基づいて、GUIを実現するためのソースコードを自動的に生成することができる。いくつかの実施形態において、GUIを実現するためのソースコードは、特定のコード生成テンプレートに基づいて生成することができる。たとえば、各種コード生成器アプリケーション（たとえばコード生成器126、128、および130）は、GUIモデルを入力として取り込み、場合によっては異なるプログラミング言語でおよび/または異なるプラットフォームのために、たとえば異なるプログラミング言語および/または異なるプラットフォームのためのコード生成テンプレート140に基づいて、GUIを実現するためのコードを生成することができる。

10

【0128】

コード生成器は、このコード生成器に固有の言語でGUIを実現するコードを生成することができる。インプリメンテーションは、1つ以上のプロセッサが実行可能なGUIの実行可能なインプリメンテーションであってもよい。たとえば、コード生成器は、GUIモデルを入力として取り込み、第1のプラットフォーム（たとえばiOS（登録商標））のためのインプリメンテーションを第1の言語で生成することができる。別のコード生成器システムは、第1のプラットフォームのための同じGUIモデルを用いて第2の言語でGUIのインプリメンテーションを生成することができ、第3のコード生成器は、Android（登録商標）プラットフォームのための同じGUIモデルを用いてGUIの別のインプリメンテーションを生成することができる。コード生成テンプレートは、コンピュータシステムの1つ以上のプロセッサが実行することが可能な実行可能GUIを生成するためにコンパイルまたは解釈することが可能なハイレベルコード（方法、機能、クラス、イベントハンドラなどを含み得る）を含む1つ以上のソースコードから構成されていてもよい。

20

30

【0129】

このようにして、GUIの実行可能なインプリメンテーションを、GUIモデルに基づいて自動的に生成することが可能であり、この実行可能なインプリメンテーションは、GUI設計情報に記述されている、GUIのルック・アンド・フィール、GUIの機能、およびUIコンポーネントに対応するデータモデルに対するマッピングまたは関係を、カプセル化する。

【0130】

いくつかの実施形態において、開発者はさらに、追加コードを用いてコードテンプレートインプリメンテーションをさらに拡張することにより、コードベースを完成または向上させる（たとえばコードベースに追加機能を加える）ことができる。たとえば、コード生成器は、GUIモデルに対応するマークアップコード（たとえばXML）を含む1つ以上のファイルを受け、マークアップコードを（ハイレベル）ソースコード（たとえばJava、C++、またはその他の言語）に変換することにより1つ以上のソースコードを含むGUIインプリメンテーションを出力するように、構成することができる。次に、コードインプリメンテーションをコンパイルする（または解釈するまたはこれに対してその他何かの処理を実行する）ことにより、GUIの実行可能なバージョンを生成することができる。いくつかの実施形態において、GUIモデル（たとえばJSONフォーマット）を用いて、マークアップ言語（たとえばHTMLもしくはXML）またはスタイルシート言語（たとえばカスケーディングスタイルシート（CSS））のような異なるプログラミング言語でコードを生成することができる。

40

50

【 0 1 3 1 】

先に述べたように、ほとんどのOCR技術の短所は、暗い背景上に色が薄いテキストが存在するときに、たとえば青色の背景に白色のテキストがあるときに、テキストコンテンツを検出しないことである。特定の実施形態に従うと、全層畳み込みネットワークモデル（たとえば上記EAST技術）を用いて、GUI画面画像内のテキストコンテンツアイテムを検出し、テキストコンテンツアイテムの位置（たとえばバウンディングボックスの座標）および/または寸法を求めることができる。次に、求めた位置のサブ画像を抽出することができ、改良されたOCRプロセスを、抽出した各サブ画像に対して実行することにより、各テキストコンテンツアイテムに対応付けられたテキスト情報を抽出することができる。

10

【 0 1 3 2 】

図7は、特定の実施形態に係る、GUI画面の画像からテキストコンテンツアイテムを抽出するために実行されるハイレベル処理を示す簡略化されたフローチャート700である。フローチャート700に示される処理は、たとえばOCRモジュール352によって実行されてもよく、606および504に関して先に述べたように、GUI画面画像または訓練画像からテキスト情報を抽出するために使用することができる。図7に示される処理は、1つ以上の処理ユニット（たとえばプロセッサ、コア）によって実行されるソフトウェア（たとえばコード、命令、プログラム）、ハードウェア、またはその組み合わせで、実現することができる。ソフトウェアは、（たとえばメモリデバイスのような非一時的なコンピュータ読取可能記憶媒体上の）メモリに格納されていてもよい。図7に示される特定の一連の処理ステップは限定を意図したものではない。なお、これらの処理ステップは、図7に示される順序と異なる順序で実行されてもよく、図7に示されるすべてのステップを実行する必要はある訳ではない。

20

【 0 1 3 3 】

710で、入力画像（たとえばGUI画面画像）をテキスト抽出モジュール（たとえばOCRモジュール352）に与えることができる。720で、入力画像内のテキスト領域を、たとえば上記EAST技術を用いて検出することができる。EAST技術は、入力画像内においてテキストコンテンツアイテムが存在する領域の境界を示す座標の正確な推定値を提供することができる。730で、テキストコンテンツアイテムに対応する各領域のサブ画像を抽出することができる。いくつかの実施形態において、サブ画像は、テキストコンテンツアイテムに対応する領域に追加されたマージンを含み得る。各サブ画像を、780で実行されるOCRプロセスの前に、705で前処理することにより、サブ画像内のテキストコンテンツを認識することができる。

30

【 0 1 3 4 】

いくつかの実施形態において、705における前処理は、740で適応しきい値処理を用いてRGBフォーマットのサブ画像を2値サブ画像に変換することを含み得る。たとえば、画素の強度値がしきい値よりも高い場合、画素の値を、白色画素を表す値（たとえば「1」）に設定してもよい。そうでなければ、画素の値を、黒色画素を表す値（たとえば「0」）に設定してもよい。750で、2値サブ画像についてヒストグラムを生成することができる。760で、テキスト抽出モジュールは、各画素の強度値を相補値に変更することによってサブ画像を反転させる必要があるか否かを判断することができる。一般的に、テキストに対応する画素の数は、各サブ画像内の背景に対応する画素の数よりも少ない可能性がある。よって、ヒストグラムが、2値サブ画像内の黒色画素の数が白色画素の数よりも多いことを示す場合、テキストは、背景よりも高い（すなわち明るい）強度値を有する可能性があるため、RGBサブ画像を反転させる必要があるであろう。そうでなければ、テキストは背景よりも低い（暗い）強度値を有する可能性があるため、RGBサブ画像を反転させる必要はないであろう。いくつかの実施形態において、ヒストグラムをRGBサブ画像について生成し、テキストが背景よりも明るいかが否かを判断するために使用してもよい。770で、RGBサブ画像を反転させることができる。たとえば、最大強度値が255で各画素の強度値が x_i の場合、反転画像内の各画素の値は $255 - x_i$ になる

40

50

であろう。

【0135】

780で、RGBサブ画像または反転されたRGBサブ画像を処理することにより、サブ画像内のテキストコンテンツアイテムを認識することができる。RGBサブ画像または反転されたRGBサブ画像内のテキストは背景よりも暗いので、テキストコンテンツアイテムをより正確に認識できる。先に述べたように、いくつかの実施形態において、機械学習ベースの技術を用いることにより、たとえば手書き文字を含み得るテキストコンテンツアイテムを認識することができる。790で、各テキストコンテンツアイテム内の認識したテキストと、各テキストコンテンツアイテムの位置（たとえばバウンディングボックスの座標）を、記憶装置に格納するか、または、モデル生成器（モデル生成器120または350）における次の処理モジュールに与えることができる。先に述べたように、いくつかの実施形態において、サブ画像をブレースホルダに置き換えることによりテキストコンテンツアイテムを取り除いてもよい。よって、その結果得られた画像を、テキストコンテンツを含まないようにすることができ、UIコンポーネントの検出および分類のために次の処理モジュールに与えることができる。

10

【0136】

図8は、特定の実施形態に係る、UIコンポーネントおよび/またはテキストコンテンツアイテムをクラスタリングすることにより、グラフィックユーザインターフェイス画面について生成することができるUIコンポーネント階層800の一例を示す。UIコンポーネント階層800は、UIコンポーネントおよび/またはテキストコンテンツアイテムのクラスタリングに基づいて生成することができ、GUI画面の最適なレイアウトを決定するために使用することができる。いくつかの実施形態において、UIコンポーネント階層800は、先に述べたようにGUI画面においてUIコンポーネントが如何にして階層的に組織されているかを記述することができる。図8に示される例において、GUI画面810はKのクラスタ820を含み得る。各クラスタ820は1つ以上のUIコンポーネント830を含み得る。たとえば、クラスタ1はMのUIコンポーネント830を含んでいてもよく、クラスタ2はNのUIコンポーネント830を含んでいてもよく、...、クラスタKはLのUIコンポーネント830を含んでいてもよい。各UIコンポーネント830は、先に述べたように判断することができる、対応付けられた属性を有し得る。たとえば、各UIコンポーネント830は、先に述べたように、対応付けられたID、位置およびサイズ情報、機械学習ベースの分類器によって分類されたUIコンポーネントのタイプ、（もしあれば）対応付けられたテキスト、ならびにUIコンポーネントに関するその他の情報、たとえば画像または画像のソース、機能、対応するアクション、対応するコード、他のアプリケーションまたはGUI画面へのリンクなどを有し得る。各UIコンポーネントごとの、これらの対応付けられた属性を、モデル生成器（たとえばモデル生成器120または350）が用いてGUIモデルを生成することができる。

20

30

【0137】

以下の説明では特定の実施形態について述べる。これらの特定の実施形態は専ら例示を目的としているのであって、限定を意図したものではない。これらの実施形態のうちの一部において、本明細書に開示される自動GUIモデルおよびコード生成技術を、Oracle（登録商標）Visual Builder Cloud Service開発環境のような統合開発環境（IDE）の一部として実現することができる。いくつかの実施形態において、本明細書に開示される自動GUIモデルおよびコード生成技術は、スタンドアロンツールとして実現してもよい。

40

【0138】

図9は、特定の実施形態に係る、GUI画面のためのGUIモデルおよびGUIコードを自動的に生成するためのソフトウェアアプリケーションのGUI画面900の一例を示す。図9に示されるように、IDEにおいて最小限の新規ウェブアプリケーションwebappl 910をスタートすることができる。この最小限の新規ウェブアプリケーションwebappl 910は、スタートページ940のみを含み得る。webappl

50

9 1 0 のファイル構造 9 2 0 が左側のパネルに示される。新規 GUI を追加するには、UI 開発者等のユーザが、「+」ボタン 9 3 0 をクリックすればよい。

【 0 1 3 9 】

図 1 0 は、特定の実施形態に係る、GUI 画面のための GUI モデルおよび GUI コードを自動的に生成するためのソフトウェアアプリケーションの GUI 画面 1 0 0 0 の一例を示す。GUI 画面 1 0 0 0 は、ユーザが「+」ボタン 9 3 0 をクリックした後にユーザに対して表示されるユーザインターフェイス 1 0 1 0 を示す。ユーザインターフェイス 1 0 1 0 は、GUI 設計ファイルを入力ボックス 1 0 2 0 に入力またはアップロードするようユーザを促すことができる。GUI 設計ファイルは、GUI 設計者がコンピュータ上または紙の上で設計しデジタルフォーマットで保存した、画素またはベクトルを含む画像ファイルのような 1 つ以上の画像ファイルを含み得る。入力ボックス 1 0 2 0 が選択される（たとえばクリックされる）と、ユーザがローカルまたは遠隔保存されている GUI 設計ファイルの位置を指定できるように、ポップアップウィンドウ 1 0 3 0 がユーザに対して表示されてもよい。

10

【 0 1 4 0 】

図 1 1 は、特定の実施形態に係る、GUI 画面のための GUI モデルおよび GUI コードを自動的に生成するためのソフトウェアアプリケーションの GUI 画面 1 1 0 0 の一例を示す。図 1 1 に示されるように、ユーザは、ポップアップウィンドウ 1 1 3 0 から GUI 設計ファイル（たとえば画像ファイル）の位置を指定した後に、GUI 設計ファイルを選択する、または、入力ボックス 1 1 2 0 へのアイコン 1 1 1 0 によって示されるように選択した GUI 設計ファイルをドラッグすることができる。

20

【 0 1 4 1 】

図 1 2 は、特定の実施形態に係る、GUI 画面のための GUI モデルおよび GUI コードを自動的に生成するためのソフトウェアアプリケーションの GUI 画面 1 2 0 0 の一例を示す。図 1 2 に示されるように、GUI 設計ファイルがアプリケーションにアップロードされた後に、スタートページ 1 2 4 0 に加えて新たなページ 1 2 1 0 が生成されてもよい。ファイル構造 1 2 2 0 に示されるように、新たな GUI のための、JSON ファイル 1 2 1 6、Java Script ファイル 1 2 1 4、および HTML ファイル 1 2 1 2 が生成されてもよい。JSON ファイル 1 2 1 6 は GUI モデルを含んでいてもよく、HTML ファイル 1 2 1 2 は html フォーマットの GUI のインプリメンテーションを含んでいてもよい。

