

(19) 日本国特許庁(JP)

## (12) 特許公報(B2)

(11) 特許番号

特許第6168928号  
(P6168928)

(45) 発行日 平成29年7月26日(2017.7.26)

(24) 登録日 平成29年7月7日(2017.7.7)

(51) Int.Cl.

H04N 1/387 (2006.01)  
G06T 11/60 (2006.01)

F 1

H04N 1/387  
G06T 11/60 100D

請求項の数 13 (全 35 頁)

(21) 出願番号 特願2013-184047 (P2013-184047)  
 (22) 出願日 平成25年9月5日 (2013.9.5)  
 (65) 公開番号 特開2015-53543 (P2015-53543A)  
 (43) 公開日 平成27年3月19日 (2015.3.19)  
 審査請求日 平成28年8月30日 (2016.8.30)

(73) 特許権者 000001007  
 キヤノン株式会社  
 東京都大田区下丸子3丁目30番2号  
 (74) 代理人 100126240  
 弁理士 阿部 琢磨  
 (74) 代理人 100124442  
 弁理士 黒岩 創吾  
 (72) 発明者 加藤 真夫  
 東京都大田区下丸子3丁目30番2号キヤ  
 ノン株式会社内  
 (72) 発明者 橋井 雄介  
 東京都大田区下丸子3丁目30番2号キヤ  
 ノン株式会社内

最終頁に続く

(54) 【発明の名称】画像処理装置、画像処理方法およびプログラム

## (57) 【特許請求の範囲】

## 【請求項 1】

複数の画像を含む画像群を取得する取得手段と、  
 前記取得手段により取得した画像群から所定の数の画像を選択してテンプレートに配置  
 するための処理を実行することにより、複数のレイアウト候補を作成する作成手段と、  
 前記作成手段により作成されたレイアウト候補の評価値を求める評価手段と、  
 前記評価手段により求められた評価値と、所定のパラメータと、に基づいて、複数のレ  
 イアウト候補の中から、提示用の複数のレイアウト候補を選択するとともに、前記選択さ  
 れたレイアウト候補の提示順を決定する決定手段と、

前記決定手段により決定された提示順でレイアウト候補を提示する提示手段と、  
 を備え、

前記決定手段により、前記評価値が第1の値のレイアウト候補は、前記評価値が前記第  
 1の値よりも低い第2の値のレイアウト候補よりも選択される確率が高くなり、且つ前記  
 提示順は前記評価値の順とは異なる順となることを特徴とする画像処理装置。

## 【請求項 2】

前記決定手段は、所定の数のレイアウト候補を提示する場合、評価値の高いレイアウト  
 候補の数が評価値の低いレイアウト候補の数より多くなるように提示順を決定することを  
 特徴とする請求項1に記載の画像処理装置。

## 【請求項 3】

前記作成手段は、レイアウト内に配置する画像の選択を少なくとも変化させて複数のレ

10

20

イアウト候補を作成することを特徴とする請求項 1 又は 2 に記載の画像処理装置。

【請求項 4】

前記決定手段は、前記評価手段による評価結果の評価順及び乱数に基づいて、提示順を決定することを特徴とする請求項 1 ~ 3 のいずれか 1 項に記載の画像処理装置。

【請求項 5】

前記決定手段は、前記評価手段による評価結果の評価順の度数分布に基づいて、提示順を決定することを特徴とする請求項 1 ~ 3 のいずれか 1 項に記載の画像処理装置。

【請求項 6】

前記提示手段は、アプリケーションの起動時、所定の時間が経過した後、前記決定手段により決定したレイアウトを印刷した後、及び P C の起動時のうち少なくともいずれかのタイミングで提示することを特徴とする請求項 1 ~ 5 のいずれか 1 項に記載の画像処理装置。10

【請求項 7】

複数の画像を含む画像群を取得する取得工程と、

前記取得工程において取得した画像群から所定の数の画像を選択してテンプレートに配置するための処理を実行することにより、複数のレイアウト候補を作成する作成工程と、

前記作成工程において作成されたレイアウト候補の評価値を求める評価工程と、

前記評価工程により求められた評価値と、所定のパラメータと、に基づいて、複数のレイアウト候補の中から、提示用の複数のレイアウト候補を選択するとともに、前記選択されたレイアウト候補の提示順を決定する決定工程と、20

前記決定工程において決定された提示順でレイアウト候補を提示する提示工程と、を備え、

前記決定工程により、前記評価値が第 1 の値のレイアウト候補は、前記評価値が前記第 1 の値よりも低い第 2 の値のレイアウト候補よりも選択される確率が高くなり、且つ前記提示順が前記評価工程における評価値の順とは異なる順となることを特徴とする画像処理方法。20

【請求項 8】

コンピュータに、

複数の画像を含む画像群を取得する取得工程と、

前記取得工程において取得した画像群から所定の数の画像を選択してテンプレートに配置するための処理を実行することにより、複数のレイアウト候補を作成する作成工程と、30

前記作成工程において作成されたレイアウト候補の評価値を求める評価工程と、

前記評価工程により求められた評価値と、所定のパラメータと、に基づいて、複数のレイアウト候補の中から、提示用の複数のレイアウト候補を選択するとともに、前記選択されたレイアウト候補の提示順を決定する決定工程と、30

前記決定工程において決定された提示順でレイアウト候補を提示する提示工程と、を実行させるプログラムであり、

前記決定工程により、前記評価値が第 1 の値のレイアウト候補は、前記評価値が前記第 1 の値よりも低い第 2 の値のレイアウト候補よりも選択される確率が高くなり、且つ前記提示順が前記評価工程における評価値の順とは異なる順となることを特徴とするプログラム。40

【請求項 9】

前記決定工程では、所定の数のレイアウト候補を提示する場合、評価値の高いレイアウト候補の数が評価値の低いレイアウト候補の数より多くなるように提示順を決定することを特徴とする請求項 8 に記載のプログラム。

【請求項 10】

前記作成工程では、レイアウト内に配置する画像の選択を少なくとも変化させて複数のレイアウト候補を作成することを特徴とする請求項 8 又は 9 に記載のプログラム。

【請求項 11】

前記決定工程では、前記評価工程による評価結果の評価順及び乱数に基づいて、提示順

10

20

30

40

50

を決定することを特徴とする請求項 8 ~ 10 のいずれか 1 項に記載のプログラム。

**【請求項 12】**

前記決定工程では、前記評価工程による評価結果の評価順の度数分布に基づいて、提示順を決定することを特徴とする請求項 8 ~ 11 のいずれか 1 項に記載のプログラム。

**【請求項 13】**

前記提示工程では、アプリケーションの起動時、所定の時間が経過した後、前記決定工程により決定したレイアウトを印刷した後、及び PC の起動時のうち少なくともいずれかのタイミングで提示することを特徴とする請求項 8 ~ 12 のいずれか 1 項に記載のプログラム。

**【発明の詳細な説明】**

10

**【技術分野】**

**【0001】**

本発明は、画像処理のための画像処理装置、画像処理方法及びプログラムに関する。

**【背景技術】**

**【0002】**

近年、デジタルカメラやカメラ付携帯端末の普及により、容易に写真を撮影できる環境が整ってきている。また、PCを利用することで、複数の写真を組み合わせてコラージュやアルバムなどの様々な成果物を作成する方法が知られている。特許文献 1 では、画像に含まれるオブジェクトの向きを特定し、特定されたオブジェクトの向きに基づいて画像を出力領域内にレイアウトし、レイアウトした画像を出力する画像出力方法が提案されている。また、特許文献 2 には、複数画像を配置する際に、基本レイアウトを作成し、基本レイアウトに対して配置画像数等を変化させたレイアウト候補を作成し、最適なレイアウトを指定させる方法が提案されている。

20

**【先行技術文献】**

**【特許文献】**

**【0003】**

**【特許文献 1】特開 2006 - 287917 号公報**

**【特許文献 2】特開 2009 - 245071 号公報**

**【発明の概要】**

**【発明が解決しようとする課題】**

30

**【0004】**

しかしながら、特許文献 1 に記載の方法では、テンプレートに定められた向きに合致する画像を配置してレイアウトを生成する。このように、テンプレートに指定された条件に合致する画像群を抽出する方法では、テンプレートに定められた条件から外れるレイアウトは生成することができず、同じような結果のレイアウトしか得ることができないという問題があった。また、画像群の特徴を生かした好適なレイアウトを作成することができないという問題があった。

**【0005】**

一方、特許文献 2 に記載の方法では、様々なバリエーションのレイアウト結果が得られるが、選択された画像が好ましくない場合、良好なレイアウト結果が得られないという問題があった。

40

**【0006】**

本発明は、上述した事情に鑑み、様々なバリエーションの良好なレイアウト結果を得ることができる画像処理装置、画像処理方法、及びプログラムを提供することを目的とする。

**【課題を解決するための手段】**

**【0007】**

上述のような課題を解決するための本発明の画像処理装置は、複数の画像を含む画像群を取得する取得手段と、前記取得手段により取得した画像群から所定の数の画像を選択してテンプレートに配置するための処理を実行することにより、複数のレイアウト候補を作

50

成する作成手段と、前記作成手段により作成されたレイアウト候補の評価値を求める評価手段と、前記評価手段により求められた評価値と、所定のパラメータと、に基づいて、複数のレイアウト候補の中から、提示用の複数のレイアウト候補を選択するとともに、前記選択されたレイアウト候補の提示順を決定する決定手段と、前記決定手段により決定された提示順でレイアウト候補を提示する提示手段と、を備え、前記決定手段により、前記評価値が第1の値のレイアウト候補は、前記評価値が前記第1の値よりも低い第2の値のレイアウト候補よりも選択される確率が高くなり、且つ前記提示順は前記評価値の順とは異なる順となることを特徴とする。

【発明の効果】

【0008】

本発明では、複数のレイアウト候補を作成し、それらを評価してレイアウトリストを作成するため、好ましいレイアウト結果を得られる可能性が向上する。

【0009】

これにより、画像群に応じて全体整合のとれた良質なレイアウト結果を複数バリエーションを得ることができる。

【図面の簡単な説明】

【0010】

【図1】実施形態1に係る画像処理装置のハードウェア構成図である。

【図2】実施形態1に係るソフトウェアブロック図である。

【図3】実施形態1に係る画像解析処理のフローチャートである。

20

【図4】実施形態1に係る画像解析処理のフローチャートである。

【図5】実施形態1に係る人物グループ生成処理のフローチャートである。

【図6】実施形態1に係る自動レイアウト提案処理のフローチャートである。

【図7】実施形態1の人物グループの表示例を示す図である。

【図8】実施形態1の画像群の表示例を示す図である。

【図9】画像解析結果の保存形式の例を示す図である。

【図10】手動でお気に入り度を入力するためのUIの例及び手動でイベント情報を入力するためのUI例を示す図である。

【図11】手動で人物属性情報を入力するためのUI例を示す図である。

30

【図12】人物属性情報の保存形式の例を示す図である。

【図13】レイアウトテンプレートの一例を示す図である。

【図14】図13のレイアウトテンプレートの保持形式の例を示す図である。

【図15】レイアウトテンプレートの一例を示す図である。

【図16】図15のレイアウトテンプレートの保持形式の例を示す図である。

【図17】実施形態1における自動レイアウト生成処理のフローチャートである。

【図18】実施形態1における不要画像フィルタリング処理のフローチャートである。

【図19】自動トリミング処理の説明図、明るさ適正度の算出方法の説明図、及び彩度適正度の算出方法の説明図である。

【図20】トリミング欠け判定処理の説明図である。

【図21】実施形態1に係る自動レイアウト生成結果の表示例を示す図である。

40

【図22】決定されたテーマと主人公情報の保持例を示す図である。

【図23】決定されたテーマと主人公情報の保持例を示す図である。

【図24】生成したレイアウト情報の保持例を示す図である。

【図25】実施形態1に係る提案レイアウトのリストの決定処理のフローチャートである。

【図26】実施形態1に係る評価スコアに対する提案レイアウトの対応付けを制御するための確率分布を示す図である。

【図27】実施形態2の評価スコアに対する提案レイアウトの対応付けを制御するための確率分布を示す図である。

【図28】実施形態3に係る提案レイアウトのリストの決定処理のフローチャートである

50

。

【発明を実施するための形態】

【0011】

(実施形態1)

以下では、入力画像群を用いて自動でレイアウト出力物を生成するために、本発明における実施形態1について説明する。これはあくまで実施の1つの形態を例として示したものであり、本発明は以下の実施形態に限定されるものではない。

【0012】

図1は実施形態1の画像処理装置のハードウェア構成例を示すブロック図である。

【0013】

図1において、画像処理装置115は、CPU100と、ROM101と、RAM102と、2次記憶装置103と、表示装置104と、入力装置105と、IF107と、IF108と、無線LAN109を備えている。さらに、内部撮像デバイス106を備えている。これらは、制御バス/データバス110により相互に接続されている。

【0014】

画像処理装置115は、例えば、コンピュータである。CPU100(中央演算装置)は、実施形態1で説明する情報処理をプログラムに従って実行する。ROM101は、CPU100により実行される以下に示すアプリケーション等のプログラムが記憶されている。RAM102は、CPU100によるプログラムの実行時に、各種情報を一時的に記憶するためのメモリを提供している。2次記憶装置103は、ハードディスク等であり、画像ファイルや画像解析結果を保存するデータベース等を保存するための記憶媒体である。表示装置104は、例えば、ディスプレイであり、実施形態1の処理結果や以下に示すUI(User Interface)等をユーザに提示する装置である。表示装置104は、タッチパネル機能を備えても良い。入力装置105は、ユーザが画像補正の処理の指示等を入力するためのマウスやキーボード等である。

【0015】

また、内部撮像デバイス106で撮像された画像は、所定の画像処理を経た後、2次記憶装置103に記憶される。また、画像処理装置115は、インターフェース(IF108)を介して接続された外部撮像デバイス111から画像データを読み込むこともできる。さらに、無線LAN(Local Area Network)108はインターネット113に接続されている。画像処理装置115は、インターネット113に接続された外部サーバー114より画像データを取得することもできる。

【0016】

画像等を出力するためのプリンタ112は、IF107を介して画像処理装置115に接続されている。尚、プリンタ112はさらにインターネット上に接続されており、無線LAN109経由でプリントデータのやり取りをすることもできる。

【0017】

図2は本実施形態における上記アプリケーション等ソフトウェア構成のブロック図である。

【0018】

まずハードウェアが取得した画像データは、通常JPEG(Joint Photographic Expert Group)等の圧縮形式になっている。そのため、画像コードック部200は、該圧縮形式を解凍していわゆるRGB点順次のビットマップデータ形式に変換する。変換されたビットマップデータは、表示・UI制御部201に伝達され、ディスプレイ等の表示装置104上に表示される。

【0019】

上記ビットマップデータは、さらに画像センシング部203(アプリケーション)に入力され、同部において、詳細は後述するが、画像の様々な解析処理が行われる。上記解析処理の結果得られた画像の様々な属性情報は、所定の形式に従ってデータベース部202において、上述した2次記憶装置103に保存される。なお、以降においては、画像解析

10

20

30

40

50

処理とセンシング処理は同義で扱う。

**【0020】**

シナリオ生成部204(アプリケーション)では、ユーザが入力した様々な条件に応じて、自動で生成すべきレイアウトの条件を生成し、レイアウト生成部205(アプリケーション)では上記シナリオに従って、自動でレイアウトを生成する処理を行う。

**【0021】**

生成したレイアウトは、レンダリング部206で表示用のビットマップデータを生成し、該ビットマップデータは表示・UI制御部201に送られ、結果がディスプレイ等の表示装置104に表示される。一方で、レンダリング結果はさらにプリントデータ生成部207に送られ、同部でプリンタ用コマンドデータに変換され、プリンタに送出される。

10

**【0022】**

図3～6は、本実施形態のアプリケーションの基本的な画像処理のフローチャートである。具体的には、図3及び4は、画像センシング部203のフローを示しており、複数の画像データ群を取得して、それぞれについて解析処理を施し、その結果をデータベースに格納するまでの処理の流れを示している。図5は、検出した顔位置情報に基づいて、同じ人物と思われる顔情報をグループ化するための処理の流れを示している。図6は、画像の解析情報およびユーザが入力した様々な情報に基づいて、レイアウト作成のためのシナリオを決定し、該シナリオに基づいて、自動でレイアウトを生成するための処理の流れを示している。

**【0023】**

20

図3のS301では、1以上の画像データを含む画像データ群の取得を行う。画像データ群は、例えば、ユーザが、撮影画像が格納された撮像装置やメモリカードを画像処理装置115に接続して、これらから撮像画像を読み込むことで取得する。また、内部撮像装置で撮影され、2次記憶装置に保存されていた画像データ群を取得してもよい。あるいは、無線LANを介して、インターネット上に接続された外部サーバー114等、画像処理装置115以外の場所から画像データ群を取得してもよい。

**【0024】**

画像データ群を取得すると、そのサムネイル群が図8(a)や図8(b)に示すようにUIに表示される。図8(a)の801に示すように2次記憶装置103内のフォルダ単位で画像のサムネイル802を表示してもよいし、図8(b)に示すようにカレンダーのようなUI901で日付ごとに画像データが管理されていてもよい。日付の部分902をクリックすることにより、同日に撮影された画像を、図8(a)のようなサムネイル一覧で表示する。

30

**【0025】**

次に、S302において、各画像のデコードを行う。具体的には、アプリケーションが、新規で保存され未だセンシング処理が行われていない画像をサーチし、抽出された各画像について、画像コーデック部200が圧縮データからビットマップデータに変換する。

**【0026】**

次に、S303において、上記ビットマップデータに対して、各種センシング処理を実行する。ここでいうセンシング処理には、次の表1に示すような様々な処理が含まれる。本実施形態では、センシング処理の例として、顔検出、画像の特徴量解析、シーン解析を挙げており、それぞれ表1に示すようなデータ型の結果を算出する。

40

**【0027】**

【表1】

(表1)画像解析した結果得られる属性情報の例

センシング大分類	センシング小分類	データの型	値
画像基本特徴量	平均輝度	int	0~255
	平均彩度	int	0~255
	平均色相	int	0~359
顔検出	人物顔の個数	int	0~MAXFACE
	座標位置	int * 8	0~Width or Height
	顔領域内の平均Y	int	0~255
	顔領域内の平均Cb	int	-128~127
	顔領域内の平均Cr	int	-128~127
シーン解析	シーン結果	char	Landscape Nightscape Portrait Underexposure Others

## 【0028】

以下、それぞれのセンシング処理について説明する。

## 【0029】

画像の基本的な特徴量である全体の平均輝度、平均彩度は、公知の方法で求めればよいため、詳細な説明は省略する。平均輝度は、画像の各画素について、R G B 成分を公知の輝度色差成分（例えばY C b C r 成分）に変換し（変換式省略）、Y成分の平均値を求めればよい。また、平均彩度は、上記C b C r 成分について画素毎に以下を算出し、下記Sの平均値を求めればよい。

## 【0030】

## 【数1】

$$S = \sqrt{Cb^2 + Cr^2}$$

## 【0031】

また、画像内の平均色相（A v e H）は、画像の色合いを評価するための特徴量である。各画素毎の色相は、公知のH I S変換式を用いて求めることができ、それらを画像全体で平均化することにより、A v e Hを求めることができる。

## 【0032】

また、これらの特徴量は、上述したように画像全体で算出してもよいし、例えば、画像を所定サイズの領域に分割し、各領域毎に算出してもよい。

次に、人物の顔検出処理について説明する。本実施形態で使用する人物の顔検出手法としては、公知の方法を用いることができる。

## 【0033】

特開2002-183731号に記載されている方法では、入力画像から目領域を検出し、目領域周辺を顔候補領域とする。この顔候補領域に対して、画素毎の輝度勾配、および輝度勾配の重みを算出し、これらの値を、あらかじめ設定されている理想的な顔基準画像の勾配、および勾配の重みと比較する。そのときに、各勾配間の平均角度が所定の閾値

10

20

30

40

50

以下であった場合、入力画像は顔領域を有すると判定する。

【0034】

また、特開2003-30667号に記載されている方法では、まず画像中から肌色領域を検出し、同領域内において、人間の虹彩色画素を検出することにより、目の位置を検出することができる。

【0035】

特開平8-63597号に記載されている方法では、まず、複数の顔の形状をしたテンプレートと画像とのマッチング度を計算する。そのマッチング度が最も高いテンプレートを選択し、最も高かったマッチング度があらかじめ定められた閾値以上であれば、選択されたテンプレート内の領域を顔候補領域とする。同テンプレートを用いることで、目の位置を検出することができる。10

【0036】

さらに、特開2000-105829号に記載されている方法では、まず、鼻画像パターンをテンプレートとし、画像全体、あるいは画像中の指定された領域を走査し最もマッチする位置を鼻の位置として出力する。次に、画像の鼻の位置よりも上の領域を目が存在する領域と考え、目画像パターンをテンプレートとして目存在領域を走査してマッチングをとり、ある閾値よりもマッチ度が度置きい画素の集合である目存在候補位置集合を求める。そして、目存在候補位置集合に含まれる連続した領域をクラスタとして分割し、各クラスタと鼻位置との距離を算出する。その距離が最も短くなるクラスタを目が存在するクラスタと決定することで、器官位置の検出することができる。20

【0037】

その他の人間の顔検出方法としては、特開平8-77334、特開2001-216515、特開平5-197793、特開平11-53525、特開2000-132688、特開2000-235648、特開平11-250267に記載されるような顔および器官位置を検出する方法が挙げられる。また、人物の顔検出処理は、特許第2541688号に記載された方法でもよく、方法は特に限定されるものではない。

【0038】

人物の顔検出処理により、各入力画像について、人物顔の個数と各顔毎の座標位置を取得することができる。また、画像中の顔座標位置が分かることにより、顔領域の特徴量を解析することができる。例えば、顔領域毎に顔領域内に含まれる画素値の平均YCbCr値を求ることにより、顔領域の平均輝度および平均色差を得ることができる。30

【0039】

また、画像の特徴量を用いてシーン解析処理を行うことができる。シーン解析処理は、例えば、出願人が開示している特開2010-251999号や特開2010-273144号等で開示されている方法により行うことができる。シーン解析処理により、風景(Landscape)、夜景(Nightscape)、人物(Portrait)、露出不足(Underexposure)、その他(Others)、という撮影シーンを区別するためのIDを取得することができる。

【0040】

なお、本実施形態では、上記のセンシング処理によりセンシング情報を取得したが、その他のセンシング情報を利用してもよい。40

【0041】

上記のようにして取得したセンシング情報は、データベース部202に保存する。データベース部202への保存形式については、例えば、図9に示すような汎用的なフォーマット(例えば、XML: eXtensible Markup Language)で記述し、格納すればよい。

【0042】

図9においては、各画像毎の属性情報を、3つのカテゴリに分けて記述する例を示している。

【0043】

10

20

30

40

50

1番目のBaseInfoタグは、画像サイズや撮影時情報として、あらかじめ取得した画像ファイルに付加されている情報を格納するためのタグである。ここには、画像毎の識別子ID、画像ファイルが格納されている保存場所、画像サイズ、撮影日時などが含まれる。

#### 【0044】

2番目のSensInfoタグは、上述した画像解析処理の結果を格納するためのタグである。画像全体の平均輝度、平均彩度、平均色相やシーン解析結果が格納され、さらに、画像中に存在する人物の顔位置や顔色に関する情報が格納される。

#### 【0045】

3番目のUserInfoタグは、ユーザが画像毎に入力した情報を格納することができるタグであるが、詳細については後述する。

#### 【0046】

なお、画像属性情報のデータベース格納方法については、上記に限定されるものではない。その他公知のどのような形式で格納してもよい。

#### 【0047】

図3のS305では、上述したS302及びS303の処理を行った画像が最後の画像か否かを判定する。最後の画像である場合は、S306へ進み、最後の画像ではない場合は、S302へ戻る。

#### 【0048】

S306において、S303で検出された顔位置情報を用いて、人物毎のグループを生成する処理を行う。あらかじめ人物の顔を自動でグループ化しておくことにより、その後ユーザが各人物に対して名前を付ける作業を効率化することができる。

#### 【0049】

ここで的人物グループ形成は、公知の個人認識方法を用いて、図5の処理フローにより実行する。

#### 【0050】

なお、個人認識処理は、主に、顔の中に存在する眼や口といった器官の特徴量抽出と、それらの関係性の類似度を比較することにより実行される。個人認識処理は、例えば、特許第3469031号等に開示されているので、ここでの詳細な説明は省略する。

#### 【0051】

図5は人物グループ生成処理S306の基本的なフローチャートである。

#### 【0052】

まず、S501で、2次記憶装置に保存されている画像を順次読みだしてデコード処理を行う。さらにS502でデータベース部202にアクセスし、該画像中に含まれる顔の個数と顔の位置情報を取得する。次に、S504において、個人認識処理を行うための正規化顔画像を生成する。