30

【 0 1 4 2 】

図 1 3 は、特定の実施形態に係る、GUI 画面のために生成された JSON フォーマットの GUI モデル 1 3 0 0 の一例を示す。GUI モデル 1 3 0 0 は、図 1 2 に示される JSON ファイル 1 2 1 6 に対応し得る。図 1 3 に示されるように、GUI モデル 1 3 0 0 は、GUI 画面のレイアウト、ラベル、画像、およびテキスト入力ボックス等の GUI 画面の各種コンポーネントを記述することができる。GUI モデル 1 3 0 0 は、いくつかの UI コンポーネント内にテキストを表示することもできる。ユーザは、必要であればテキストエディタを用いて GUI モデルを修正することができる。

40

【 0 1 4 3 】

図 1 4 は、特定の実施形態に係る、本明細書に開示される技術に基づいて生成されるコードを用いて表示される、入力 GUI 画面画像 1 4 1 0 および対応する GUI 画面画像 1 4 2 0 の一例を示す。入力 GUI 画面画像 1 4 1 0 は、設計された GUI 画面を示すフォトビューアによって表示される JPG ファイルである。GUI 画面画像 1 4 2 0 は、HTML ファイル 1 2 1 2 のような HTML ファイルに基づいてウェブブラウザによって表示される。

【 0 1 4 4 】

図 1 5 は、特定の実施形態に係る、本明細書に開示される技術に基づいて入力 GUI 画面の一例に対して生成される html コードを示すスクリーンショット 1 5 0 0 の一例である。図 1 5 に示されるように、ファイル構造 1 5 1 0 内のツールによって生成された H

50

T M Lファイル1520をユーザが選択すると、ソースHTMLコードがユーザに対して表示されてもよく、ユーザは必要に応じてHTMLコードを修正してもよい。

【0145】

図16Aおよび図16Bは、特定の実施形態に係る、本明細書に開示される技術に基づいて生成されるコードを用いて表示される、入力GUI画面画像1600および対応するGUI画面画像1650の別の例を示す。入力GUI画面画像1600は、設計されたGUI画面を示すフォトビューアによって表示されるPNGファイルである。GUI画面画像1650は、本明細書に開示される技術を実現するツールによって自動的に生成されるHTMLファイルに基づいてウェブブラウザによって表示されるGUIのスナップショットである。

10

【0146】

本明細書に記載のモデル生成システム102または330は、クラウド環境（たとえばプライベート、パブリック、およびハイブリッドクラウド環境を含む各種クラウド）、オンプレミス環境、ハイブリッド環境などを含む、さまざまな異なる環境で実現することができる。

【0147】

図17は、ある実施形態を実現するための分散型システム1700の簡略図を示す。示されている実施形態において、分散型システム1700は、サーバ1712を介して1つ以上の通信ネットワーク1710に結合された1つ以上のクライアントコンピューティングデバイス1702、1704、1706、および1708を含む。クライアントコンピューティングデバイス1702、1704、1706、および1708は、1つ以上のアプリケーションを実行するように構成し得る。

20

【0148】

各種実施形態において、サーバ1712は、あるアプリケーションのユーザインターフェイスコンポーネントと、データオブジェクトとの間に、このアプリケーションに関する要求情報を用いて、関係またはマッピングを自動的に生成する1つ以上のサービスまたはソフトウェアアプリケーションを実行または提供するのに適するようにすることができる。たとえば、特定の実施形態において、サーバ1712は、あるアプリケーションについての要求情報を受けてもよく、この要求情報は、アプリケーションの所望の機能の記述を含み得るものであり、また、アプリケーションの所望のGUIの画像を含み得る。そうすると、サーバ1712は、この要求情報を用いることにより、アプリケーションのユーザインターフェイスコンポーネントと1つ以上のデータオブジェクトとの間にマッピングを生成することができる。

30

【0149】

特定の実施形態において、サーバ1712はまた、非仮想または仮想環境に提供し得るその他のサービスまたはソフトウェアアプリケーションを提供することができる。いくつかの実施形態において、これらのサービスは、サービスとしてのソフトウェア（Software as a Service）（SaaS）モデルのようなウェブベースまたはクラウドサービスとして、クライアントコンピューティングデバイス1702、1704、1706、および/または1708のユーザに提供されてもよい。そうすると、クライアントコンピューティングデバイス1702、1704、1706、および/または1708を操作しているユーザは、1つ以上のクライアントアプリケーションを利用することにより、サーバ1712とやり取りして、これらのコンポーネントが提供するサービスを利用することができる。

40

【0150】

図17に示される構成において、サーバ1712は、サーバ1712が実行する機能を実現する1つ以上のコンポーネント1718、1720および1722を含み得る。これらのコンポーネントは、1つ以上のプロセッサ、ハードウェアコンポーネント、またはその組み合わせによって実行可能なソフトウェアコンポーネントを含み得る。なお、分散型システム1700とは異なり得る多種多様なシステム構成が可能であることが理解される

50

はずである。よって、図17に示される実施形態は、実施形態のシステムを実現するための分散型システムの一例であり限定を意図したものではない。

【0151】

本開示の教示に従い、ユーザは、クライアントコンピューティングデバイス1702、1704、1706、および/または1708を用いることにより、要求情報をサーバ1712に提供し、サーバ1712から情報(たとえばマッピング情報)を受け取ることができる。クライアントデバイスは、クライアントデバイスのユーザがクライアントデバイスとやり取りすることを可能にするインターフェイスを提供し得る。クライアントデバイスはまた、このインターフェイスを介して情報をユーザに出力する。図17は4つのクライアントコンピューティングデバイスしか示していないが、任意の数のクライアントコンピューティングデバイスをサポートすることができる。

10

【0152】

クライアントデバイスは、ポータブルハンドヘルドデバイス、パーソナルコンピュータおよびラップトップのような汎用コンピュータ、ワークステーションコンピュータ、ウェアラブルデバイス、ゲームシステム、シンクライアント、各種メッセージングデバイス、センサまたはその他のセンシングデバイスなどのような、さまざまな種類のコンピューティングシステムを含み得る。これらのコンピューティングデバイスは、さまざまな種類およびバージョンのソフトウェアアプリケーションおよびオペレーティングシステム(たとえばMicrosoft Windows(登録商標)、Apple Macintosh(登録商標)、UNIX(登録商標)またはUNIX系オペレーティングシステム、Linux(登録商標)またはLinux系オペレーティングシステム、たとえば、各種モバイルオペレーティングシステム(たとえばMicrosoft Windows Mobile(登録商標)、iOS(登録商標)、Windows Phone(登録商標)、Android(登録商標)、BlackBerry(登録商標)、Palm OS(登録商標))を含むGoogle Chrome(登録商標)OS)を含み得る。ポータブルハンドヘルドデバイスは、セルラードフォン、スマートフォン(たとえばiPhone(登録商標))、タブレット(たとえばiPad(登録商標))、携帯情報端末(PDA)などを含み得る。ウェアラブルデバイスは、Google Glass(登録商標)ヘッドマウントディスプレイおよびその他のデバイスを含み得る。ゲームシステムは、各種ハンドヘルドゲームデバイス、インターネット接続可能なゲームデバイス(たとえばKinect(登録商標)ジェスチャ入力デバイスを備えたMicrosoft Xbox(登録商標)ゲーム機、Sony PlayStation(登録商標)システム、Nintendo(登録商標)が提供する各種ゲームシステムなどを含み得る。クライアントデバイスは、各種インターネット関連アプリケーション、通信アプリケーション(たとえばEメールアプリケーション、ショートメッセージサービス(SMS)アプリケーション)のような多種多様なアプリケーションを実行可能であってもよく、各種通信プロトコルを使用することができる。

20

30

【0153】

通信ネットワーク1710は、利用可能な多様なプロトコルのうちのいずれかを用いてデータ通信をサポートできる、当該技術の当業者には周知のいずれかの種類のネットワークであればよく、上記プロトコルは、TCP/IP(伝送制御プロトコル/インターネットプロトコル)、SNA(システムネットワークアーキテクチャ)、IPX(インターネットパケット交換)、AppleTalk(登録商標)などを含むがこれらに限定されない。単に例として、通信ネットワーク1710は、ローカルエリアネットワーク(LAN)、Ethernet(登録商標)に基づくネットワーク、トークンリング、ワイドエリアネットワーク(WAN)、インターネット、仮想ネットワーク、仮想プライベートネットワーク(VPN)、イントラネット、エクストラネット、公衆交換電話網(PSTN)、赤外線ネットワーク、無線ネットワーク(たとえば電気電子学会(IEEE)802.11プロトコルスイートのいずれかの下で動作する無線ネットワーク、Bluetoothおよび/または任意の他の無線プロトコル)、および/またはこれらおよび/または他

40

50

のネットワークの任意の組み合わせを含み得る。

【0154】

サーバ1712は、1つ以上の汎用コンピュータ、専用サーバコンピュータ（一例としてPC（パーソナルコンピュータ）サーバ、UNIX（登録商標）サーバ、ミッドレンジサーバ、メインフレームコンピュータ、ラックマウント型サーバなどを含む）、サーバファーム、サーバクラスタ、またはその他の適切な構成および/または組み合わせで構成されてもよい。サーバ1712は、仮想オペレーティングシステムを実行する1つ以上の仮想マシン、または仮想化を伴う他のコンピューティングアーキテクチャを含み得る。これはたとえば、サーバに対して仮想記憶装置を維持するように仮想化できる論理記憶装置の1つ以上のフレキシブルツールなどである。各種実施形態において、サーバ1712を、

10

【0155】

サーバ1712内のコンピューティングシステムは、上記オペレーティングシステムのうちのいずれかを含む1つ以上のオペレーティングシステム、および、市場で入手可能なサーバオペレーティングシステムを実行し得る。また、サーバ1712は、HTTP（ハイパーテキスト転送プロトコル）サーバ、FTP（ファイル転送プロトコル）サーバ、CGI（コモンゲートウェイインターフェイス）サーバ、JAVA（登録商標）サーバ、データベースサーバなどを含むさまざまなさらに他のサーバアプリケーションおよび/または中間層アプリケーションのうちのいずれかを実行し得る。例示されるデータベースサーバは、Oracle（登録商標）、Microsoft（登録商標）、Sybase（登録商標）、IBM（登録商標）（International Business Machines）などから市場で入手可能なものを含むが、それらに限定されるものではない。

20

【0156】

いくつかの実装例において、サーバ1712は、クライアントコンピューティングデバイス1702、1704、1706および1708のユーザから受信したデータフィードおよび/またはイベントアップデートを解析および整理統合するための1つ以上のアプリケーションを含み得る。一例として、データフィードおよび/またはイベントアップデートは、センサデータアプリケーション、金融株式相場表示板、ネットワーク性能測定ツール（たとえば、ネットワークモニタリングおよびトラフィック管理アプリケーション）、

30

【0157】

分散型システム1700はまた、1つ以上のデータリポジトリ1714、1716を含み得る。特定の実施形態において、これらのデータリポジトリを用いてデータおよびその他の情報を格納することができる。たとえば、データリポジトリ1714、1716のうちの1つ以上を用いて、サーバ1712が使用するまたは生成した、1つ以上のアプリケーションについての要求情報、マッピング情報、およびその他の情報などの情報を格納することができる。データリポジトリ1714、1716は、さまざまな場所に存在し得る。たとえば、サーバ1712が使用するデータリポジトリは、サーバ1712のローカル位置にあってもよく、またはサーバ1712から遠隔の位置にあってもよく、ネットワークベースのまたは専用接続を介してサーバ1712と通信する。データリポジトリ1714、1716は、異なる種類であってもよい。特定の実施形態において、サーバ1712が使用するデータリポジトリは、データベース、たとえば、Oracle Corporation（登録商標）が提供するデータベースのようなリレーショナルデータベースで

40

50

あってもよい。これらのデータベースのうちの1つ以上を、SQLフォーマットのコマンドに応じて、データの格納、アップデート、およびデータベースとの間での取り出しを可能にするのに適するようにしてもよい。アプリケーションが使用するデータリポジトリは、たとえば、キー値ストアリポジトリ、オブジェクトストアリポジトリ、またはファイルシステムがサポートする汎用ストレージリポジトリのようなさまざまな種類のものであってもよい。

【0158】

特定の実施形態において、本開示に記載の特徴は、クラウド環境を介してサービスとして提供することができる。図18は、特定の実施形態に係る、各種サービスをクラウドサービスとして提供し得るクラウドベースのシステム環境の簡略化されたブロック図である。図18に示される実施形態において、クラウドインフラストラクチャシステム1802は、ユーザが1つ以上のクライアントデバイス1804、1806、および1808を用いて要求し得る1つ以上のクラウドサービスを提供することができる。クラウドインフラストラクチャシステム1802は、サーバ1712に関して先に述べたものを含み得る1つ以上のコンピュータおよび/またはサーバを含み得る。クラウドインフラストラクチャシステム1802内のコンピュータを、汎用コンピュータ、専用サーバコンピュータ、サーバファーム、サーバクラスタ、またはその他任意の適切な配置および/または組み合わせとして、組織してもよい。

10

【0159】

ネットワーク1810は、クライアントデバイス1804、1806、および1808と、クラウドインフラストラクチャシステム1802との間におけるデータの通信および交換を容易にすることができる。ネットワーク1810は、1つ以上のネットワークを含み得る。ネットワークは同じ種類であっても異なる種類であってもよい。ネットワーク1810は、通信を容易にするために、有線および/または無線プロトコルを含む、1つ以上の通信プロトコルをサポートすることができる。

20

【0160】

図18に示される実施形態は、クラウドインフラストラクチャシステムの一例にすぎず、限定を意図したものではない。なお、その他いくつかの実施形態において、クラウドインフラストラクチャシステム1802が、図18に示されるものよりも多くのまたは少ないコンポーネントを有していてもよく、2つ以上のコンポーネントを組み合わせてもよく、または、コンポーネントの異なる構成または配置を有していてもよいことが、理解されるはずである。たとえば、図18は3つのクライアントコンピューティングデバイスを示しているが、代替実施形態において任意の数のクライアントコンピューティングデバイスをサポートすることができる。

30

【0161】

クラウドサービスという用語は一般的に、サービスプロバイダのシステム(たとえばクラウドインフラストラクチャシステム1802)により、インターネット等の通信ネットワークを介してオンデマンドでユーザが利用できるようにされるサービスを意味するために使用される。典型的に、パブリッククラウド環境では、クラウドサービスプロバイダのシステムを構成するサーバおよびシステムは、顧客自身のオンプレミスサーバおよびシステムとは異なる。クラウドサービスプロバイダのシステムは、クラウドサービスプロバイダによって管理される。よって、顧客は、別途ライセンス、サポート、またはハードウェアおよびソフトウェアリソースをサービスのために購入しなくても、クラウドサービスプロバイダが提供するクラウドサービスを利用できる。たとえば、クラウドサービスプロバイダのシステムはアプリケーションをホストすることができ、ユーザは、アプリケーションを実行するためにインフラストラクチャリソースを購入しなくても、インターネットを介しオンデマンドでアプリケーションをオーダーし使用することができる。クラウドサービスは、アプリケーション、リソースおよびサービスに対する容易でスケーラブルなアクセスを提供するように設計される。いくつかのプロバイダがクラウドサービスを提供する。たとえば、ミドルウェアサービス、データベースサービス、Javaクラウドサービス

40

50

その他などのいくつかのクラウドサービスが、カリフォルニア州レッドウッド・ショアーズの Oracle Corporation（登録商標）から提供される。

【0162】

特定の実施形態において、クラウドインフラストラクチャシステム1802は、ハイブリッドサービスモデルを含む、サービスとしてのソフトウェア（SaaS）モデル、サービスとしてのプラットフォーム（PaaS）モデル、サービスとしてのインフラストラクチャ（IaaS）モデルその他のような、異なるモデルを使用して、1つ以上のクラウドサービスを提供し得る。クラウドインフラストラクチャシステム1802は、各種クラウドサービスのプロビジョンを可能にする、アプリケーション、ミドルウェア、データベース、およびその他のリソースのスイートを含み得る。

10

【0163】

SaaSモデルは、アプリケーションまたはソフトウェアを、インターネットのような通信ネットワークを通して、顧客が基本となるアプリケーションのためのハードウェアまたはソフトウェアを購入しなくても、サービスとして顧客に配信することを可能にする。たとえば、SaaSモデルを用いることにより、クラウドインフラストラクチャシステム1802がホストするオンデマンドアプリケーションに顧客がアクセスできるようにすることができる。Oracle Corporation（登録商標）が提供するSaaSサービスの例は、人的資源/資本管理のための各種サービス、カスタマー・リレーションシップ・マネジメント（CRM）、エンタープライズ・リソース・プランニング（ERP）、サプライチェーン・マネジメント（SCM）、エンタープライズ・パフォーマンス・マネジメント（EPM）、解析サービス、ソーシャルアプリケーションその他を含むがこれらに限定されるものではない。特定の実施形態において、アプリケーションに関する要求情報から機能アプリケーションを自動的に生成する機能を、SaaSモデルに基づくサービスとして提供することができる。

20

【0164】

IaaSモデルは一般的に、インフラストラクチャリソース（たとえばサーバ、ストレージ、ハードウェアおよびネットワークリソース）を、クラウドサービスとして顧客に提供することにより、柔軟な計算およびストレージ機能を提供するために使用される。各種IaaSサービスがOracle Corporation（登録商標）から提供される。

30

【0165】

PaaSモデルは一般的に、顧客が、アプリケーションおよびサービスを、環境リソースを調達、構築、または管理しなくても、開発、実行、および管理することを可能にするプラットフォームおよび環境リソースをサービスとして提供するために使用される。Oracle Corporation（登録商標）が提供するPaaSサービスの例は、Oracle Java Cloud Service（JCS）、Oracle Database Cloud Service（DBCS）、データ管理クラウドサービス、各種アプリケーション開発ソリューションサービスその他を含むがこれらに限定されるものではない。

【0166】

クラウドサービスは一般的に、オンデマンドのセルフサービスベースで、サブスクリプションベースで、柔軟にスケーラブルで、信頼性が高く、可用性が高い、安全なやり方で提供される。たとえば、顧客は、サブスクリプションオーダーを介し、クラウドインフラストラクチャシステム1802が提供する1つ以上のサービスをオーダーすることができる。クラウドインフラストラクチャシステム1802はそうすると、処理を実行することにより、顧客のサブスクリプションオーダーで要求されたサービスを提供する。たとえば、ユーザは、アプリケーションについての要求情報を、クラウドインフラストラクチャシステムに与え、クラウドインフラストラクチャシステムに、要求情報から機能アプリケーションを自動的に生成するよう要求する。これは、先に述べたように、アプリケーションのユーザインターフェイスコンポーネントとデータオブジェクトとの間にマッピングを自動的に生成することを含む。クラウドインフラストラクチャシステム1802を、1つま

40

50

たは複数のクラウドサービスを提供するように構成してもよい。

【0167】

クラウドインフラストラクチャシステム1802は、異なるデプロイメントモデルを介してクラウドサービスを提供し得る。パブリッククラウドモデルにおいて、クラウドインフラストラクチャシステム1802は、第三者クラウドサービスプロバイダによって所有されていてもよく、クラウドサービスは一般のパブリックカスタマーに提供される。このカスタマーは個人でも企業でもよい。その他特定の実施形態において、プライベートクラウドモデルでは、クラウドインフラストラクチャシステム1802がある組織内で（たとえば企業組織内で）機能してもよく、サービスはこの組織内の顧客に提供される。たとえば、この顧客は、人事部、給与部その他のようなさまざまな部署であってもよく、企業内の個人であってもよい。その他の特定の実施形態において、コミュニティクラウドモデルでは、クラウドインフラストラクチャシステム1802および提供されるサービスは、関連コミュニティ内の数組織で共有されてもよい。上記モデルの混成モデルのようなその他各種モデルを使用することもできる。

10

【0168】

クライアントデバイス1804、1806、および1808は、異なるタイプであってもよく（たとえば図18に示されるクライアントデバイス1804、1806、および1808）、1つ以上のクライアントアプリケーションを操作可能であってもよい。ユーザは、クライアントデバイスを用いることにより、クラウドインフラストラクチャシステム1802が提供するサービスを要求することなど、クラウドインフラストラクチャシステム1802とのやり取りを行うことができる。たとえば、ユーザは、クライアントデバイスを用いることにより、アプリケーションについての要求情報をクラウドインフラストラクチャシステム1802に提供し、クラウドインフラストラクチャシステムに、要求情報から機能アプリケーションを自動的に生成するよう要求することができる。これは、本開示で述べたように、アプリケーションのユーザインターフェイスコンポーネントとデータオブジェクトとの間にマッピングを自動的に生成することを含む。

20

【0169】

いくつかの実施形態において、クラウドインフラストラクチャシステム1802が、要求されたサービスを提供するために実行する処理は、ビッグデータ解析を含み得る。この解析は、大きなデータセットを使用し、解析し、操作することにより、このデータ内のさまざまな傾向、挙動、関係などを検出し可視化することを含み得る。この解析は、1つ以上のプロセッサが、場合によっては、データを並列に処理し、データを用いてシミュレーションを実行するなどして、実行してもよい。この解析に使用されるデータは、構造化データ（たとえばデータベースに格納されたまたは構造化モデルに従って構造化されたデータ）および/または非構造化データ（たとえばデータブロッブ(blob) (binary large object: バイナリ・ラージ・オブジェクト))を含み得る。

30

【0170】

図18の実施形態に示されるように、クラウドインフラストラクチャシステム1802は、クラウドインフラストラクチャシステム1802が提供する各種クラウドサービスのプロビジョニングを容易にするために利用されるインフラストラクチャリソース1830を含み得る。インフラストラクチャリソース1830は、たとえば、処理リソース、ストレージまたはメモリリソース、ネットワークリソースなどを含み得る。

40

【0171】

特定の実施形態において、異なる顧客に対しクラウドインフラストラクチャシステム1802が提供する各種クラウドサービスをサポートするためのこれらのリソースを効率的にプロビジョニングし易くするために、リソースを、リソースセットまたはリソースモジュール（「ポッド」とも呼ばれる）になるようにまとめてもよい。各リソースモジュールまたはポッドは、1種類以上のリソースを予め一体化し最適化した組み合わせを含み得る。特定の実施形態において、異なるポッドを異なる種類のクラウドサービスに対して予めプロビジョニングしてもよい。たとえば、第1のポッドセットをデータベースサービスの

50

ためにプロビジョニングしてもよく、第1のポッドセット内のポッドと異なるリソースの組み合わせを含み得る第2のポッドセットをJavaサービスなどのためにプロビジョニングしてもよい。いくつかのサービスについて、これらのサービスをプロビジョニングするために割り当てられたリソースをサービス間で共有してもよい。

【0172】

クラウドインフラストラクチャシステム1802自身が、クラウドインフラストラクチャシステム1802の異なるコンポーネントによって共有されクラウドインフラストラクチャシステム1802によるサービスのプロビジョニングを容易にするサービス1832を、内部で使用してもよい。これらの内部共有サービスは、セキュリティ・アイデンティティサービス、統合サービス、エンタープライズリポジトリサービス、エンタープライズマネージャサービス、ウィルススキャン・ホワイトリストサービス、高可用性バックアップリカバリサービス、クラウドサポートを可能にするサービス、Eメールサービス、通知サービス、ファイル転送サービスなどを含み得るが、これらに限定されるものではない。

10

【0173】

クラウドインフラストラクチャシステム1802は複数のサブシステムを含み得る。これらのサブシステムは、ソフトウェア、またはハードウェア、またはその組み合わせで実現し得る。図18に示されるように、サブシステムは、クラウドインフラストラクチャシステム1802のユーザまたは顧客がクラウドインフラストラクチャシステム1802とやり取りすることを可能にするユーザインターフェイスサブシステム1812を含み得る。ユーザインターフェイスサブシステム1812は、ウェブインターフェイス1814、クラウドインフラストラクチャシステム1802が提供するクラウドサービスが宣伝広告され消費者による購入が可能なオンラインストアインターフェイス1816、およびその他のインターフェイス1818のような、各種異なるインターフェイスを含み得る。たとえば、顧客は、クライアントデバイスを用い、クラウドインフラストラクチャシステム1802が提供する1つ以上のサービスを、インターフェイス1814、1816、および1818のうちの一つ以上を用いてサブスクリプションオーダーを行うことによって要求する(サービス要求1834)ことができる。これらのインターフェイスは、たとえば、顧客が、クラウドインフラストラクチャシステム1802が提供するクラウドサービスを特定するオンラインストアにアクセスしてこれをブラウズし、顧客が申し込むことを希望する、提供されたサービスのうちの一つ以上についてサブスクリプションオーダーを行うための、オンラインストアユーザインターフェイス1816を含み得る。その他のインターフェイスはウェブインターフェイス1814およびその他のインターフェイス1818を含み得る。

20

30

【0174】

特定の実施形態において、サービス要求またはサブスクリプション要求は、要求しているユーザまたは顧客を特定する情報と、顧客が申込みを希望する1つ以上のサービスとを含み得る。たとえば、顧客は、クラウドインフラストラクチャシステム1802が提供する自動アプリケーション生成サービスに対してサブスクリプションオーダーを行うことができる。