#### 【0053】

ここで正規化顔画像とは、画像内に様々な大きさ、向き、解像度で存在する顔を切り出して、すべて所定の大きさと向きになるよう、変換して切り出した顔画像のことである。個人認識を行うためには、眼や口といった器官の位置が重要となるため、正規化顔画像のサイズは、上記器官が確実に認識できる程度であることが望ましい。このように正規化顔画像を生成することにより、特徴量検出処理において、様々な解像度の顔に対応する必要がなくなる。

#### 【0054】

次に、S505で、正規化顔画像から顔特徴量を算出する。ここでの顔特徴量とは眼や口、鼻といった器官の位置、大きさや、さらには顔の輪郭などを含むことを特徴とする。

#### 【0055】

さらに、S506で、あらかじめ人物の識別子（辞書ID）毎に顔特徴量が格納されているデータベース（以降、顔辞書と呼ぶ）の顔特徴量と類似しているか否かの判定を行う。類似度は、例えば、辞書ID内部で管理されている特徴量と、新たに入力された特徴量

10

20

30

40

50

を比較して算出する。ここで用いる特徴量は、保持されている目、鼻、口といった器官の位置や、器官間の距離等の情報である。類似度は、上記の特徴量が類似しているほど高く、類似していない場合には低い値を取るものとし、例えば0～100の値を取り得るものとする。そして、類似しているか否かの判定は、算出した類似度を予め保持されている閾値と比較し、類似度が閾値よりも高い場合には辞書IDと同一人物であると判断する。一方、類似度が閾値よりも低い場合には、同一人物ではないものとして判定する。このような類似度判定のための閾値は、全ての辞書IDに対して固定の値を一つだけ保持するようにいてもよいし、各辞書ID毎に異なった閾値を保持するようにしてもよい。

#### 【0056】

S506の判定がYesの場合S509に進み、同じ人物として同じ人物の辞書IDに該顔の特徴量を追加する。 10

#### 【0057】

S506の判定がNoの場合S508に進み、現在評価対象となっている顔は、これまで顔辞書に登録された人物とは異なる人物であると判断して、新規辞書IDを発行して顔辞書に追加する。S502～S509までの処理を、入力画像群の中から検出した顔領域全てに適用して、登場した人物のグループ化を行う。

#### 【0058】

人物グループ生成処理の結果は、図12のXMLフォーマットで示すように、各顔毎にIDタグを用いて記述し、上述したデータベースに保存する。

#### 【0059】

なお、上記実施形態においては、図3に示すように、全ての画像のセンシング処理が終了した後に人物グループ生成処理を実行したが、これ以外の方法としてもよい。例えば、図4に示すように、1つの画像に対してS403でセンシング処理を実行した後に、顔検出位置情報をを利用してグループ化処理S405を行うという作業を繰り返したとしても、同様の結果を生成することができる。

#### 【0060】

また、人物グループ生成処理によって得られた各人物グループは、図7のようなUI701にて表示されることになる。同7において、702は人物グループの代表顔画像を表示しており、その横には、該人物グループの名前を表示する領域703が存在する。自動の人物グループ化処理を終了した直後は、同図に示すように人物名は「No name 1」「No name 2」などと表示されている。これらの人物名を以下「人物ID」とする。また、704は該人物グループに含まれる複数の顔画像である。後述するが、図7のUI701においては、「No name X」の領域703を指定して人物名を入力したり、人物毎に誕生日や続柄等の情報を入力することができる。 30

#### 【0061】

また、上記センシング処理は、オペレーティングシステムのバックグラウンドタスクを利用して実行してもよい。この場合、ユーザはコンピュータ上で別の作業を行っていたとしても、画像群のセンシング処理を継続させることができる。

#### 【0062】

本実施形態においては、ユーザが手動で画像に関する様々な属性情報を入力することもできる。 40

#### 【0063】

その属性情報（以降、手動登録情報）の例の一覧を、表2に記載する。手動登録情報は大きく、画像単位に設定するものと、上記処理によりグループ処理した人物に設定する情報に分かれる。

#### 【0064】

【表2】

(表2)ユーザが手動入力可能な属性情報の例

分類	内容	データの型	値
画像	お気に入り度	int	0~5
	イベント	char	"travel""graduation" "wedding"
人物	名前	char	"NAME"
	誕生日	char	YYYYMMDD
	続柄	char	"family"

10

## 【0065】

まず、画像毎に設定する属性情報として、ユーザのお気に入り度がある。お気に入り度は、その画像を気に入っているかどうかを、ユーザが手動で段階的に入力するものである。例えば、図10(a)に示すように、UI1301上で、所望のサムネイル画像1302をマウスポインタ1303で選択し、右クリックをすることでお気に入り度を入力できるダイアログを表示する。ユーザはメニューの中で自分の好みに応じて、の数を選択することができる。本実施形態では、お気に入り度が高いほど の数が多くなるよう設定する。

20

## 【0066】

また、上記お気に入り度については、ユーザが手動で設定せずに、自動で設定するようにもよい。例えば、ユーザが図8(a)に示す画像サムネイル一覧表示の状態から、所望の画像ファイルをクリックし、1画像表示画面に遷移したとする。その遷移した回数を計測して、回数に応じてお気に入り度を設定してもよい。例えば、閲覧した回数が多いほど、ユーザが該画像を気に入っていると判断する。

30

## 【0067】

また、他の例として、プリント回数をお気に入り度に設定してもよい。例えば、プリント行為を行った場合、当然その画像を気に入っていると判断してお気に入り度が高いと設定すればよい。この場合は、プリント回数を計測して、プリント回数に応じてよりお気に入り度を設定する。

## 【0068】

以上説明したように、お気に入り度については、ユーザが手動で設定してもよく、閲覧回数に応じてお気に入り度を設定してもよく、プリント回数に応じてお気に入り度を設定してもよい。これらの設定及び計測した情報は、それぞれ個別に、図9で示すようなXMLフォーマットで、データベース部202のUser Infoタグ内に格納される。例えば、お気に入り度はFavoriteRateタグで、閲覧回数はViewingTimesタグで、プリント回数はPrintingTimesタグにそれぞれ格納される。

40

## 【0069】

また、画像毎に設定する別の情報として、イベント情報が挙げられる。イベント情報は、例えば、家族旅行 "travel"、卒業式 "graduation"、結婚式 "wedding" が挙げられる。

## 【0070】

イベントの指定は、図10(b)に示すように、カレンダー上で所望の日付をマウスポインタ1402などで指定して、その日のイベント名を入力することにより行うことができるようすればよい。指定されたイベント名は、画像の属性情報の一部として、図9に示すXMLフォーマットに含まれることになる。図9のフォーマットでは、User Infoタグ内のEventタグを使って、イベント名と画像を紐付けている。なお、以下、

50

「紐づけ」とは、関連付けることを指す。

#### 【0071】

次に、人物の属性情報について説明する。

#### 【0072】

図11は、人物の属性情報を入力するためのUIを示している。図11において、1502は所定人物（この場合は“father”）の代表顔画像を示している。1503は、所定人物の人物名（人物ID）の表示領域である。また、1504は、他の画像の中から検出し、S506で顔特徴量が類似していると判断された画像（サムネイル）である。このように、図11では、人物ID1503の下に、S506で顔特徴量が類似していると判断された画像1504の一覧が表示される。

10

#### 【0073】

センシング処理が終了した直後は、図7に示すように各人物グループには名前が入力されていないが、「No name」の部分702をマウスポインタで指示することにより、任意の人物名を入力することができる。

#### 【0074】

また、人物毎の属性情報として、それぞれの人物の誕生日やアプリを操作しているユーザから見た続柄を設定することもできる。図11の人物の代表顔1502をクリックすると、画面下部に図示するように、第1の入力部1505ではクリックした人物の誕生日を入力することができる。また、第2の入力部1506では、クリックした人物の続柄情報を入力することができる。

20

#### 【0075】

以上、入力された人物属性情報は、これまでの画像に関連付けられた属性情報とは異なり、図12のようなXMLフォーマットによって、画像属性情報とは別にデータベース部202内で管理される。

#### 【0076】

本実施形態では、あらかじめ用意した様々なレイアウトテンプレートを用いてレイアウト生成処理を行う。レイアウトテンプレートとは図13や図15に示すようなものであり、レイアウトする用紙サイズ上に、複数の画像配置枠1702、1902、1903（以降、スロットと同義）を備えている。

#### 【0077】

このようなテンプレートは多数用意されており、あらかじめ本実施例を実行するためのソフトウェアが画像処理装置115にインストールされた時点で、2次記憶装置103に保存しておけばよい。また、その他の方法として、IF107や無線LAN109を介して接続されたインターネット上に存在する外部サーバー114から、任意のテンプレート群を取得してもよい。

30

#### 【0078】

これらのテンプレートは汎用性の高い構造化言語、例えば上述したセンシング結果の格納と同様にXMLで記載されているものとする。XMLデータの例を図14及び図16に示す。

#### 【0079】

これらの例では、まずBASICタグに、レイアウトページの基本的な情報を記述する。基本的な情報とは、例えば該レイアウトのテーマやページサイズ、およびページの解像度（dpi）等が考えられる。同例Xにおいて、テンプレートの初期状態では、レイアウトテーマであるThemeタグはブランクとなっている。また、基本情報として、ページサイズはA4、解像度は300dpiを設定している。

40

#### 【0080】

また、ImageSlotタグは、上述した画像配置枠の情報を記述している。ImageSlotタグの中にはIDタグとPOSITIONタグの2つを保持し、画像配置枠のIDと位置を記述している。該位置情報については、図14や図16で図示するように、例えば左上を原点とするX-Y座標系において定義される。

50

**【0081】**

また、上記 `ImageSlot` は、その他にそれぞれのスロットに対して、スロットの形状および配置すべき推奨人物グループ名を設定する。例えば、図 13 のテンプレートにおいては、図 14 の `Shape` タグで示すように、すべてのスロットは矩形 “`rectangle`” 形状で、人物グループ名は `PersonGroup` タグによって “`MainGroup`” を配置することを推奨している。

**【0082】**

また、図 15 のテンプレートにおいては、図 34 に示すように、中央に配置している `ID = 0` のスロットは矩形形状であることが記載されている。また、人物グループは “`SubGroup`” を配置し、その他の `ID = 1, 2` と続くスロットは橢円 “`ellipse`” 形状で、人物グループは “`MainGroup`” を配置することを推奨している。  
10

**【0083】**

本実施形態においては、上述したようなテンプレートを多数保持する。

**【0084】**

本実施形態に係るアプリケーションは、入力された画像群に対して解析処理を実行し、人物を自動的にグループ化して UI で表示することができる。また、ユーザはその結果を見て、人物グループ毎に名前や誕生日などの属性情報を入力したり、画像毎にお気に入り度などを設定することができる。さらに、テーマごとに分類された多数のレイアウトテンプレートを保持することができる。

**【0085】**

20

以上の条件を満たす本実施形態のアプリケーションは、ある所定のタイミングで、自動的にユーザに好まれそうなコラージュレイアウトを生成し、ユーザに提示する処理を行う（以下、レイアウトの提案処理という）。

**【0086】**

以上の条件を満たすと、本実施例のアプリケーションは、ある所定のタイミングに、自動的にユーザに好まれそうなコラージュレイアウトを生成し、ユーザに提示する処理を行う。これを、レイアウトの提案処理と呼ぶこととする。

**【0087】**

図 6 は、レイアウトの提案処理を行うための基本的なフローチャートを示している。

**【0088】**

30

まず、S601において、レイアウトの提案処理のシナリオを決定する。シナリオには、提案するレイアウトのテーマ及びテンプレートの決定、レイアウト内で重視する人物（主人公）の設定、レイアウト生成に用いる画像群の選定情報などが含まれる。

**【0089】**

以下では、2つのシナリオを例示して、シナリオの決定方法について説明する。

**【0090】**

例えば、2週間前に自動的に各人物に関する誕生日のレイアウトの提案処理を行う設定がされていたとする。図 11 で自動グループ化されている人物 “son” の1歳の誕生日が近いとする。この場合には、提案するレイアウトのテーマは成長記録 “growth” と決定する。次にテンプレートの選択を行うが、この場合には成長記録に適した図 15 のようなものを選択し、図 22 に示すように、XML の `Theme` タグの部分に “`growth`” と記載する。次にレイアウトを行なう際に注目する主人公 “`MainGroup`” として、“`son`” を設定する。また、レイアウトを行なう際に副次的に注目する “`SubGroup`” として “`son`” と “`father`” を設定する。次に、レイアウトに利用するための画像群を選定する。この例の場合には、データベースを参照し、上記人物 “`son`” の誕生日からこれまでに撮影した画像群のうち、“`son`” を含む画像群を大量に抽出してリスト化する。以上が、成長記録レイアウトのためのシナリオ決定である。  
40

**【0091】**

上記とは異なる例として、1ヶ月以内に所定のイベント情報が登録されていた場合、レイアウトの提案処理を実行する設定がされているとする。図 10 (b) で登録したイベン  
50

ト情報から、例えば数日前に家族旅行に行きその画像が大量に2次記憶装置に保存されていることがわかると、シナリオ決定部は、家族旅行のレイアウトを提案するためのシナリオを決定する。この場合には、提案するレイアウトのテーマは旅行“travel”と決定する。次にテンプレートの選択を行うが、この場合には図13のようなレイアウトを選択し、図36に示すように、XMLのThemeタグの部分に“travel”と記載する。次にレイアウトを行う際に注目する主人公“Main Group”として、“son”、“mother”、“father”を設定する。このように、XMLの特性を活かせば、“Main Group”として複数の人物を設定することができる。次に、レイアウトに利用するための画像群を選定する。この例の場合には、データベースを参照し、上記旅行イベントに紐付けられた画像群を大量に抽出してリスト化する。以上が、家族旅行レイアウトのためのシナリオ決定である。

10

#### 【0092】

次に、図6のS603において、上述したシナリオに基づくレイアウトの自動生成処理を実行する。図17はレイアウト処理部の詳細な処理フローを示している。以降は、同図に沿って、各処理ステップの説明を行う。

#### 【0093】

まず、S2101で、上述したシナリオ生成処理で決定され、テーマと人物グループ情報が設定された後のテンプレート情報を取得する。

#### 【0094】

次に、S2103においては、上記シナリオで決定した画像リストに基づいて、各画像毎に該画像の特徴量をデータベースから取得し、画像群属性情報リストを生成する。ここでいう画像群情報リストとは、図9に示したIMAGEINFOタグが画像リスト分だけ並んだ構成となっている。

20

#### 【0095】

このように、本実施形態の自動レイアウト生成処理では、このように画像データそのものを直接扱うのではなく、あらかじめ画像毎にセンシング処理を行ってデータベース保存しておいた属性情報を利用する。レイアウト生成処理を行う際に、画像データそのものを対象とてしまうと、画像群を記憶するために非常に巨大なメモリ領域を必要としてしまうことを避けるためである。すなわち、これにより、レイアウト生成処理で必要なメモリ量を低減させることができる。

30

#### 【0096】

具体的には、まず、S2105において、入力された画像群の属性情報を用いて、入力された画像群の中から不要画像のフィルタリングを行う。フィルタリング処理は、図18のフローにて行う。図18では、各画像毎に、まずS2201で全体の平均輝度がある閾値( $ThY\_Low$ と $ThY\_High$ )内に含まれているかの判定を行う。否の場合はS2206に進み、注目画像はレイアウト対象から除去する。

#### 【0097】

同様に、S2202～S2205では、注目画像に含まれる顔領域それぞれについて、平均輝度、平均色差成分が、良好な肌色領域を示す所定閾値に含まれているかの判定を行う。S2202～S2205のすべての判定がYesとなる画像のみ、以降のレイアウト生成処理に適用される。具体的には、S2202では、ID=Nである顔領域のAvEYが所定閾値( $ThfY\_Low$ と $ThfY\_High$ )の範囲に含まれているか否かの判定を行う。S2203では、ID=Nである顔領域のAvEcHが所定閾値( $ThfY\_Low$ と $ThfY\_High$ )の範囲に含まれているか否かの判定を行う。S2204では、ID=Nである顔領域のAvEcRが所定閾値( $ThfY\_Low$ と $ThfY\_High$ )の範囲に含まれているか否かの判定を行う。S2205では、最後の顔であるか否かを判定する。最後の顔ではない場合は、S2202へ戻り、最後の顔である場合は、処理を終了する。

40

#### 【0098】

なお、このフィルタリング処理では、以降の一時レイアウト作成処理に明らかに不要と

50

判断できる画像の除去を目的としているため、上記閾値は比較的湯緩やかに設定することが望ましい。例えばS2201の画像全体輝度の判定において、ThY\_HighとThY\_Lowの差が画像ダイナミックレンジに比して極端に小さいと、それだけYesと判定される画像が少なくなってしまう。本実施形態のフィルタリング処理ではそうならないよう、両者の差ができる限り広く設定し、かつ明らかに異常画像と判断されるものは除去できるような閾値に設定する。

#### 【0099】

次に図17のS2107において、上記処理でレイアウト対象となった画像群を用いて、大量(L個)の一時レイアウトを生成する。一時レイアウトの生成は、取得したテンプレートの画像配置枠に対して、入力画像を任意に当てはめる処理を繰り返す。このときに、10 例えば、以下のパラメータ(画像選択・配置・トリミング)をランダムで決定する。

#### 【0100】

画像選択基準としては、例えば、レイアウト中の画像配置枠がN個の時、画像群の中からどの画像を選択するかが挙げられる。配置基準としては、例えば、選択した複数の画像を、どの配置枠に配置するかが挙げられる。トリミング基準としては、配置した際に、どの程度のトリミング処理を行うかというトリミング率が挙げられる。トリミング率は例えば0~100%で表わされ、トリミングは、図19(a)に示すように、画像の中心を基準として所定のトリミング率で行われる。図19(a)では、2301は画像全体を示し、2302はトリミング率50%でトリミングした際の切り取り枠を示している。

#### 【0101】

上述したような画像選択・配置・トリミング基準に基づいて、可能な限り数多くの一時レイアウトを生成する。生成した各一時レイアウトは、図37のXMLのように表わすことができる。各スロットに対して、選択され配置された画像のIDがImageIDタグに記述され、トリミング率がTrimmingRatioタグに記述される。

#### 【0102】

なお、ここで生成する一時レイアウトの数Lは、後述するレイアウト評価ステップでの評価処理の処理量と、それを処理する画像処理装置115の性能に応じて決定されるが、例えば、数十万通り以上の一時レイアウトを生成するのが好ましい。

#### 【0103】

レイアウト評価ステップでの評価処理の処理量は、作成するレイアウトテンプレートの複雑度合いに応じて増減する。例えば、テンプレート内で取り扱うスロット数が多いほど評価処理量は増加し、また、各スロットに指定されたレイアウト条件が複雑であるほど評価処理量は増加する。したがって、生成しようとしているテンプレートの複雑度合をあらかじめ見積もって、それにより動的にLを決定してもよい。

#### 【0104】

以上述べたように適切にLを設定することで、自動レイアウト作成時のレスポンスとレイアウト結果の品質を最適にコントロールできる。

#### 【0105】

生成したレイアウトは、それぞれIDを付加して図24のXML形式で2次記憶装置103にファイル保存してもよいし、構造体など別のデータ構造を用いてRAM102上に記憶してもよい。

#### 【0106】

次に、図17のS2108において、大量に生成した一時レイアウトの定量評価を行う。具体的には、作成したL個の一時レイアウトに対して、それぞれ所定のレイアウト評価量を用いて評価を行う。本実施形態におけるレイアウト評価量の一覧を、表3に示す。表3に示すように、本実施形態で用いるレイアウト評価量は、主に3つのカテゴリに分けることができる。

#### 【0107】

10

20

30

40

【表3】

(表3)自動レイアウトを行う際のレイアウト評価値の例

カテゴリ	内容	スコアレンジ	テーマ毎重要度(重みW)		
			growth	travel	...
画像個別評価	明るさ適正度	0~100	0.5	1.0	
	彩度適正度	0~100	0.5	1.0	
画像・スロット適合度評価	人物適合度	0~100	1.0	0.5	
	トリミング欠け判定	0~100	1.0	0.5	
ページ内バランス評価	画像類似性	0~100	0.5	1.0	
	色合いばらつき	0~100	0.5	1.0	
	顔サイズばらつき	0~100	0.5	1.0	
その他	ユーザー嗜好性	0~100	0.8	0.8	

## 【0108】

一つ目は、画像個別の評価量である。これは画像の明るさや彩度、ブレほけ量等の状態を判断し、スコア化するものである。以下、本実施形態のスコア化の一例について説明する。明るさの適正度は、図19(b)に示すように、平均輝度がある所定レンジ範囲においてはスコア値100とし、所定レンジ範囲から外れるとスコア値を下げるよう設定する。彩度の適正度は、図19(c)に示すように、画像全体の平均彩度がある所定の彩度値より大きい場合にはスコア値100とし、所定値より小さい場合にはスコア値を除々に下げるよう設定する。

## 【0109】

二つ目は、画像とスロットの適合度の評価である。画像とスロットの適合度の評価としては、例えば、人物適合度、トリミング欠け判定が挙げられる。人物適合度は、スロットに指定されている人物と、実際に該スロットに配置された画像内に存在する人物の適合率を表したものである。例を挙げると、あるスロットが、XMLで指定されているPersonGroupで、“father”、“son”が指定されているものとする。この時、該スロットに割り当てられた画像に該2人の人物が写っていたとすると、該スロットの人物適合度はスコア値100とする。片方の人物しか写っていなかったとすると、適合度はスコア値50とし、両者とも写っていなかった場合は、スコア値0とする。ページ内の適合度は、各スロット毎に算出した適合度の平均値とする。トリミング領域2702の欠け判定は、例えば、図20に示すように、画像中に存在する顔の位置2703が判明している場合、欠けた部分の面積に応じて、0から100までのスコア値を算出する。欠けた面積が0の場合、スコアは100とし、逆にすべての顔領域が欠けた場合、スコア値は0とする。

## 【0110】

三つめは、レイアウトページ内のバランスを評価である。バランスを評価するための評価値としては、例えば、画像類似性、画素値分布のばらつき、オブジェクトのばらつきが挙げられる。

## 【0111】

レイアウト頁内のバランスを評価するための評価量として、画像類似性について説明する。画像の類似性は、大量に生成した一時レイアウト毎に算出されるレイアウト頁内のそれぞれの画像の類似性である。例えば、旅行テーマのレイアウトを作成したい場合、あまりに似通った類似度の高い画像ばかりが並んでいたとすると、それは良いレイアウトとは言えない場合がある。したがって、例えば、類似性は、撮影日時によって評価することができる。撮影日時が近い画像は、同じような場所で撮影された可能性が高いが、撮影日時が離れていれば、その分、場所もシーンも異なる可能性が高いからである。撮影日時は、

10

20

30

40

50

図9で示したように、画像属性情報として、予めデータベース部202に保存されている、画像毎の属性情報から取得することができる。撮影日時から類似度を求めるには以下のようない算を行う。例えば、今注目している一時レイアウトに表4で示すような4つの画像がレイアウトされているものとする。

#### 【0112】

なお、画像IDで特定される画像には、それぞれ撮影日時情報が付加されている。具体的には、撮影日時として、年月日及び時間（西暦：YYYYYY、月：MM、日：DD、時：HH、分：MM、秒：SS）が付加されている。このとき、この4つの画像間で、撮影時間間隔が最も短くなる値を算出する。

#### 【0113】

#### 【表4】

(表4)

画像ID	撮影日時(YYYYYYMMDD:HHMMSS)
25	20100101:120000
86	20100101:150000
102	20100101:170000
108	20100101:173000

#### 【0114】

この場合は、画像ID102と108間の30分が最も短い間隔である。この間隔をMinIntervalとし、秒単位で格納する。すなわち30分 = 1800秒である。このMinIntervalをL個の各一時レイアウト毎に算出して配列stMinInterval[1]に格納する。次に、該stMinInterval[1]の中で最大値MaxMinInterval値を求める。すると、1番目の一時レイアウトの類似度評価値Similarity[1]は以下のようにして求めることができる。

$$\text{Similarity}[1] = 100 \times \text{stMinInterval}[1] / \text{MaxMinInterval}$$

すなわち、上記Similarity[1]は、最小撮影時間間隔が大きいほど100に近づき、小さいほど0に近づく値となっているため、画像類似度評価値として有効である。

#### 【0115】

レイアウト頁内のバランスを評価するための評価量として、画素値分布のばらつきについて説明する。ここでは、画素値分布のばらつきとして、色合いのバラつきを例に挙げて説明する。例えば旅行テーマのレイアウトを作成したい場合、あまりに似通った色（例えば、青空の青、山の緑）の画像ばかりが並んでいたとすると、それは良いレイアウトとは言えない場合がある。この場合は、色のばらつきの大きいものを高い評価とする。注目している1番目の一時レイアウト内に存在する画像の平均色相AveHの分散を算出して、それを色合いのバラつき度tmpColorVariance[1]として格納する。次に、tmpColorVariance[1]の中での最大値MaxColorVariance値を求める。すると、1番目の一時レイアウトの色合いバラつき度の評価値ColorVariance[1]は以下のようにして求めることができる。

$$\text{ColorVariance}[1] = 100 \times \text{tmpColorVariance}[1] / \text{MaxColorVariance}$$