【0175】

図18に示される実施形態のような特定の実施形態において、クラウドインフラストラクチャシステム1802は、新規サブスクリプションオーダーを処理するように構成されたオーダー管理サブシステム(OMS)1802を含み得る。この処理の一部として、OMS1820は、既に作成されていなければ顧客のアカウントを作成し、要求されたサービスを顧客に提供するために顧客に対して課金するのに使用する課金および/またはアカウント情報を顧客から受け、顧客情報を検証し、検証後、顧客のためにこのオーダーを予約し、各種ワークフローを調整することにより、プロビジョニングのためにオーダーを準備するように、構成されてもよい。

40

【0176】

適切に妥当性確認がなされると、OMS1820は、処理、メモリ、およびネットワー

50

キングリソースを含む、このオーダーのためのリソースをプロビジョニングするように構成されたオーダープロビジョニングサブシステム（OPS）1824を呼び出すことができる。プロビジョニングは、オーダーのためのリソースを割り当てることと、顧客オーダーが要求するサービスを容易にするようにリソースを構成することを含み得る。オーダーのためにリソースをプロビジョニングするやり方およびプロビジョニングされるリソースのタイプは、顧客がオーダーしたクラウドサービスのタイプに依存し得る。たとえば、あるワークフローに従うと、OPS 1824を、要求されている特定のクラウドサービスを判断し、この特定のクラウドサービスのために予め構成されたであろうポッドの数を決定するように構成されてもよい。あるオーダーのために割り当てられるポッドの数は、要求されたサービスのサイズ/量/レベル/範囲に依存し得る。たとえば、割り当てるポッドの数は、サービスがサポートすべきユーザの数、サービスが要求されている期間などに基づいて決定してもよい。次に、割り当てられたポッドを、要求されたサービスを提供するために、要求している特定の顧客に合わせてカスタマイズしてもよい。

10

【0177】

クラウドインフラストラクチャシステム1802は、要求されたサービスがいつ使用できるようになるかを示すために、レスポンスまたは通知1844を、要求している顧客に送ってもよい。いくつかの例において、顧客が、要求したサービスの利益の使用および利用を開始できるようにする情報（たとえばリンク）を顧客に送信してもよい。

【0178】

クラウドインフラストラクチャシステム1802はサービスを複数の顧客に提供し得る。各顧客ごとに、クラウドインフラストラクチャシステム1802は、顧客から受けた1つ以上のサブスクリプションオーダーに関連する情報を管理し、オーダーに関連する顧客データを維持し、要求されたサービスを顧客に提供する責任がある。また、クラウドインフラストラクチャシステム1802は、申し込まれたサービスの顧客による使用に関する使用統計を収集してもよい。たとえば、統計は、使用されたストレージの量、転送されたデータの量、ユーザの数、ならびにシステムアップタイムおよびシステムダウンタイムの量などについて、収集されてもよい。この使用情報を用いて顧客に課金してもよい。課金はたとえば月ごとに行ってもよい。

20

【0179】

クラウドインフラストラクチャシステム1802は、サービスを複数の顧客に並列に提供してもよい。クラウドインフラストラクチャシステム1802は、場合によっては著作権情報を含む、これらの顧客についての情報を保存してもよい。特定の実施形態において、クラウドインフラストラクチャシステム1802は、顧客の情報を管理し管理される情報の分離を提供することで、ある顧客に関する情報が別の顧客に関する情報からアクセスされたりこの情報と混ざり合ったりしないように構成された、アイデンティティ管理サブシステム（IMS）1828を含む。IMS 1828は、アイデンティティサービス、たとえば情報アクセス管理、認証および許可サービス、顧客のアイデンティティおよび役割ならびに関連する能力などを管理するためのサービスのような、各種セキュリティ関連サービスを提供するように構成されてもよい。

30

【0180】

図19は、特定の実施形態を実現するために使用し得る例示的なコンピュータシステム1900を示す。たとえば、いくつかの実施形態において、コンピュータシステム1900を使用することにより、図1に示されるモデル生成システム102のような、上述のさまざまなサーバおよびコンピュータシステムのいずれかを実現することができる。図19に示されるように、コンピュータシステム1900は、バスサブシステム1902を介して複数の他のサブシステムと通信する処理サブシステム1904を含むさまざまなサブシステムを含む。これらの他のサブシステムは、処理加速ユニット1906、I/Oサブシステム1908、ストレージサブシステム1918および通信サブシステム1924を含み得る。ストレージサブシステム1918は、記憶媒体1922およびシステムメモリ1910を含む、非一時的なコンピュータ読取可能記憶媒体を含み得る。

40

50

【0181】

バスサブシステム1902は、コンピュータシステム1900のさまざまなコンポーネントおよびサブシステムに意図されるように互いに通信させるための機構を提供する。バスサブシステム1902は単一のバスとして概略的に示されているが、バスサブシステムの代替実施形態は複数のバスを利用してもよい。バスサブシステム1902は、さまざまなバスアーキテクチャのうちのいずれかを用いる、メモリバスまたはメモリコントローラ、周辺バス、ローカルバスなどを含むいくつかのタイプのバス構造のうちのいずれかであってもよい。たとえば、そのようなアーキテクチャは、業界標準アーキテクチャ（Industry Standard Architecture）（ISA）バス、マイクロチャネルアーキテクチャ（Micro Channel Architecture）（MCA）バス、エンハンスドISA（Enhanced ISA）（EISA）バス、ビデオ・エレクトロニクス・スタンダーズ・アソシエーション（Video Electronics Standards Association）（VESA）ローカルバス、およびIEEE P1386.1規格に従って製造される中二階バスとして実現可能な周辺コンポーネントインターコネクト（Peripheral Component Interconnect）（PCI）バスなどを含み得る。

10

【0182】

処理サブシステム1904は、コンピュータシステム1900の動作を制御し、1つ以上のプロセッサ、特定用途向け集積回路（ASIC）、またはフィールドプログラマブルゲートアレイ（FPGA）を含み得る。プロセッサは、シングルコアまたはマルチコアプロセッサを含み得る。コンピュータシステム1900の処理リソースを、1つ以上の処理ユニット1932、1934などになるように組織することができる。処理ユニットは、1つ以上のプロセッサ、同一のまたは異なるプロセッサからの1つ以上のコア、コアとプロセッサとの組み合わせ、またはコアとプロセッサとのその他の組み合わせを含み得る。いくつかの実施形態において、処理サブシステム1904は、グラフィックスプロセッサ、デジタル信号プロセッサ（DSP）などのような1つ以上の専用コプロセッサを含み得る。いくつかの実施形態では、処理サブシステム1904の処理ユニットの一部または全部は、特定用途向け集積回路（ASIC）またはフィールドプログラマブルゲートアレイ（FPGA）などのカスタマイズされた回路を使用して実現することができる。

20

【0183】

いくつかの実施形態において、処理サブシステム1904内の処理ユニットは、システムメモリ1910またはコンピュータ読取可能記憶媒体1922に格納された命令を実行することができる。さまざまな実施形態において、処理ユニットはさまざまなプログラムまたはコード命令を実行し、同時に実行する複数のプログラムまたはプロセスを維持することができる。任意の所定の時点で、実行されるべきプログラムコードの一部または全部は、システムメモリ1910および/または潜在的に1つ以上の記憶装置を含むコンピュータ読取可能記憶媒体1910に常駐していてもよい。適切なプログラミングを介して、処理サブシステム1904は、先に述べたさまざまな機能を提供することができる。コンピュータシステム1900が1つ以上の仮想マシンを実行している例において、1つ以上の処理ユニットに各仮想マシンを割り当ててもよい。

30

【0184】

特定の実施形態において、コンピュータシステム1900によって実行される全体的な処理を加速するよう、カスタマイズされた処理を実行するために、または処理サブシステム1904によって実行される処理の一部をオフロードするために、処理加速ユニット1906を任意に設けることができる。

40

【0185】

I/Oサブシステム1908は、コンピュータシステム1900に情報を入力するための、および/またはコンピュータシステム1900から、もしくはコンピュータシステム1900を介して、情報を出力するための、デバイスおよび機構を含むことができる。一般に、「入力デバイス」という語の使用は、コンピュータシステム1900に情報を入力するためのすべての考えられ得るタイプのデバイスおよび機構を含むよう意図される。ユーザインターフェイス入力デバイスは、たとえば、キーボード、マウスまたはトラックボ

50

ールなどのポインティングデバイス、ディスプレイに組み込まれたタッチパッドまたはタッチスクリーン、スクロールホイール、クリックホイール、ダイヤル、ボタン、スイッチ、キーパッド、音声コマンド認識システムを伴う音声入力デバイス、マイクロフォン、および他のタイプの入力デバイスを含んでもよい。ユーザインターフェイス入力デバイスは、ユーザが入力デバイスを制御しそれと対話することを可能にするMicrosoft Kinect（登録商標）モーションセンサ、Microsoft Xbox（登録商標）360ゲームコントローラ、ジェスチャおよび音声コマンドを用いる入力を受信するためのインターフェイスを提供するデバイスなど、モーションセンシングおよび/またはジェスチャ認識デバイスも含んでもよい。ユーザインターフェイス入力デバイスは、ユーザから目の動き（たとえば、写真を撮っている間および/またはメニュー選択を行っている間の「まばたき」）を検出し、アイジェスチャを入力デバイス（たとえばGoogle Glass（登録商標））への入力として変換するGoogle Glass（登録商標）瞬き検出器などのアイジェスチャ認識デバイスも含んでもよい。また、ユーザインターフェイス入力デバイスは、ユーザが音声コマンドを介して音声認識システム（たとえばSirri（登録商標）ナビゲータ）と対話することを可能にする音声認識感知デバイスを含んでもよい。

10

【0186】

ユーザインターフェイス入力デバイスの他の例は、三次元（3D）マウス、ジョイスティックまたはポインティングスティック、ゲームパッドおよびグラフィックタブレット、ならびにスピーカ、デジタルカメラ、デジタルカムコーダ、ポータブルメディアプレーヤ、ウェブカム、画像スキャナ、指紋スキャナ、バーコードリーダ3Dスキャナ、3Dプリンタ、レーザレンジファインダ、および視線追跡デバイスなどの聴覚/視覚デバイスも含んでもよいが、それらに限定されるものではない。また、ユーザインターフェイス入力デバイスは、たとえば、コンピュータ断層撮影、磁気共鳴撮像、ポジションエミッショントモグラフィ、および医療用超音波検査デバイスなどの医療用画像化入力デバイスを含んでもよい。ユーザインターフェイス入力デバイスは、たとえば、MIDIキーボード、デジタル楽器などの音声入力デバイスも含んでもよい。

20

【0187】

一般的に、出力デバイスという語の使用は、コンピュータシステム1900からユーザまたは他のコンピュータに情報を出力するための考えられるすべてのタイプのデバイスおよび機構を含むことを意図している。ユーザインターフェイス出力デバイスは、ディスプレイサブシステム、インジケータライト、または音声出力デバイスなどのような非ビジュアルディスプレイなどを含んでもよい。ディスプレイサブシステムは、陰極線管（CRT）、液晶ディスプレイ（LCD）またはプラズマディスプレイを使うものなどのフラットパネルデバイス、投影デバイス、タッチスクリーンなどであってもよい。たとえば、ユーザインターフェイス出力デバイスは、モニタ、プリンタ、スピーカ、ヘッドフォン、自動車ナビゲーションシステム、プロッタ、音声出力デバイスおよびモデムなどの、テキスト、グラフィックスおよび音声/映像情報を視覚的に伝えるさまざまな表示デバイスを含んでもよいが、それらに限定されるものではない。

30

【0188】

ストレージサブシステム1918は、コンピュータシステム1900によって使用される情報およびデータを格納するためのリポジトリまたはデータストアを提供する。ストレージサブシステム1918は、いくつかの実施形態の機能を提供する基本的なプログラミングおよびデータ構成を格納するための有形の非一時的なコンピュータ読取可能記憶媒体を提供する。処理サブシステム1904によって実行されると上述の機能を提供するソフトウェア（たとえばプログラム、コードモジュール、命令）が、ストレージサブシステム1918に格納されてもよい。ソフトウェアは、処理サブシステム1904の1つ以上の処理ユニットによって実行されてもよい。ストレージサブシステム1918はまた、本開示の教示に従って使用されるデータを格納するためのリポジトリを提供してもよい。

40

【0189】

50

ストレージサブシステム 1918 は、揮発性および不揮発性メモリデバイスを含む 1 つ以上の非一時的メモリデバイスを含み得る。図 19 に示すように、ストレージサブシステム 1918 は、システムメモリ 1910 およびコンピュータ読取可能記憶媒体 1922 を含む。システムメモリ 1910 は、プログラム実行中に命令およびデータを格納するための揮発性主ランダムアクセスメモリ (RAM) と、固定命令が格納される不揮発性読出専用メモリ (ROM) またはフラッシュメモリとを含む、いくつかのメモリを含み得る。いくつかの実装例において、起動中などにコンピュータシステム 1900 内の要素間における情報の転送を助ける基本的なルーチンを含むベーシックインプット/アウトプットシステム (basic input/output system) (BIOS) は、典型的には、ROM に格納されてもよい。典型的に、RAM は、処理サブシステム 1904 によって現在動作するようにされ実行されているデータおよび/またはプログラムモジュールを含む。いくつかの実装例において、システムメモリ 1910 は、スタティックランダムアクセスメモリ (SRAM)、ダイナミックランダムアクセスメモリ (DRAM) などのような複数の異なるタイプのメモリを含み得る。

10

20

30

40

50

【0190】

一例として、限定を伴うことなく、図 19 に示されるように、システムメモリ 1910 は、ウェブブラウザ、中間層アプリケーション、リレーショナルデータベース管理システム (RDBMS) などのような各種アプリケーションを含み得る、実行中のアプリケーションプログラム 1912、プログラムデータ 1914、およびオペレーティングシステム 1916 を、ロードしてもよい。一例として、オペレーティングシステム 1916 は、Microsoft Windows (登録商標)、Apple Macintosh (登録商標) および/または Linux オペレーティングシステム、さまざまな市場で入手可能な UNIX (登録商標) または UNIX 系オペレーティングシステム (さまざまな GNU/Linux オペレーティングシステム、Google Chrome (登録商標) OS などを含むがそれらに限定されない)、および/または、iOS (登録商標)、Windows (登録商標) Phone、Android (登録商標) OS、BlackBerry (登録商標) OS、Palm (登録商標) OS オペレーティングシステムのようなさまざまなバージョンのモバイルオペレーティングシステムなどを、含み得る。

【0191】

コンピュータ読取可能記憶媒体 1922 は、いくつかの実施形態の機能を提供するプログラミングおよびデータ構成を格納することができる。コンピュータ読取可能記憶媒体 1922 は、コンピュータシステム 1900 のための、コンピュータ読取可能命令、データ構造、プログラムモジュール、および他のデータのストレージを提供することができる。処理サブシステム 1904 によって実行されると上記機能を提供するソフトウェア (プログラム、コードモジュール、命令) は、ストレージサブシステム 1918 に格納されてもよい。一例として、コンピュータ読取可能記憶媒体 1922 は、ハードディスクドライブ、磁気ディスクドライブ、CD-ROM、DVD、Blu-Ray (登録商標) ディスクなどの光ディスクドライブ、またはその他の光学媒体のような不揮発性メモリを含み得る。コンピュータ読取可能記憶媒体 1922 は、Zip (登録商標) ドライブ、フラッシュメモリカード、ユニバーサルシリアルバス (USB) フラッシュドライブ、セキュアデジタル (SD) カード、DVD ディスク、デジタルビデオテープなどを含んでもよいが、それらに限定されるものではない。コンピュータ読取可能記憶媒体 1922 は、フラッシュメモリベースの SSD、エンタープライズフラッシュドライブ、ソリッドステート ROM などのような不揮発性メモリに基づくソリッドステートドライブ (SSD)、ソリッドステート RAM、ダイナミック RAM、スタティック RAM のような揮発性メモリに基づく SSD、DRAM ベースの SSD、磁気抵抗 RAM (MRAM) SSD、および DRAM とフラッシュメモリベースの SSD との組み合わせを使用するハイブリッド SSD も含み得る。

【0192】

特定の実施形態において、ストレージサブシステム 1918 は、コンピュータ読取可能

記憶媒体 1922 にさらに接続可能なコンピュータ読取可能記憶媒体リーダー 1920 も含み得る。コンピュータ読取可能記憶媒体リーダー 1920 は、ディスク、フラッシュドライブなどのようなメモリデバイスからデータを受け、読取るように構成されてもよい。

【0193】

特定の実施形態において、コンピュータシステム 1900 は、処理およびメモリリソースの仮想化を含むがこれに限定されない仮想化技術をサポートし得る。たとえば、コンピュータシステム 1900 は、1つ以上の仮想マシンを実行するためのサポートを提供することができる。特定の実施形態において、コンピュータシステム 1900 は、仮想マシンの構成および管理を容易にするためのハイパーバイザなどのプログラムを実行することができる。各仮想マシンは、メモリ、計算（たとえばプロセッサ、コア）、I/O、およびネットワークリソースを割り当てられてもよい。各仮想マシンは通常、他の仮想マシンから独立して実行される。仮想マシンは、典型的に、コンピュータシステム 1900 によって実行される他の仮想マシンによって実行されるオペレーティングシステムと同じでも異なってもよい、それ自体のオペレーティングシステムを実行する。したがって、潜在的に複数のオペレーティングシステムがコンピュータシステム 1900 によって同時に実行され得る。

10

【0194】

通信サブシステム 1924 は、他のコンピュータシステムおよびネットワークに対するインターフェイスを提供する。通信サブシステム 1924 は、他のシステムとコンピュータシステム 1900 との間のデータの送受のためのインターフェイスとして働く。たとえば、通信サブシステム 1924 は、コンピュータシステム 1900 が、1つ以上のクライアントデバイスとの間で情報を送受信するために、インターネットを介して1つ以上のクライアントデバイスへの通信チャネルを確立することを可能にすることができる。

20

【0195】

通信サブシステム 1924 は、有線および/または無線通信プロトコルの両方をサポートすることができる。たとえば、ある実施形態において、通信サブシステム 1924 は、（たとえば、セルラー電話技術、3G、4GもしくはEDGE（グローバル進化のための高速データレート）などの先進データネットワーク技術、WiFi（IEEE 802.XXファミリー規格、もしくは他のモバイル通信技術、またはそれらのいずれかの組み合わせを用いて）無線音声および/またはデータネットワークにアクセスするための無線周波数（RF）送受信機コンポーネント、グローバルポジショニングシステム（GPS）受信機コンポーネント、および/または他のコンポーネントを含み得る。いくつかの実施形態において、通信サブシステム 1924 は、無線インターフェイスに加えてまたはその代わりに、有線ネットワーク接続（たとえばEthernet）を提供することができる。

30

【0196】

通信サブシステム 1924 は、さまざまな形式でデータを受信し、送信することができる。たとえば、いくつかの実施形態において、通信サブシステム 1924 は、構造化データフィールドおよび/または非構造化データフィールド 1926、イベントストリーム 1928、イベントアップデート 1930 などの形式で入力通信を受信することができる。たとえば、通信サブシステム 1924 は、ソーシャルメディアネットワークおよび/またはTwitter（登録商標）フィールド、Facebook（登録商標）アップデート、Rich Site Summary（RSS）フィールドなどのウェブフィールド、および/または1つ以上の第三者情報源からのリアルタイムアップデートなどのような他の通信サービスのユーザから、リアルタイムでデータフィールド 1926 を受信（または送信）するように構成されてもよい。

40

【0197】

特定の実施形態において、通信サブシステム 1924 は、連続データストリームの形式でデータを受信するように構成されてもよく、当該連続データストリームは、明確な終端を持たない、本来は連続的または無限であり得るリアルタイムイベントのイベントストリーム 1928 および/またはイベントアップデート 1930 を含んでもよい。連続データ

50

を生成するアプリケーションの例としては、たとえば、センサデータアプリケーション、金融株式相場表示板、ネットワーク性能測定ツール（たとえばネットワークモニタリングおよびトラフィック管理アプリケーション）、クリックストリーム解析ツール、自動車交通モニタリングなどを挙げることができる。

【0198】

通信サブシステム1924は、コンピュータシステム1900からのデータをその他のコンピュータシステムまたはネットワークに伝えるように構成されてもよい。このデータは、構造化および/または非構造化データフィード1926、イベントストリーム1928、イベントアップデート1930などのような各種異なる形式で、コンピュータシステム1900に結合された1つ以上のストリーミングデータソースコンピュータと通信し得る1つ以上のデータベースに、伝えることができる。

10

【0199】

コンピュータシステム1900は、ハンドヘルドポータブルデバイス（たとえばiPhone（登録商標）セルラフォン、iPad（登録商標）コンピューティングタブレット、PDA）、ウェアラブルデバイス（たとえばGoogle Glass（登録商標）ヘッドマウントディスプレイ）、パーソナルコンピュータ、ワークステーション、メインフレーム、キオスク、サーバラック、またはその他のデータ処理システムを含む、さまざまなタイプのもののうちの1つであればよい。常に変化するコンピュータおよびネットワークの性質のため、図19に示されるコンピュータシステム1900の記載は、単に具体的な例を意図している。図3に示されるシステムよりも多くのコンポーネントまたは少ないコンポーネントを有するその他多くの構成が可能である。本明細書における開示および教示に基づいて、当業者は、さまざまな実施形態を実現するための他の態様および/または方法を理解するであろう。

20

【0200】

特定の実施形態について説明したが、さまざまな修正、変更、代替構成、および均等物が可能である。実施形態は、特定のデータ処理環境内の動作に限定されず、複数のデータ処理環境内で自由に動作することができる。さらに、実施形態を特定の連続のトランザクションおよびステップを使用して説明したが、これが限定を意図しているのではないことは当業者には明らかであるはずである。いくつかのフローチャートは動作を逐次的プロセスとして説明しているが、これらの動作のうちの多くは並列または同時に実行できる。加えて、動作の順序を再指定してもよい。プロセスは図面に含まれない追加のステップを有し得る。上記実施形態の各種特徴および側面は、個別に使用されてもよく、またはともに使用されてもよい。

30

【0201】

さらに、特定の実施形態をハードウェアとソフトウェアとの特定の組み合わせを用いて説明したが、ハードウェアとソフトウェアとの他の組み合わせも可能であることが理解されるはずである。特定の実施形態は、ハードウェアでのみ、またはソフトウェアでのみ、またはそれらの組み合わせを用いて実現されてもよい。本明細書に記載されたさまざまなプロセスは、同じプロセッサまたは任意の組み合わせの異なるプロセッサ上で実現できる。

40

【0202】

デバイス、システム、コンポーネントまたはモジュールが特定の動作または機能を実行するように構成されると記載されている場合、そのような構成は、たとえば、動作を実行する電子回路を設計すること、プログラミング可能な電子回路（マイクロプロセッサなど）をプログラミングすることにより、たとえば非一時的なメモリ媒体に格納されたコードまたは命令またはその任意の組み合わせを実行するようにプログラミングされたコンピュータ命令もしくはコード、またはプロセッサもしくはコアを実行するなどして、動作を実行するように電子回路を設計することによって、動作を実行すること、またはそれらの任意の組み合わせによって達成され得る。プロセスは、プロセス間通信のための従来の技術を含むがこれに限定されないさまざまな技術を使用して通信することができ、異なる対の

50

プロセスは異なる技術を使用してもよく、同じ対のプロセスは異なる時間に異なる技術を使用してもよい。

【0203】

本開示では特定の詳細事項を示すことにより実施形態が十分に理解されるようにしている。しかしながら、実施形態はこれらの特定の詳細事項がなくとも実施し得るものである。たとえば、周知の回路、プロセス、アルゴリズム、構成、および技術は、実施形態が曖昧にならないようにするために不必要な詳細事項なしで示している。本明細書は例示的な実施形態のみを提供し、他の実施形態の範囲、適用可能性、または構成を限定することを意図しているのではない。むしろ、実施形態の上記説明は、各種実施形態を実現することを可能にする説明を当業者に提供する。各種変更は要素の機能および構成の範囲内で行うことができる。

10

【0204】

したがって、明細書および図面は、限定的な意味ではなく例示的なものとみなされるべきである。しかしながら、請求項に記載されているより広範な精神および範囲から逸脱することなく、追加、削減、削除、ならびに他の修正および変更がこれらになされ得ることは明らかであろう。このように、特定の実施形態を説明してきたが、これらは限定を意図するものではない。さまざまな修正および均等物は以下の請求項の範囲に含まれる。さまざまな修正は、開示された特徴の関連の任意の組み合わせを含む。