上記ColorVariance[1]は、ページ内に配置された画像の平均色相のバラつきが大きいほど100に近づき、小さいほど0に近づく値となる。したがって、色合

10

20

30

40

50

いのばらつき度評価値として用いることができる。

**【0116】**

画素値分布のばらつきは、上述したものに限定されるものではない。

**【0117】**

レイアウト頁内のバランスを評価するための評価量として、オブジェクトのバラつきについて説明する。ここでは、オブジェクトのバラつきとして、顔の大きさのバラつきを例に挙げて説明する。例えば、旅行テーマのレイアウトを作成したい場合、レイアウト結果を見て、あまりに似通った顔のサイズの画像ばかりが並んでいたとすると、それは良いレイアウトとは言えない場合がある。レイアウト後の紙面上における顔の大きさが、小さいものもあれば大きいものもあり、それらがバランスよく配置されていることが、良いレイアウトと考える。この場合は、顔のサイズのばらつきの大きいものを高い評価とする。注目している1番目の一時レイアウト内に配置された後の顔の大きさ（顔位置の左上から右下までの対角線の距離）の分散値を、`tmpFaceVariance[1]`として格納する。次に、該`tmpFaceVariance[1]`の中での最大値`MaxFaceVariance`値を求める。すると、1番目の一時レイアウトの顔サイズバラつき度の評価値`FaceVariance[1]`は、以下のようにして求めることができる。

$$\text{FaceVariance}[1] = 100 \times \text{tmpFaceVariance}[1] / \text{MaxFaceVariance}$$

上記`FaceVariance[1]`は、紙面上に配置された顔サイズのバラつきが大きいほど100に近づき、小さいほど0に近づく値となる。したがって、顔サイズのバラつき度評価値として用いることができる。

**【0118】**

オブジェクトのバラつきは、上述したものに限定されるものではない。

**【0119】**

またその他カテゴリとして、ユーザの嗜好性評価が挙げられる。

例えば、上記に述べた各種の評価量による評価値が低いレイアウトであっても、ユーザが個人的に気に入った写真が含まれるレイアウトであれば、そのユーザにとっては良いレイアウトとなる場合もある。そのようなレイアウトの選択を阻害しないよう、ユーザ嗜好性を元にした評価量を用いて評価するのが好ましい。

**【0120】**

上述したように、ユーザは各画像に対して事前にお気に入り度を設定することができるため、これに基づいて嗜好性を評価することができる。また、例えば、自動で閲覧回数や閲覧時間に基づいて嗜好性を評価することもできる。これらの評価に用いる情報は、`FavorRate`タグ、`ViewingTimes`タグ、`PrintingTimes`タグで管理されている情報をもとにして算出できる。

**【0121】**

1番目の一時レイアウトについて、各スロットに配置された画像のタグ情報の各々の数値について全ての画像の平均値`FavorRateAve[1]`、`ViewingTimesAve[1]`、`PrintingTimesAve[1]`を求める。

**【0122】**

次に、これらの総和を取ることで、ユーザの嗜好性評価値`UserFavor[1]`は、以下のようにして求めることができます。

$$\text{UserFavor}[1] = \text{FavorRateAve}[1] + \text{ViewingTimesAve}[1] + \text{PrintingTimesAve}[1]$$

上記`UserFavor[1]`は、ユーザのお気に入り度が高い画像、閲覧回数が多い画像、印刷回数の多い画像を用いたレイアウトほど高い値を示す。したがって、本実施形態では、`UserFavor[1]`の値が高いほどレイアウトに対するユーザ嗜好性が高いと判断する。

**【0123】**

以上説明したような、各一時レイアウト毎に算出した複数の評価値を、以下では統合化

10

20

30

40

50

して、各一時レイアウト毎のレイアウト評価値とする。1番目の一時レイアウトの統合評価値を、EvalLayout[1]とし、上記で算出したN個の評価値（表3の評価値それぞれを含む）の値を、EvalValue[n]とする。このとき、統合評価値は以下で求めることができる。

【0124】

【数2】

$$EvalLayout[l] = \sum_{n=0}^N EvalValue[n] \times W[n]$$

【0125】

10

上式において、W[n]は、表3で示したシーン毎の各評価値の重みである。この重みはレイアウトのテーマ毎に異なる重みを設定する。例えば、表3に示すようにテーマを成長記録“growth”と旅行“travel”で比較した場合、旅行テーマの方は、できるだけ良質の写真をいろいろな場面で数多くレイアウトすることが望ましい場合が多い。このため、画像の個別評価値やページ内のバランス評価値を重視する傾向に設定する。一方、成長記録“growth”的場合、画像のバリエーションよりは、成長記録の対象となる主人公が確実にスロットに適合しているか否かが重要である場合が多い。このため、ページ内バランスや画像個別評価よりも、画像・スロット適合度評価を重視する傾向に設定する。なお、本実施形態におけるテーマ毎の重要度は表3に示すように設定した。このようにして算出したEvalLayout[1]を用いて、S2109では、レイアウト結果表示のためのレイアウトリストLayoutList[k]を生成する。レイアウトリストLayoutList[k]の作成方法については、詳細は後述する。このレイアウトリストLayoutList[k]は、図6のS605でレンダリング表示を行う際にレイアウト順番をk番目とした時に、評価が終了したレイアウトの順番1との対応付けを行うのに用いられる。LayoutList[k]の中身は、レイアウト順番1の値が、記載されている。このレイアウトリストLayoutList[k]のkが小さい方から、順に表示が行われる。

20

【0126】

図6に戻って、上記処理によって得られたレイアウト結果を、図6のS605でレンダリングした結果を図21のように表示する。S605では、まずLayoutList[0]に格納されているレイアウト識別子を読み出し、識別子に相当する一時レイアウト結果を、2次記憶装置103あるいはRAM102上から読み出す。レイアウト結果には、上述したようにテンプレート情報とテンプレート内に存在するスロット毎に、割り当てられた画像名が設定されている。したがって、これらの情報に基づいて、画像処理装置115上で動作するOSの描画関数を用いて、該レイアウト結果をレンダリングし、図21の2902のように表示することになる。

30

【0127】

図21では、Nextボタン2904を押下することにより、次点スコアであるLayoutList[1]の識別子を読み出し、上記と同様にレンダリング後、表示を行う。これにより、ユーザは様々なバリエーションの提案レイアウトを閲覧することができる。また、Previousボタン2903を押下することにより、前に表示したレイアウトを再表示することができる。さらに、表示されたレイアウトが気に入った場合には、プリントボタン2905を押下することで、画像処理装置115に接続されたプリンタ112からレイアウト結果2902をプリントアウトすることができる。

40

【0128】

ここで、図25及び図26を用いて、図17のS2109のレイアウトリストLayoutList[k]の作成方法について説明する。図25は、実施形態1に係る提案レイアウトのリストの決定処理のフローチャートであり、図26は、実施形態1に係る評価スコア（評価値）に対する提案レイアウトの対応付けを制御するための確率分布を示す図である。

50

**【 0 1 2 9 】**

図25において、S3301では、レイアウト評価スコア順にソートリストを作成する。具体的には、EqualLayoutのうち評価値の高い順に識別子1をならべ記憶した、ソートリストLayoutList[0]を作成する。たとえば最も良いスコアを出したものが $l = 50$ 番目に作成した一時レイアウトだった場合は、tmpSortLayoutList[0] = 50となる。同様にtmpSortLayoutList[1]は、スコア値が2番目の識別子1、というようにtmpSortLayoutList[N]にスコア値がN番目の識別子1を順に記憶する。

**【 0 1 3 0 】**

S3302では、カウンタpを初期化( $p = 0$ )する。

10

**【 0 1 3 1 】**

次にS3303において、乱数値Qを取得する。乱数値Qは、詳細は後述するが、レイアウト候補を選択する際に用いる。乱数値Qは、例えば、0から $L - 1$ の間の範囲の整数( $L$ は、一時レイアウトの総数)が不均一な確率で発生する特殊な乱数である。図26は、本実施形態で用いる乱数値Qであり、横軸は、発生させる乱数値Qを示しており、縦軸は横軸の数値の発生確率を示している。図26では、発生確率が、 $Q = 0$ の場合が最大で、 $Q = L - 1$ で最小となる直線となるような関数となっている。S3303では、乱数値Qを取得する度に、この発生確率に基づいて、異なる乱数値Qを発生させる。

**【 0 1 3 2 】**

図25のS3304では、取得した乱数値Qを元にtmpSortLayoutList[0]のQ番目の値を、LayoutList[p]に記憶させる。

20

$LayoutList[p] = tmpSortLayoutList[Q]$

となる。すなわち、最初(カウンタp = 0の場合)のLayoutListには、スコア値が上位からQ番目のレイアウトを候補として登録する。

**【 0 1 3 3 】**

S3305において、カウンタpを1進め、S3306へ進む。

**【 0 1 3 4 】**

S3306において、カウンタpが総提示レイアウト数O以下なら、S3303に戻り、再度乱数値Qの取得を行う。ここで、総提示レイアウト数Oとは、ユーザ提示のために事前に準備しておくレイアウトの個数である。総提示レイアウト数Oは、任意の値を設定することができ、ROM101等に予め設定されていてもよいし、ユーザが設定するようにもよい。その後、S3304において、S3303で新たに取得した乱数に基づいたQ番目のスコア値のレイアウトが2番目のレイアウトリストに登録される。S3305でカウンタpを1つ進め、S3306で再び総提示レイアウト数Oとの比較を行う。このように、S3303～S3305を総提示レイアウト数O個分のレイアウトリストが作成されるまで繰り返す。レイアウトリストがO個準備できたら終了する。

30

**【 0 1 3 5 】**

上述したように、本実施形態では、レイアウト評価順序と乱数を用いてレイアウトリストを作成する。このレイアウトリストは、ユーザが本発明に係るアプリケーションを起動したタイミング毎に作成する(更新する)。このとき、用いる乱数の初期値をPCのタイムスタンプ等を用いて変更することにより、毎回異なるレイアウトリストを作成することが可能となる。そして、レイアウトリストの作成結果に基づき、ユーザに対しレイアウトを提示する。

40

**【 0 1 3 6 】**

ここで、スコア順が近いレイアウトは、似通ったレイアウト結果になる可能性がある。例えば、評価スコアの算出方法からわかるように、同じレイアウトテンプレートを用いた場合等のように同じレイアウトパターンで、一部の画像が異なる場合には、双方の評価スコアは近いものとなる。具体例としては、図15に示すレイアウトパターンにおいて、中心のID = 0に同じ画像がレイアウトされ、ID = 1以降の画像のうち1画像だけが異なるレイアウトの場合、双方の評価スコアは近くなる。したがって、レイアウトリストをス

50

コア順に作成してユーザに画像を提示する場合、似通ったレイアウトを順次提示する可能性が高い。

#### 【0137】

これに対し、本実施形態では、レイアウトリスト作成には、乱数を用いてスコア順にソートされたレイアウト候補から所定のレイアウト候補を選択する。すなわち、レイアウトリストには、レイアウト候補の評価スコア順位順のうち、乱数で選択された順にレイアウトが設定される。このため、スコア値の近いレイアウトが続くこと可能性が低くなり、ユーザに似通ったレイアウトが連続して提示される可能性を低下させることができる。すなわち、連続してレイアウトを提示する際に、評価スコアが異なるレイアウト、言い換れば、似通っていないレイアウトを順に提示する可能性が高くなる。

10

#### 【0138】

このように、本実施形態では、評価結果であるスコア順に、レイアウトリストを作成し、このスコア順と乱数を用いて、ユーザに提示するレイアウトリストを決定する。このように、ユーザに提示する提示順を評価値の順とは異なる順となるように決定することにより、似通っていないレイアウトを順に提示する可能性を高くすることができる。

#### 【0139】

すなわち、本実施形態によれば、アプリケーションを使用するたびに、異なるバリエーションのレイアウトを提供できる。これにより、ユーザを飽きさせず、同一画像群を楽しみつつ、かつ自動的に好適なレイアウトを提供することができる。

#### 【0140】

また、本実施形態では、レイアウト候補の選択に用いる乱数は、図26に示すように、スコア上位のレイアウトが高確率で選択されるような確率分布を有している。したがって、評価スコア上位のレイアウトが高確率で選択される可能性が高く、高品質のレイアウトを自動的にユーザに提供できる。

20

#### 【0141】

想定されるあらゆる画像群入力に合わせてテンプレートを大量に用意しておき、画像群入力の特徴に応じてテンプレートを自動／手動で選択してレイアウトを作成した場合大量のテンプレートを保持するコストがかかる。また、画像群入力の特徴に応じて適切なテンプレートを何らかの手段で選ぶ必要がある。さらに、一度作成したレイアウト結果とは異なるレイアウト結果を得たい場合には、異なるテンプレートを有する必要がある。これに対し、本実施形態では、レイアウト評価順序と乱数を用いてレイアウトリストを作成することにより、低コストで異なるバリエーションのレイアウトを容易に提供することができる。