【図1】

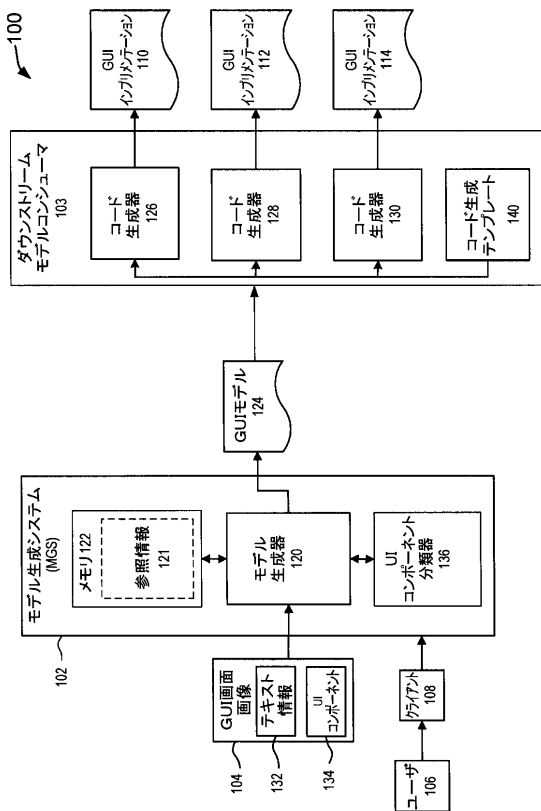


FIG. 1

【図2】

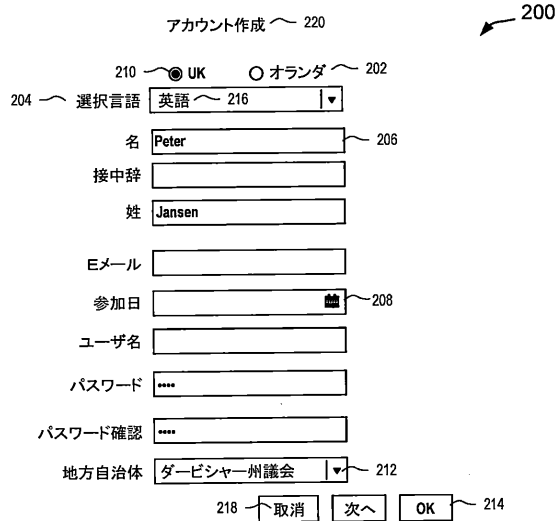


FIG. 2

【 図 3 】

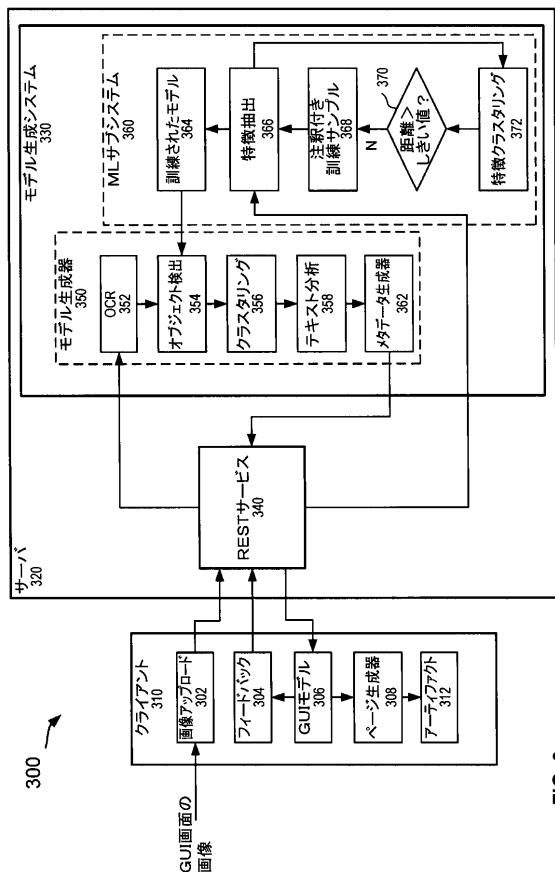


FIG. 3

【 図 4 】

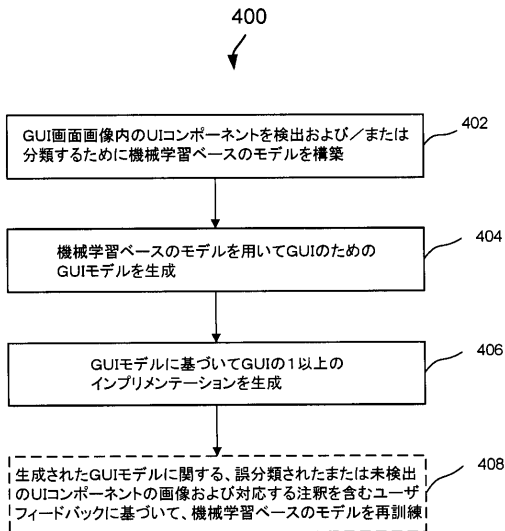


FIG. 4

【 図 5 】

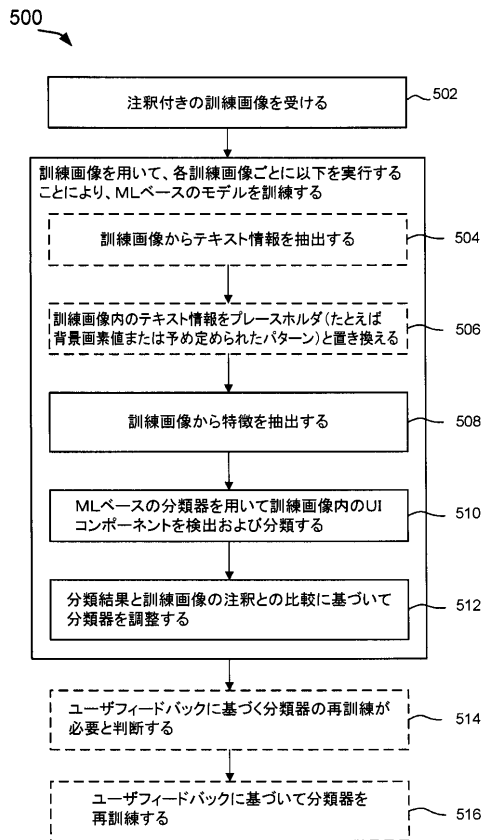


FIG. 5

【 図 6 】

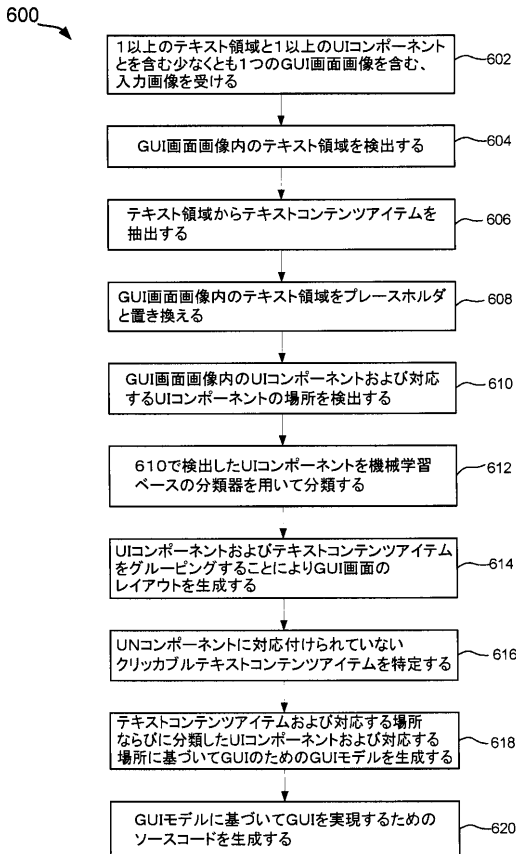
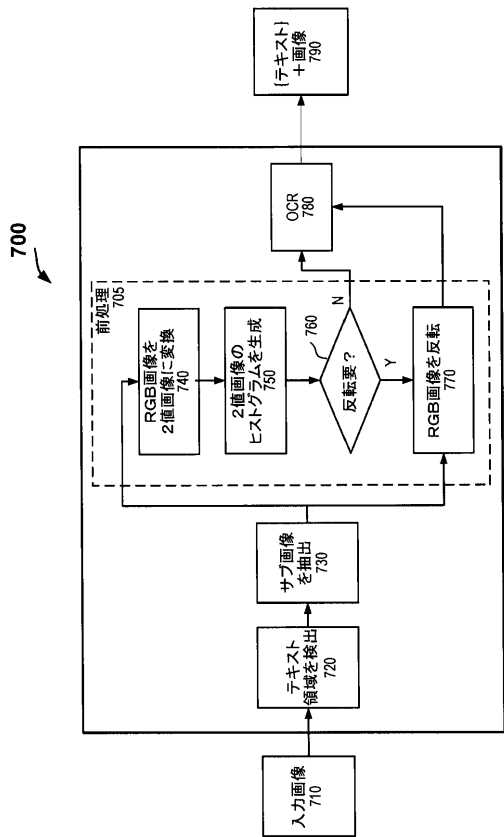


FIG. 6

【 図 7 】



【 図 8 】

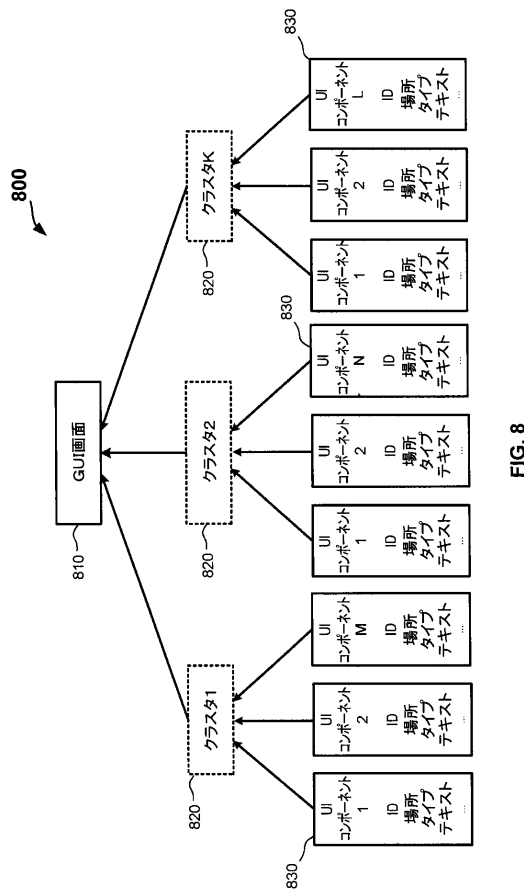


FIG. 7

FIG. 8

【 図 9 】

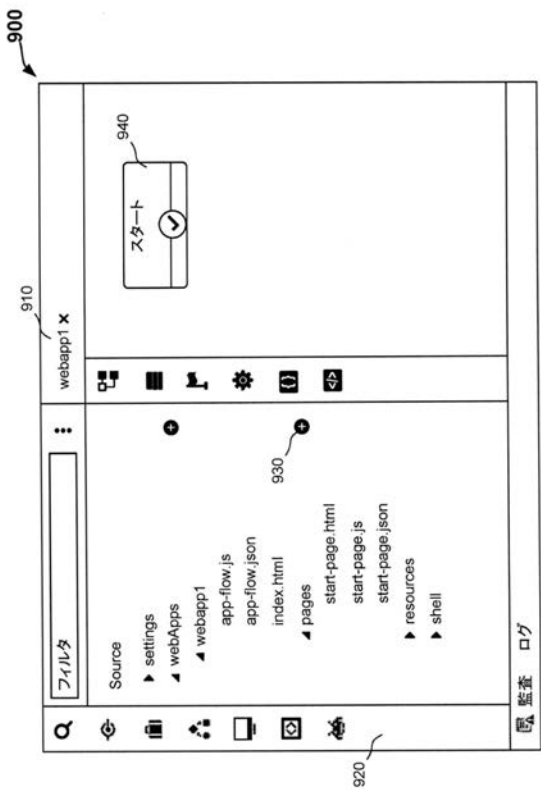


FIG. 9

【 図 10 】



FIG. 10

【 図 1 1 】

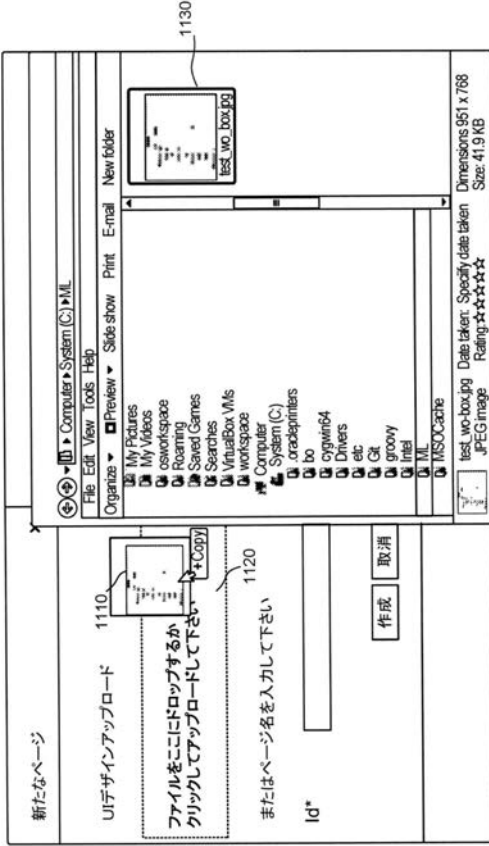


FIG. 11

【 図 1 2 】

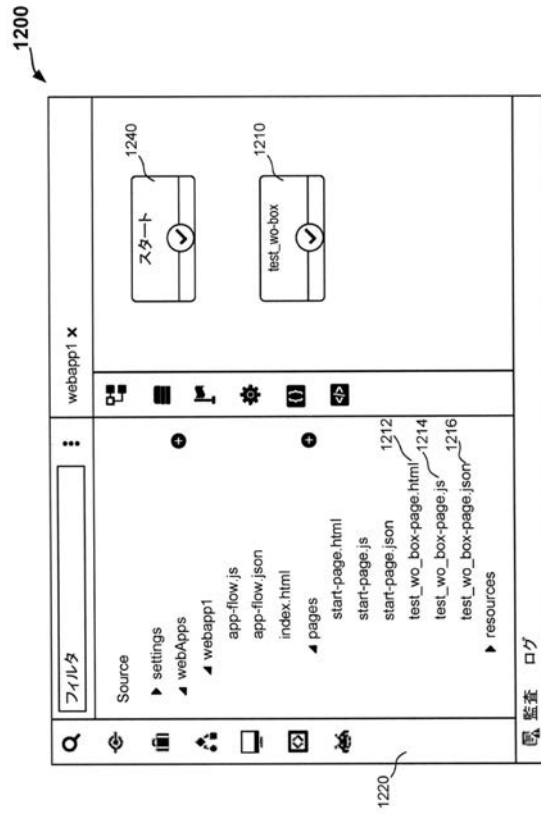


FIG. 12

```

"ui_data": {
  "rows": [
    {
      "background": "#ffffff",
      "column": [
        {
          "component_type": "label",
          "label": "Details",
          "text_within_component": null
        },
        {
          "component_type": "image",
          "image_name": "facell.jpg",
          "label": null,
          "text_within_component": null
        },
        {
          "component_type": "label",
          "label": "Change Photo",
          "text_within_component": null
        },
        {
          "component_type": "label",
          "label": "First Name*",
          "text_within_component": null
        },
        {
          "component_type": "inputtext",
          "label": null,
          "text_within_component": "Lynn"
        },
        {
          "component_type": "label",
          "label": "Mobile",
          "text_within_component": null
        },
        {
          "component_type": "inputtext",
          "label": null,
          "text_within_component": "650-555-1212"
        }
      ]
    },
    {
      "height": 80,
      "row": 0,
      "width": 80
    }
  ],
  "section": 0
}
    
```

【 図 1 4 】

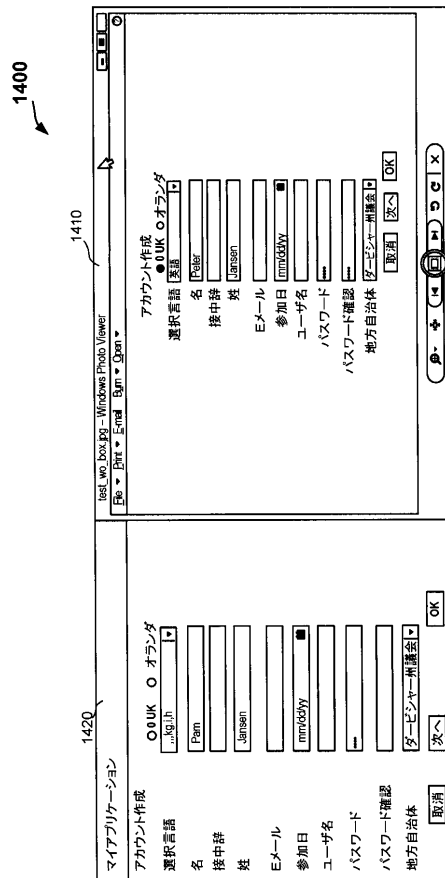


FIG. 14

【 図 1 5 】

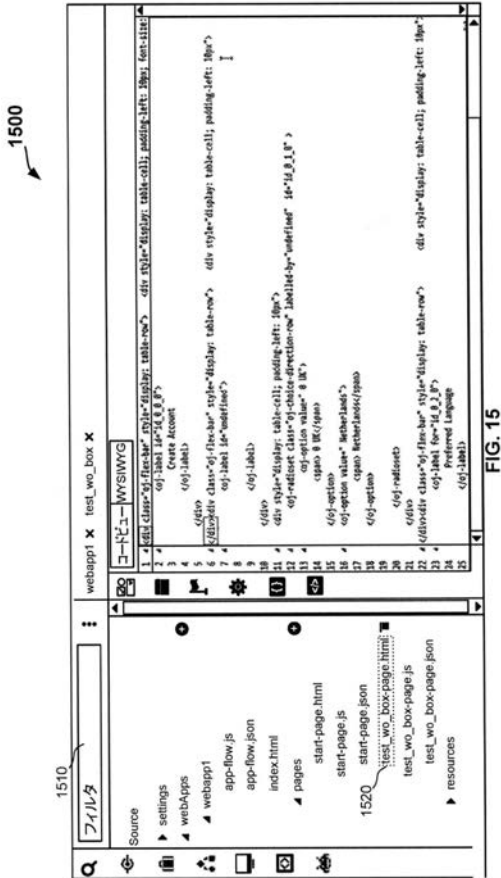


FIG. 15

【 図 1 6 A 】

1600

詳細

写真変更

名 Lynn 携帯 650-555-1212
 姓 Smith 電話 650-555-1213
 住所 14 Water Street Eメール lynn.smith@fixitfast.com
 市 Burlingame
 州 CA

FIG. 16A

【 図 1 6 B 】

1650

詳細

写真変更

名 Lynn 携帯 650-555-1212
 姓 Smith 電話 650-555-1213
 住所 14 Waterstreet Eメール lynn.smith@fixitfast.com
 市 Burlingame
 州 CA

FIG. 16B

【 図 1 7 】

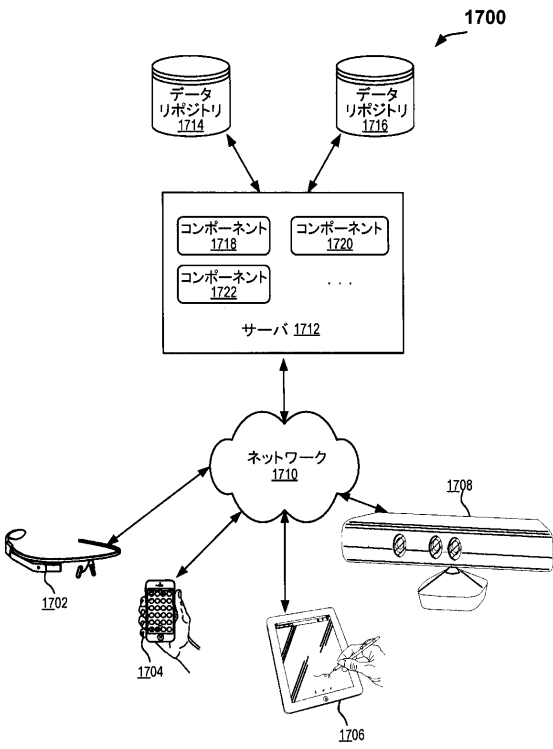


FIG. 17

【 図 1 8 】

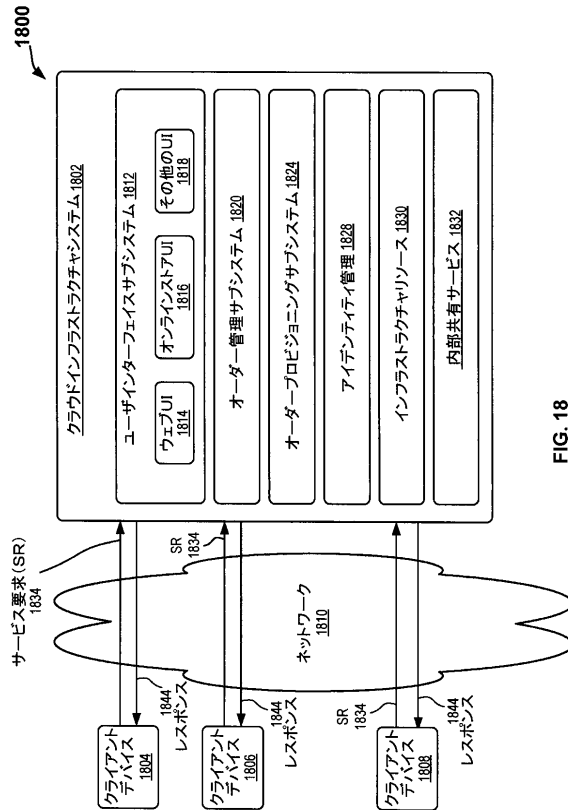


FIG. 18

【 図 1 9 】

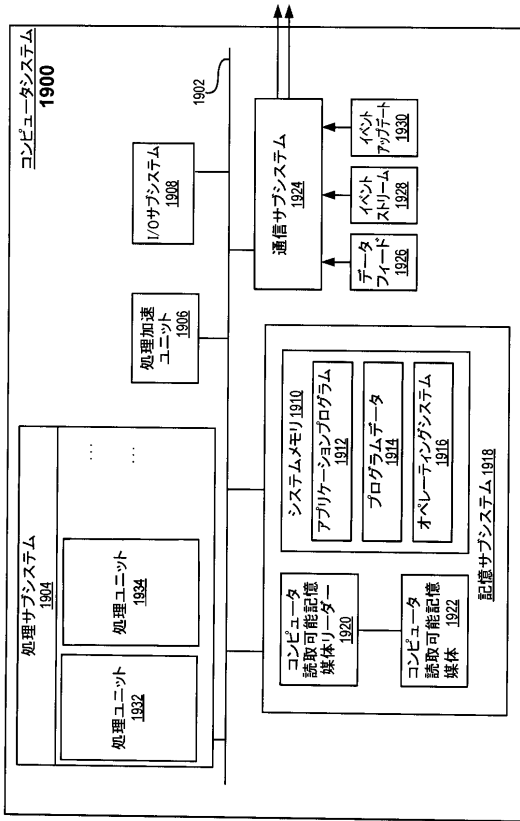


FIG. 19

【 手続 補 正 書 】

【 提 出 日 】 令 和 2 年 4 月 2 8 日 (2 0 2 0 . 4 . 2 8)

【 手 続 補 正 1 】

【 補 正 対 象 書 類 名 】 特 許 請 求 の 範 囲

【 補 正 対 象 項 目 名 】 全 文

【 補 正 方 法 】 変 更

【 補 正 の 内 容 】

【 特 許 請 求 の 範 囲 】

【 請 求 項 1 】

方 法 であ っ て、

コ ン ピ ュ ー タ シ ス テ ム が、入 力 画 像 か ら、グ ラ フ ィ ッ ク ユ ー ザ イン タ ー フ ェ イ ス (G U I) の G U I 画 面 を 示 す G U I 画 面 画 像 を 検 出 す る ス テ ッ プ と、

前 記 コ ン ピ ュ ー タ シ ス テ ム が、第 1 の テ キ ス ト コ ン テ ン ツ アイ テ ム を 含 む 前 記 G U I 画 面 画 像 の 第 1 の 領 域 を 検 出 す る ス テ ッ プ と、

前 記 コ ン ピ ュ ー タ シ ス テ ム が、前 記 G U I 画 面 画 像 内 の 前 記 第 1 の 領 域 の 位 置 を 判 断 す る ス テ ッ プ と、

前 記 コ ン ピ ュ ー タ シ ス テ ム が、前 記 第 1 の 領 域 の コ ン テ ン ツ を、テ キ ス ト コ ン テ ン ツ を 含 ま ない プ レ ス ホ ル ダ に 置 き 換 え る ス テ ッ プ と、

前 記 コ ン ピ ュ ー タ シ ス テ ム が、前 記 G U I 画 面 画 像 の 第 2 の 領 域 内 に 位 置 す る 第 1 の ユ ー ザ イン タ ー フ ェ イ ス (U I) コ ン ポ ー ネ ント を 検 出 す る ス テ ッ プ と を 含 み、前 記 第 2 の 領 域 は 前 記 プ レ ス ホ ル ダ を 有 す る 前 記 第 1 の 領 域 を 含 み、

前 記 コ ン ピ ュ ー タ シ ス テ ム が、前 記 G U I 画 面 画 像 内 の 前 記 第 1 の U I コ ン ポ ー ネ ント の 位 置 を 判 断 す る ス テ ッ プ と、

前 記 コ ン ピ ュ ー タ シ ス テ ム が、機 械 学 習 ベ ー ス の 分 類 器 を 用 い て、前 記 第 1 の U I コ ン ポ ー ネ ント の U I コ ン ポ ー ネ ント タイ プ を 判 断 す る ス テ ッ プ と を 含 み、前 記 機 械 学 習 ベ ー

スの分類器は、複数の訓練画像を含む訓練データを用いて訓練され、前記複数の訓練画像のうちの各訓練画像はUIコンポーネントを含み、前記訓練データはさらに、各訓練画像ごとに、前記訓練画像内の前記UIコンポーネントのUIコンポーネントタイプを特定する情報を含み、