30

#### 【0142】

また、単一のレイアウト結果だけでなく、複数のレイアウト候補を用意することで、レイアウトのバリエーションを増やす際に、複数のレイアウト候補間では選択される画像と同じとした場合、画像によりレイアウトが左右されてしまう。具体的には、選択されている画像自体が好ましくない場合には、それをベースに大きさや配置位置を変更したとしてもどのレイアウト候補も好ましくないものである可能性が高い。これに対し、本実施形態では、レイアウト評価順序と乱数を用いてレイアウトリストを作成することにより、異なる画像が選択されたレイアウトリストを表示することができる。したがって、より良好なレイアウト結果を得る可能性が高くなる。

40

#### 【0143】

##### (実施形態2)

本実施形態では、図25のS3303で取得される乱数値Qとして、図27に示すような乱数を用いる。なお、実施形態1と同様の部分については、説明を省略する。図27は、実施形態2に係るレイアウト作成において用いる乱数の確率分布を示す図である。

#### 【0144】

図27(A)は、図26と同様に、0からL-1の間の範囲の整数(Lは、一時レイアウトの総数)が、不均一な確率で発生する特殊な乱数である。図27(A)において、横

50

軸は発生させる乱数値Qを示しており、縦軸は横軸の乱数値Qの発生確率を示している。乱数値Qは、 $Q = 0$ の場合が発生確率が最大で、 $Q = L'$  ( $< L - 1$ )で発生確率が0となる直線と、 $L' < Q (< L - 1)$ では発生確率が0となるような関数である。本実施形態でも、実施形態1と同様に、乱数値Qを取得する度に、この発生確率に基づいて、異なる乱数値Qを発生させる。

#### 【0145】

そして、乱数値Qの結果は、図25に示すフローの通り、評価スコアが上位からQ番目の順位の候補レイアウトをレイアウトリストに追加する。

#### 【0146】

ここで、乱数値Qに基づいて所定の候補レイアウトをレイアウトリストに追加するため、乱数値Qの取りうる範囲によって、候補レイアウトとして選択する評価スコア順位の範囲を限定することが可能である。図25(A)のような特性の発生確率の乱数を用いた場合、発生する乱数値Qのレンジが $0 \sim L'$ のみに限定されており、結果として評価スコアが上位 $L'$ 番目までのレイアウトのみがレイアウトリストに登録されることになる。

10

#### 【0147】

本実施形態では、レイアウトリストに追加されるレイアウトの評価スコアが上位 $L'$ 目までとなることにより、評価順位が下位のレイアウト、すなわち、低い評価スコアのレイアウトがレイアウトリストに登録されるのを防ぐことができる。したがって、評価値の近いレイアウトが連続することを防止しつつ、実施形態1と比較してより高評価のレイアウトのみをユーザに提示できる。この結果、高評価で飽きのこない連続した自動レイアウト結果の提示ができる。

20

#### 【0148】

また、図27(B)は、図26と同様に、0から $L - 1$ の間の範囲の整数( $L$ は、一時レイアウトの総数)が、不均一な確率で発生する特殊な乱数である。図27(B)の横軸は発生させる乱数値Qを示しており、縦軸は横軸の乱数値Qの発生確率を示している。図27(B)に示す発生確率曲線は、正規乱数分布となっている。正規分布乱数は、単純乱数から生成できる乱数であり、0を中心とした正規分布の形状の発生頻度分布とことができ、発生レンジも調整も制御可能な乱数発生アルゴリズムである。図27(B)の発生確率曲線は、正規分布乱数を発生し、レンジは $\pm M'$ を超える値を有する範囲とし、発生した正規分布乱数のうち、レンジが $0 \sim M' - 1$ のもののみを選び、選ばれたレンジの範囲で正規化(発生頻度総和が1)となるようにしたものである。図27(B)の発生確率曲線は、図27(A)と同様に、0に行くほど高確率であり、 $M'$ 以上の値は発生しない。図27(B)の乱数値Qを用いてレイアウト作成を行った場合の基本的な効果は、図27(A)の乱数値Qを用いた場合と同様であるが、図27(A)の乱数値Qを用いた場合と比較してより高スコアのレイアウトが選択される。

30

#### 【0149】

図27(C)は、 $0 \sim M$ までの数値の発生確率を標準乱数とした例である。図27(C)では、図27(A)及び図27(B)と同様に、評価スコアの順位の上位側が低順位側に対して、高確率となる。標準乱数は、他の乱数に比べ比較的その実装が簡易である。したがって、実装上、メモリに制約があるケースなどは、この乱数を用いることでメモリの使用量を低減させることができる。ただし、この場合、評価スコアの順位が低順位のレイアウトまで高順位のレイアウトと同確率で出現する可能性があるので、 $M$ はあまり大きな値ではないのが好ましい。例えば、 $M < (L - 1) / 2$ がより好ましい。このような乱数値Qを用いた場合も、評価値が近いレイアウトが連続することを防止しつつ、高評価のレイアウトをユーザに提示できる。結果、飽きのこない連続した有効な自動レイアウト結果の提示が可能となる。

40

#### 【0150】

上述したように、図27(A)～(C)では、評価スコアの順位の上位側が下位側に対して、高確率になるようにしている。これにより、評価値の近いレイアウトが連続することを防止しつつ、高評価のレイアウトのみをユーザに提示でき、結果、飽きのこない連続

50

した有効な自動レイアウト結果の提示が可能となる。

**【0151】**

すなわち、本実施形態では、候補レイアウトのうち、評価スコアの順位が上位にあるレイアウトを選択してレイアウトリストに登録し、ユーザに自動提案することができる。これにより、ユーザは、より良いレイアウトを選択することが可能となる。また、それらの候補リストを所定のタイミングで更新することにより、飽きのこないレイアウトを楽しむことが可能となる。さらに、提案順番と提案されるレイアウトの評価スコアの順位とが一致しないようにレイアウトリストを作成しているため、レイアウトを提示する際に同じようなレイアウトが連続せず、より飽きのこないレイアウト結果を楽しむことができる。

**【0152】**

(実施形態3)

図28を用いて本実施形態のレイアウトリスト作成について説明する。図28は、図17のS2109のレイアウトリスト作成の詳細フローである。なお、本実施形態では、実施形態1と重複する説明は省略する。

**【0153】**

S3601において、レイアウト評価スコア(Eval Layout値)の度数分布(ヒストグラム)データベースを作成する。表3に、作成される度数分布データベースの一例を示す。インデックスナンバーR(列1)に対して、対応する度数分布のスコア範囲(列2)、該当スコア範囲に入ったレイアウト候補のリスト個数(列3)がデータベース化される。さらに、該当スコア範囲までの累積度数(%) (列4)、該当スコア範囲に入ったレイアウトリスト番号1(列5)が、データベース化される。

**【0154】**

**【表5】**

(表5) レイアウト候補画像の評価スコア度数分布表

列1 インデックスR	列2 スコア範囲	列3 リスト数S	列4 累積度数(%)	列5 リスト番号
0	~100	3	2	4, 60, 134, ..
1	~95	8	6	15, 45, ..
2	~90	6	9	..
3	~85	10	14	..
4	~80	18	23	..
5	~75	15	30	..
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
18	~10	5	100	..
19	~5	0	100	..

**【0155】**

S3602で、評価用に作成した一時レイアウト総数のうち、レイアウトリストに追加対象とするレイアウトを評価値の上位何(%)か指定するリスト個数閾値(%)を取得する。本実施形態では、例えば、T = 30(%)とする。

10

20

30

40

50

**【 0 1 5 6 】**

S 3 6 0 3 で、S 3 0 6 1 で作成したデータベース 3 6 1 0 を参照し、乱数発生範囲 r 及び総提示レイアウト数 O を設定する。本実施形態では、DB 中の列 4 の累積度数中で、S 3 6 0 2 で設定した T (= 3 0 ) % 以上となるインデックス値 R (= 5 ) を乱数発生範囲 r (= 5 ) として設定する。

**【 0 1 5 7 】**

次に、S 3 6 0 4 では、カウンタ p を初期化 ( P = 0 ) する。

**【 0 1 5 8 】**

S 3 6 0 5 では、乱数値 Q を取得する。このとき、乱数値 Q は 0 以上、乱数発生範囲 r 以下の値とする。S 3 6 0 3 で設定された乱数範囲 r (= 5 ) を元に 0 から 5 の乱数を発生させる。ここで発生乱数の確率分布形状は図 2 6 および図 2 7 ( A ) ~ ( C ) に示したような、乱数値 Q の範囲が 0 ~ r (= 5 ) であり、乱数値 Q の値が小さい方 ( 0 側 ) の発生確率が高い乱数を用いる。

**【 0 1 5 9 】**

S 3 6 0 6 では、データベースを参照してインデックス R に対するリスト番号を取得する。ここでは、S 3 6 0 5 で発生した乱数値 Q (= 2 ) を元に、再びデータベース 3 6 1 0 を参照し、乱数値 Q に対応するインデックス R の行のリスト番号値を取得する。

**【 0 1 6 0 】**

S 3 6 0 7 では、S 3 6 0 6 で取得したリスト番号のいずれか一つを LayoutList [ p ] に記憶させる。

**【 0 1 6 1 】**

S 3 6 0 8 において、カウンタ p を 1 進め、S 3 6 0 9 において、カウンタ p が総提示レイアウト数 O 以下なら、S 3 6 0 5 に戻り、再度乱数値 Q の取得を行う。ここで、総提示レイアウト数 O は、任意に設定される値で、ユーザ提示のために事前準備しておくレイアウトの個数を示している。総提示レイアウト数 O 個分のレイアウトリストが作成されるまで、S 3 3 0 3 ~ S 3 3 0 5 を繰り返す。そして、レイアウトリストが O 個準備できたら終了する。本実施形態では、レイアウトリストの作成には、レイアウト候補の評価スコア範囲は、乱数を用いて選択される。ここで、上述した通り、同じレイアウトパターンで一部の画像が異なる場合には、双方の評価スコアは近いものとなる。したがって、評価スコアが近接した場合ユーザに画像を提示した場合、似通ったレイアウトが順次提示される可能性がある。本実施形態でのレイアウトリスト作成には、評価スコア範囲をまず乱数を用いて選択し、選択されたスコア範囲内のレイアウトを選ぶ。これにより、スコア値の近いレイアウトが続くこと可能性が低くなり、ユーザに似通ったレイアウトを連続して提示する可能性を低下させることができる。

**【 0 1 6 2 】**

また、評価スコア範囲を選択する乱数が図 2 6 及び図 2 7 ( A ) ~ ( B ) のような発生確率に分布を有しているので、レイアウトリストには、スコア上位のレイアウトが高確率で選ばれる。したがって、高品質のレイアウトを自動的にユーザに提供できる。

**【 0 1 6 3 】**

(他の実施形態)

以上説明した実施形態は、本発明の効果を得るための一例であり、類似の別手法を用いたり、異なるパラメータを用いたりしたとしても、同等の効果が得られる場合は、本発明の範疇に含まれる。

**【 0 1 6 4 】**

上述した実施形態では、レイアウト候補の上位スコアの画像をより多く選択するために、図 2 6 や図 2 7 に示すような発生確率分布をもった乱数を用いてレイアウトリストを作成する場合について説明したが、これに限定されるものではない。例えば、図 2 5 の S 3 3 0 3 において、乱数値 Q の代わりに、事前に準備したルックアップテーブルを用いてよい。この場合、ルックアップテーブルには、S 3 3 0 3 のステップが呼び出される回数の総提示レイアウト数 O と同数分、すなわち、O 個分、カウンタ値 p とレイアウト候補番

10

20

30

40

50

号1とを対応付けさせておく。このとき、例えば、レイアウト1に対する発生頻度が図26に示す発生確率であれば、実施形態1と同様の効果を得ることができる。

#### 【0165】

また、ユーザに対するレイアウト候補の提示順は、評価値の順と異なる順であればよい。ここで「異なる順」とは、提示順と評価値のスコア順が完全に一致していなければよく、一部の順が同じであってもよい。例えば、提示順は、スコア順が1～3番目のレイアウト候補、6番目のレイアウト候補、8番目のレイアウト候補・・・等のように、評価値の順と一致する部分があってもよい。