前記コンピュータシステムが、前記GUIを実現するためのコードを生成するために使用可能なGUIモデルを生成するステップを含み、前記GUIモデルは、前記第1のUIコンポーネントの情報と、前記第1のテキストコンテンツアイテムの情報とを含み、前記第1のUIコンポーネントの情報は、前記第1のUIコンポーネントについて判断した前記UIコンポーネントタイプと、前記GUI画面画像内の前記第1のUIコンポーネントの位置とを示す情報を含む、方法。

【請求項2】

前記第1のUIコンポーネントを検出するステップは、前記機械学習ベースの分類器が前記GUI画面画像内の前記第1のUIコンポーネントを検出するステップを含み、

前記訓練データはさらに、各訓練画像ごとに、前記訓練画像内のUIコンポーネントの位置を含む、請求項1に記載の方法。

【請求項3】

前記GUI画面画像内の前記第1の領域の位置および前記GUI画面画像内の前記第1のUIコンポーネントの位置に基づいて、前記第1のテキストコンテンツアイテムとともに前記第1のUIコンポーネントをグルーピングするステップをさらに含み、

前記GUIモデルを生成するステップはさらに、前記GUIモデルに、前記グルーピングを示す情報を含めるステップを含む、請求項1または2に記載の方法。

【請求項4】

前記コンピュータシステムが、第2のテキストコンテンツアイテムを含む前記GUI画面画像の第3の領域を検出するステップと、

前記コンピュータシステムが、前記GUI画面画像内の前記第3の領域の位置を判断するステップと、

前記コンピュータシステムが、前記GUI画面画像内に位置する第2のUIコンポーネントを検出するステップと、

前記GUI画面画像内の前記第2のUIコンポーネントの位置を判断するステップと、

前記機械学習ベースの分類器を用いて、前記第2のUIコンポーネントのUIコンポーネントタイプを判断するステップと、

前記第1の領域の前記位置、前記第3の領域の前記位置、前記第1のUIコンポーネントの前記UIコンポーネントタイプおよび位置、ならびに前記第2のUIコンポーネントの前記UIコンポーネントタイプおよび位置に基づいて、前記第1のテキストコンテンツアイテム、前記第2のテキストコンテンツアイテム、前記第1のUIコンポーネント、および前記第2のUIコンポーネントをグルーピングするステップと、

前記グルーピングに基づいて前記GUI画面のレイアウトを決定するステップとを含み、

前記GUIモデルを生成するステップはさらに、前記GUIモデルに、前記グルーピングおよび前記GUI画面の前記レイアウトを示す情報を含めるステップを含む、請求項1～3のいずれか1項に記載の方法。

【請求項5】

前記GUI画面画像内の前記第3の領域の前記位置に基づいて、前記第2のテキストコンテンツは前記GUI画面画像内のいずれのUIコンポーネントにも対応付けられていないと判断するステップと、

前記第2のテキストコンテンツアイテムはアクションを示すと判断するステップとをさらに含み、

前記GUIモデルを生成するステップは、前記GUIモデルにおいて、前記GUI画面画像の前記第3の領域内の前記第2のテキストコンテンツアイテムはクリックブルテキストであることを示すステップを含み、前記第2のテキストコンテンツのクリックは前記ア

クションを開始させる、請求項 4 に記載の方法。

【請求項 6】

前記第 1 の領域は背景上の前記第 1 のテキストコンテンツアイテムを含み、前記方法はさらに、

前記 G U I 画面画像の前記第 1 の領域内の画素の強度値に基づいてヒストグラムを生成するステップと、

前記 G U I 画面画像の前記第 1 の領域内の、前記第 1 のテキストコンテンツアイテムに対応する画素の数と前記背景に対応する画素の数とに基づいて、前記ヒストグラムから、前記第 1 のテキストコンテンツアイテムの画素の強度値は前記背景の画素の強度値よりも高いと判断するステップと、

前記第 1 の領域内の画素の強度値を反転させるステップとを含み、前記反転により、前記第 1 のテキストコンテンツアイテムの画素の強度値は、前記第 1 の領域内の前記背景の画素の強度値よりも低くなり、

前記反転の実施後に前記第 1 の領域内の前記第 1 のテキストコンテンツアイテムを認識するステップを含む、請求項 1 ~ 5 のいずれか 1 項に記載の方法。

【請求項 7】

前記ヒストグラムを生成する前に、前記第 1 の領域内の各画素の強度値に基づいて、前記 G U I 画面画像の前記第 1 の領域を、R G B サブ画像から 2 値サブ画像に変換するステップをさらに含む、請求項 6 に記載の方法。

【請求項 8】

前記プレースホルダは画素の予め定められたパターンを含み、

前記第 1 の U I コンポーネントの U I コンポーネントタイプを判断するステップは、前記第 2 の領域内の前記プレースホルダの存在と前記プレースホルダ内の前記画素の予め定められたパターンとに基づいて、前記 G U I 画面画像の前記第 2 の領域を分類するステップを含む、請求項 1 ~ 7 のいずれか 1 項に記載の方法。

【請求項 9】

前記プレースホルダは、前記第 1 の領域内の前記第 1 のテキストコンテンツアイテムの位置の境界を前記 G U I 画面画像から判断することを可能にする画素の予め定められたパターンである、請求項 1 ~ 8 のいずれか 1 項に記載の方法。

【請求項 10】

前記画素の予め定められたパターンの画素値は、予め定められた値、または前記第 1 の領域の背景内の画素の画素値を含む、請求項 9 に記載の方法。

【請求項 11】

前記 G U I モデルに基づいて前記 G U I の 1 つ以上のインプリメンテーションを生成するステップをさらに含む、請求項 1 ~ 10 のいずれか 1 項に記載の方法。

【請求項 12】

前記 G U I モデルに基づいて前記 G U I の 1 つ以上のインプリメンテーションを生成するステップは、

前記 G U I モデルを用いて、第 1 のプラットフォームのための前記 G U I の第 1 のインプリメンテーションを生成するステップと、

前記 G U I モデルを用いて、第 2 のプラットフォームのための前記 G U I の第 2 のインプリメンテーションを生成するステップとを含み、前記第 2 のプラットフォームは前記第 1 のプラットフォームと異なる、請求項 11 に記載の方法。

【請求項 13】

前記 G U I モデルに基づいて前記 G U I の 1 つ以上のインプリメンテーションを生成するステップは、

前記 G U I モデルを用いて、第 1 のプログラミング言語で前記 G U I の第 1 のインプリメンテーションを生成するステップと、

前記 G U I モデルを用いて、第 2 のプログラミング言語で前記 G U I の第 2 のインプリメンテーションを生成するステップとを含み、前記第 2 のプログラミング言語は前記第 1

のプログラミング言語と異なる、請求項 11 または 12 に記載の方法。

【請求項 14】

前記 GUI モデルに基づいて前記 GUI の 1 つ以上のインプリメンテーションを生成するステップは、

前記 GUI モデルと 1 つ以上のコード生成テンプレートとを用いて前記 GUI の 1 つ以上のインプリメンテーションを生成するステップを含み、各コード生成テンプレートはプラットフォームまたはプログラミング言語に対応付けられている、請求項 11 に記載の方法。

【請求項 15】

前記 GUI モデルを生成するステップは、前記 GUI モデルの情報を、JavaScript オブジェクト表記 (JSON) フォーマットで格納するステップを含む、請求項 1 ~ 14 のいずれか 1 項に記載の方法。

【請求項 16】

前記機械学習ベースの分類器は、画像ヒストグラムベースの非線形サポートベクターマシン分類器、または、人工ニューラルネットワークベースの分類器を含む、請求項 1 ~ 15 のいずれか 1 項に記載の方法。

【請求項 17】

前記 GUI モデルに関するユーザフィードバックを受けるステップをさらに含み、前記ユーザフィードバックは、

前記 GUI モデルにおいて前記第 1 の UI コンポーネントについて指定された前記 UI コンポーネントタイプの代わりに前記第 1 の UI コンポーネントに対応付けられる新たな UI コンポーネントタイプを特定する情報、または、

GUI 画面内に存在するが前記 GUI モデルに含まれない新たな UI コンポーネントを特定する情報、および前記新たな UI コンポーネントの UI コンポーネントタイプを示す情報を含み、

前記ユーザフィードバックに基づいて前記機械学習ベースの分類器を再訓練するステップをさらに含む、請求項 1 ~ 16 のいずれか 1 項に記載の方法。

【請求項 18】

前記ユーザフィードバックはさらに、前記第 1 の UI コンポーネントまたは前記新たな UI コンポーネントのサブ画像を含み、前記方法はさらに、

前記複数の訓練画像から特徴を抽出するステップと、

前記複数の訓練画像から抽出した前記特徴を、多次元空間内のデータポイントにマッピングするステップとを含み、前記データポイントは、前記多次元空間内の一組のクラスタを形成し、

前記第 1 の UI コンポーネントまたは前記新たな UI コンポーネントの前記サブ画像から特徴を抽出するステップと、

前記第 1 の UI コンポーネントまたは前記新たな UI コンポーネントの前記サブ画像から抽出した特徴を、前記多次元空間内のデータポイントにマッピングするステップと、

前記第 1 の UI コンポーネントまたは前記新たな UI コンポーネントの前記サブ画像に対応する前記データポイントと、前記一組のクラスタの各クラスタの中心との間の距離を判断するステップと、

前記距離がしきい値未満であるという判断に応じて、前記第 1 の UI コンポーネントまたは前記新たな UI コンポーネントの前記サブ画像を前記訓練データに含めるステップとを含む、請求項 17 に記載の方法。

【請求項 19】

請求項 1 ~ 18 のいずれか 1 項に記載の方法を 1 つ以上のプロセッサに実行させるためのコンピュータ読取可能プログラム。

【請求項 20】

システムであって、

1 つ以上のプロセッサと、

前記 1 つ以上のプロセッサに結合されたメモリとを備え、前記メモリは、前記 1 つ以上のプロセッサによって実行されると前記システムに、

入力画像から、グラフィックユーザインターフェイス (G U I) の G U I 画面を示す G U I 画面画像を検出することと、

第 1 のテキストコンテンツアイテムを含む前記 G U I 画面画像の第 1 の領域を検出することと、

前記 G U I 画面画像内の前記第 1 の領域の位置を判断することと、

前記第 1 の領域のコンテンツを、テキストコンテンツを含まないブレースホルダに置き換えることと、

前記 G U I 画面画像の第 2 の領域内に位置する第 1 のユーザインターフェイス (U I) コンポーネントを検出することとを実行させ、前記第 2 の領域は前記ブレースホルダを有する前記第 1 の領域を含み、

前記 G U I 画面画像内の前記第 1 の U I コンポーネントの位置を判断することと、

機械学習ベースの分類器を用いて、前記第 1 の U I コンポーネントの U I コンポーネントタイプを判断することとを実行させ、前記機械学習ベースの分類器は、複数の訓練画像を含む訓練データを用いて訓練され、前記複数の訓練画像のうちの各訓練画像は U I コンポーネントを含み、前記訓練データはさらに、各訓練画像ごとに、前記訓練画像内の前記 U I コンポーネントの U I コンポーネントタイプを特定する情報を含み、

前記 G U I を実現するためのコードを生成するために使用可能な G U I モデルを生成することを実行させ、前記 G U I モデルは、前記第 1 の U I コンポーネントの情報と、前記第 1 のテキストコンテンツアイテムの情報とを含み、前記第 1 の U I コンポーネントの情報は、前記第 1 の U I コンポーネントについて判断した前記 U I コンポーネントタイプと、前記 G U I 画面画像内の前記第 1 の U I コンポーネントの位置とを示す情報を含む、システム。

【 国際調査報告 】

INTERNATIONAL SEARCH REPORT

International application No PCT/US2018/041038

A. CLASSIFICATION OF SUBJECT MATTER INV. G06F8/38 G06N3/08 ADD.		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) G06F G06N		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) EPO-Internal		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2014/068553 A1 (BALASUBRAMANIAN KAMESH SIVARAMAN [US]) 6 March 2014 (2014-03-06) paragraph [0062] paragraph [0064] paragraph [0074] - paragraph [0076] paragraph [0089] -----	1-20
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> See patent family annex.		
* Special categories of cited documents : "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family		
Date of the actual completion of the international search 22 October 2018		Date of mailing of the international search report 29/10/2018
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016		Authorized officer Ebert, Werner

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2018/041038

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2014068553	A1	NONE	

フロントページの続き

(81)指定国・地域 AP(BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), EA(AM, AZ, BY, KG, KZ, RU, TJ, TM), EP(AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG), AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT

(特許庁注：以下のものは登録商標)

1 . P H O T O S H O P

(72)発明者 セン , プランティック

インド、 5 6 0 0 7 6 バンガロール、 ビィ・ティ・エム・ステージ・2、 トゥウェンティナイン
ス・メイン、 フォーティーンズ・エイ・クロス、 ハウス・ナンバー・1 2 5 4、 ドア・ナンバー・
0 3

Fターム(参考) 5B376 BC21 BC31 BC43

5L096 BA08 BA18 FA06 FA69 KA04 KA15