#### 【0166】

また、上述した実施形態では、アプリケーションを起動したタイミング毎にレイアウトリストを更新したが、更新のタイミングはこれに限定されるものではない。例えば、時間や日付が一定期間が経過した場合には、自動的にレイアウトリスト更新し、そのタイミングで乱数の初期値を変更するようにしてもよい。ある一定期間は、同じ状態である方がユーザにとっては作業上メリットがあるケースでは、このように所定のタイミングでレイアウトリストを更新するのが好ましい。例えば、あるレイアウトを選択、プリントした後、再度同一レイアウトをプリントするようなケースでは、再度プリントするまでは同じレイアウトリストである方が好ましい。この場合は、時間がたつと、同一レイアウトを出力する必要性が低くなると思われる所以、一定時間たつとレイアウトリストを更新する。これにより、異なるバリエーションのレイアウトを提供でき、ユーザの飽きさせず、同一画像群を楽しみつつ、かつ自動的にレイアウトを提供できる。また、他の例としては、アプリケーション上にユーザが入力できる何らかの指示手段があり、ユーザの指示があったタイミングでレイアウトリスト更新し、そのタイミングで乱数の初期値を変更してもよい。この場合、ユーザが飽きたかどうかの意図をより反映しやすい。さらに他の例としては、ユーザが提示されたレイアウト候補の中から、プリントを行った後に、自動的にレイアウトリスト更新ならびに乱数の初期化を行うようにしてもよい。この場合は、次にプリント動作を行う際に、提示順序が変わるため目新しさをユーザに提供することができる。さらに他の例としては、PCの起動時にレイアウトリストを更新するようにしてもよい。この場合は、アプリ起動前にレイアウトを作成するようなケースにおいて、バックグラウンド処理で行える。したがって、ユーザはアプリ起動時に瞬時に新たなレイアウトを見ることができる。

10

20

30

#### 【0167】

また、上述した実施形態では、テンプレートに画像を配置する領域数に対して選択できる画像数（例えば、入力される画像数）が充分に多い前提で説明をしたが、選択できる画像数が充分多くない場合には、レイアウトを複数作成しても、似た画像が選ばれてしまう。例えば、テンプレートにおいて、画像を配置する領域数が3で、選択できる画像数が4画像の場合、複数のレイアウトを生成しようとしても、必ず2画像は、同じ画像となってしまう。このように、選択できる画像数とテンプレート等に画像を配置する領域数によっては、前に提示するレイアウトと後に提示するレイアウトにおいて重複する画像を所定割合以上含めないようにすると、複数のレイアウトを表示することができない場合がある。以下、後に提示するレイアウトにおいて前に提示するレイアウトと重複する画像を許容する割合を許容割合ともいう。

40

#### 【0168】

従って、レイアウトを複数必ず表示させるためには、選択できる画像数、テンプレートに画像を配置する領域数に基づいて、動的に許容割合を変化させるのが好ましい。具体的には、例えば、入力される画像数が、レイアウトされる画像の配置数の近い程、レイアウト類似の評価値の判定基準を下げて、似ても良い閾値に設定する。また、入力される画像数が、レイアウトされる画像の配置数の遠い程、レイアウト類似評価値の判定基準を上げて、あまり似ない閾値を設定すればよい。また、例えば、許容割合を、選択できる画像数とレイアウトされる画像の配置数（例えば、テンプレートの画像を配置する領域数）に基づいて自動的に算出するようにしてもよい。例えば、必要なレイアウト数とテンプレート

50

の画像を配置する領域数を掛けた値に対して、複数の画像情報の画像数で割った値を許容割合としてもよい。この場合、所定以下の式となる。

所定割合 = 必要なレイアウト数 × (テンプレートの画像を配置する領域数 / 選択できる画像数) × 100 %

例えば、必要なレイアウト数が 10 で、テンプレートの画像を配置する領域数を 3、選択できる画像数が 100 画像とする。この場合、以下の式となり、許容割合が 30 となる。 $許容割合 = 10 \times (3 / 100) \times 100 = 30$  このように、選択できる画像数に応じて、所定の割合を算出すればよく、いずれの方法により算出するようにしてもよい。上述したように、選択できる画像数に応じて、許容割合を変えることで、生成できるレイアウトの有無を考慮しつつ、似たレイアウトが続きにくくバリエーションのあるレイアウトを提供することが可能となる。

#### 【 0 1 6 9 】

ここでは、許容割合を予め設定又は自動で設定する場合について説明したが、許容割合は上述した方法に限定されるものではない。例えば、許容割合をユーザが設定してもよい。この場合、必ず似たような画像を配置したくないなどのユーザの要求に応じた設定とすることができる。

#### 【 0 1 7 0 】

また、上述した実施形態では、候補レイアウトの評価スコア順とは異なる順序で候補レイアウトを並べたレイアウトリストを作成した。このようなレイアウトリストと、候補レイアウトの評価スコア順と同じ順に候補レイアウトを並べたレイアウトリストも同じプログラム内に保持するようにしてもよい。そして、例えば、ユーザの指定等の何らかの指定により、提示するレイアウトリストを切り換える構成としてもよい。この場合は、レイアウト評価値は共用可能なので、処理負荷は低減させることができる。このように、異なるレイアウトリストを保持させることにより、ユーザは自分が好ましい形態を選択できる。さらに、評価スコアの算出時にユーザ志向を学習して、より、レイアウトリストにユーザの意図を反映できる構成としてもよい。このような構成とした場合には、異なるレイアウトリストを保持させることは特に有効となる。

#### 【 0 1 7 1 】

また、上述した実施形態では、オブジェクトとして人物を例に挙げて説明したが、オブジェクトは人物とは限らない。犬や猫などのペットの認識処理を行ってこれらを認識することにより、オブジェクトとしてペットを設定することができる。また、エッジ検出などの形を認識処理によれば、建物や小物なども認識できるため、オブジェクトとして、建物や小物などを設定することができる。

#### 【 0 1 7 2 】

上述した実施形態では、レイアウト出力物として、1ページに複数の画像を配置した出力物を生成する例を挙げて説明したが、本発明は、複数ページのアルバム出力にも適用することができる。

#### 【 0 1 7 3 】

また、本発明は、複数の機器（例えばホストコンピュータ、インターフェイス機器、リーダ、プリンタ等）から構成されるシステムに適用しても、一つの機器からなる装置（例えば、プリンタ、複写機、ファクシミリ装置等）に適用しても良い。

#### 【 0 1 7 4 】

上述した実施形態は、以下の処理を実行することによっても実現される。すなわち、上述した実施形態の機能を実現するソフトウェア（プログラム）を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステム或いは装置のコンピュータ（CPUやMPU等）がプログラムを読み出して実行する処理である。また、プログラムは、1つのコンピュータで実行させても、複数のコンピュータを連動させて実行させるようにしてもよい。また、上記した処理の全てをソフトウェアで実現する必要はなく、一部又は全部をハードウェアによって実現するようにしてもよい。

#### 【 符号の説明 】

10

20

30

40

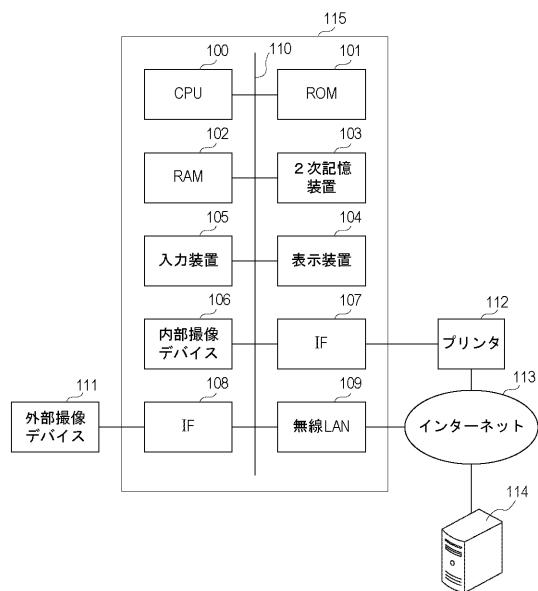
50

【 0 1 7 5 】

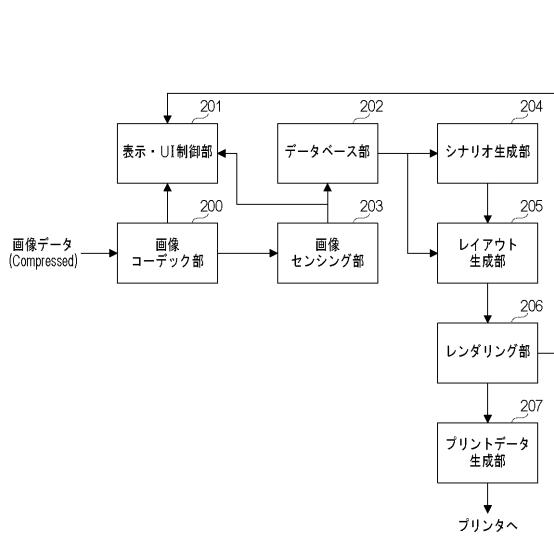
1 0 0 C P U  
 1 0 1 R O M  
 1 0 2 R A M 7 0 1 人物表示画面  
 7 0 2 人物表示画面に表示する人物グループの代表顔画像  
 7 0 3 人物グループ名  
 7 0 4 顔画像  
 8 0 1 サムネイル表示画面  
 8 0 2 画像のサムネイル  
 9 0 1 カレンダー形式表示画面  
 9 0 2 画像のサムネイル  
 1 3 0 1 画像のサムネイル表示画面  
 1 3 0 2 画像のサムネイル  
 1 3 0 3 マウスポインタ

10

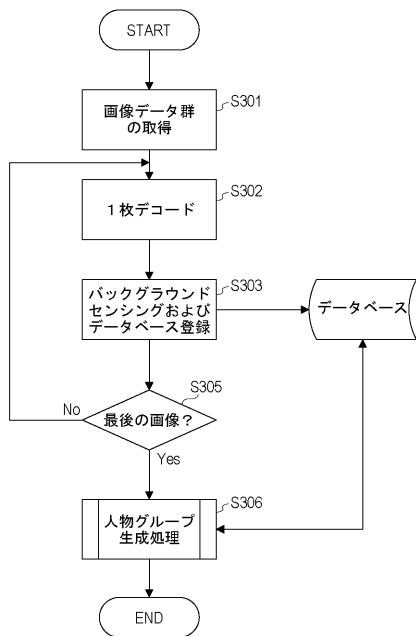
【図1】



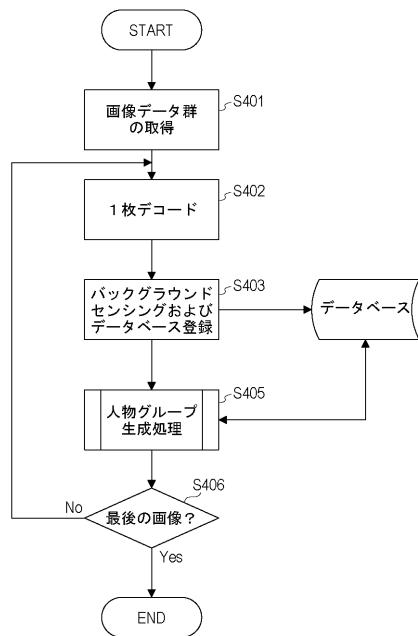
【図2】



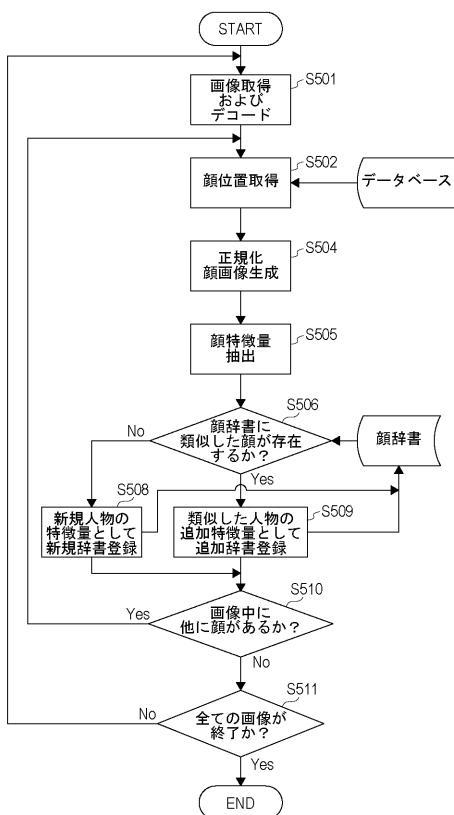
【図3】



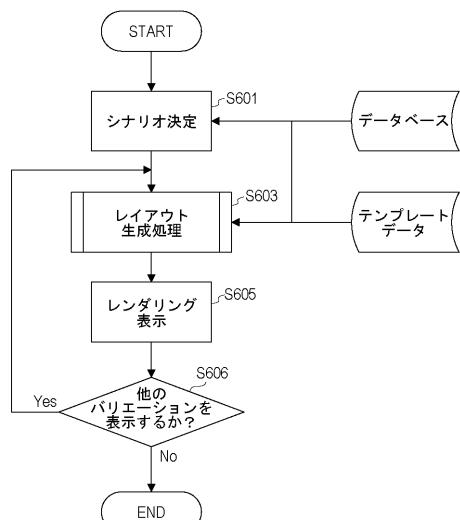
【図4】



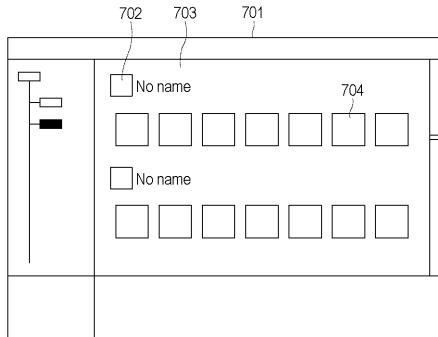
【図5】



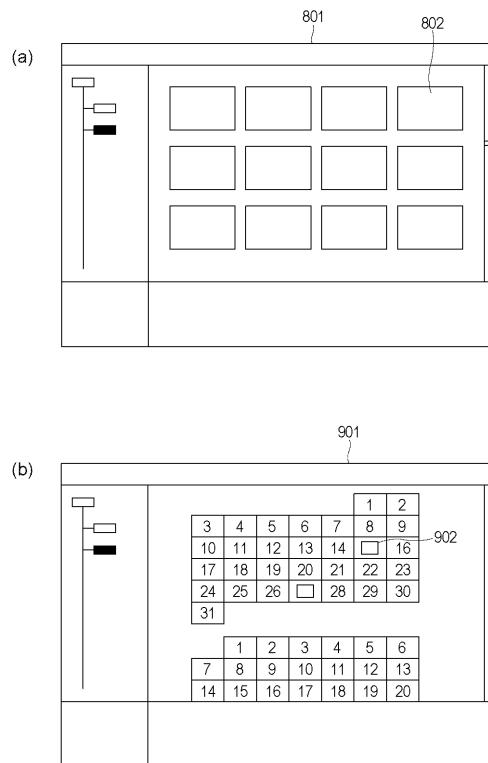
【図6】



【図7】



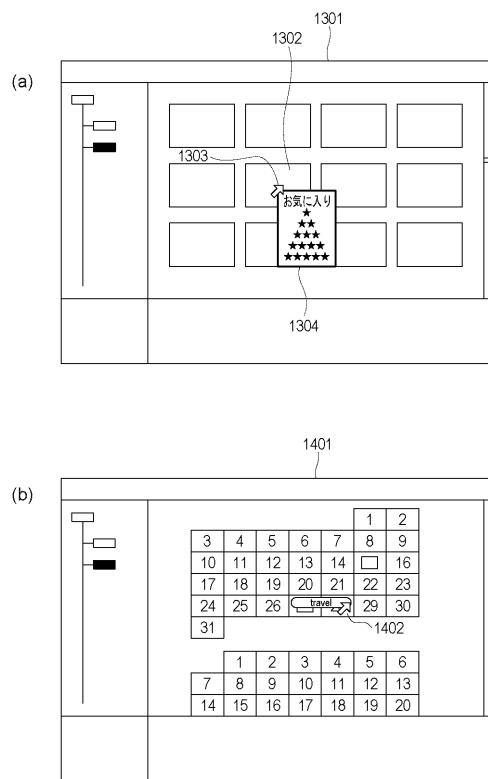
【図8】



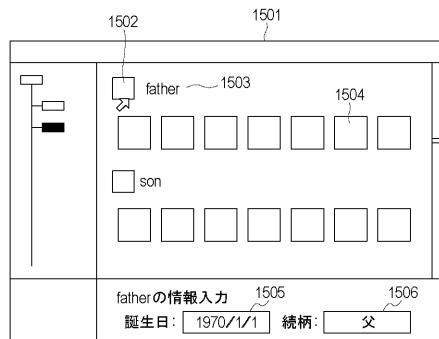
【図9】

```
<?xml version="1.0" encoding="utf-8"?>
<IMAGEINFO>
  <BaseInfo>
    <ID>0x00000001</ID>
    <ImagePath>C:\My Picture\IMG0001.jpg</ImagePath>
    <ImageSize width=3000 height=2000 />
    <CaptureDateTime>20100101:120000</CaptureDateTime>
  </BaseInfo>
  <SensInfo>
    <AveY>122</AveY>
    <AveS>38</AveS>
    <AveH>50</AveH>
    <SceneType>Landscape</SceneType>
    <Person>
      <ID>0</ID>
      <Position>
        <LeftTop x=420 y=200/>
        <LeftBottom x=420 y=300/>
        <RightTop x=520 y=200/>
        <RightBottom x=520 y=300/>
      </Position>
      <AveY>128</AveY>
      <AveCb>20</AveCb>
      <AveCr>20</AveCr>
    </Person>
    <...>
  </SensInfo>
  <UserInfo>
    <FavoriteRate>3</FavoriteRate>
    <ViewingTimes>5</ViewingTimes>
    <PrintingTimes>3</PrintingTimes>
    <Event>Travel</Event>
  </UserInfo>
  <...>
</IMAGEINFO>
```

【図10】



【図11】

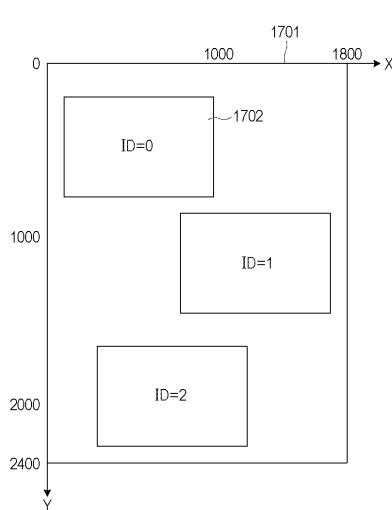


【図12】

```
<?xml version="1.0" encoding="utf-8" ?>
<PERSONINFO>
<ID></ID>
<Name>father</Name>
<Birthday>19700101</Birthday>
<Relationship>father</Relationship>
</PERSONINFO>
<PERSONINFO>
<ID>1</ID>
<Name>son</Name>
<Birthday>20000101</Birthday>
<Relationship>son</Relationship>
</PERSONINFO>
...

```

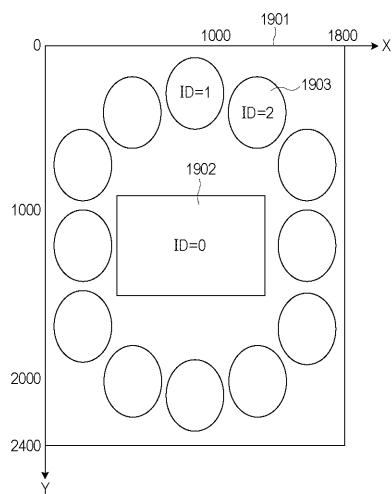
【図13】



【図14】

```
<?xml version="1.0" encoding="utf-8" ?>
<LayoutInfo>
<BASIC>
<Theme></Theme>
<PageSize>A4</PageSize>
<Resolution>300</Resolution>
</BASIC>
<ImageSlot>
<ID>0</ID>
<POSITION>
<LeftTop x=100, y=200>
<LeftBottom x=100, y=800>
<RightTop x=1000, y=200>
<RightBottom x=1000, y=800>
</POSITION>
<Shape>Rectangle</Shape>
<Person Group>Main Group</Person Group>
</ImageSlot>
<ImageSlot>
<ID>1</ID>
<POSITION>
<LeftTop x=800, y=700>
<LeftBottom x=800, y=1500>
<RightTop x=1700, y=700>
<RightBottom x=1700, y=1500>
</POSITION>
<Shape>Rectangle</Shape>
<Person Group>Main Group</Person Group>
</ImageSlot>
<ImageSlot>
<ID>2</ID>
<POSITION>
<LeftTop x=300, y=1700>
<LeftBottom x=300, y=2300>
<RightTop x=1200, y=1700>
<RightBottom x=1200, y=2300>
</POSITION>
<Shape>Rectangle</Shape>
<Person Group>Main Group</Person Group>
</ImageSlot>
</LayoutInfo>
```

【図15】



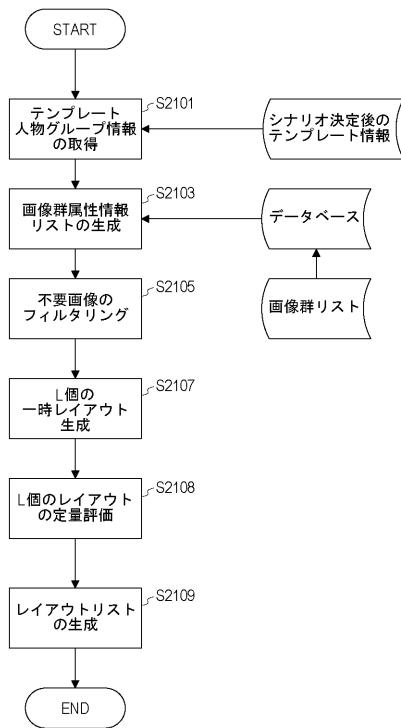
【図16】

```

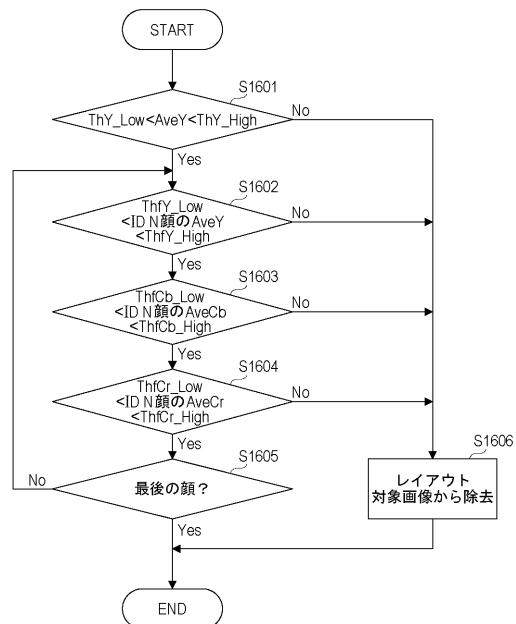
<?xml version="1.0" encoding="utf-8"?>
<LayoutInfo>
  <BASIC>
    <Theme></Theme>
    <PageSize>A4</PageSize>
    <Resolution>300</Resolution>
  </BASIC>
  <ImageSlot>
    <ID>0</ID>
    <POSITION>
      <LeftTop x=400, y=900>
      <LeftBottom x=400, y=1500>
      <RightTop x=1300, y=900>
      <RightBottom x=1300, y=1500>
    </POSITION>
    <Shape>Rectangle</Shape>
    <PersonGroup>SubGroup</Person Group>
  </ImageSlot>
  <ImageSlot>
    <ID>1</ID>
    <POSITION>
      <LeftTop x=700, y=100>
      <LeftBottom x=700, y=500>
      <RightTop x=1100, y=100>
      <RightBottom x=1100, y=500>
    </POSITION>
    <Shape>Ellipse</Shape>
    <PersonGroup>MainGroup</Person Group>
  </ImageSlot>
  <ImageSlot>
    <ID>2</ID>
    <POSITION>
      <LeftTop x=1100, y=200>
      <LeftBottom x=1100, y=600>
      <RightTop x=1450, y=200>
      <RightBottom x=1450, y=600>
    </POSITION>
    <Shape>Ellipse</Shape>
    <PersonGroup>MainGroup</Person Group>
  </ImageSlot>
  ...
</LayoutInfo>

```

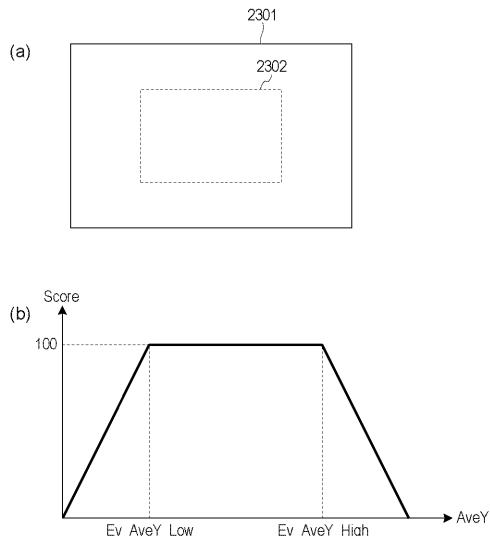
【図17】



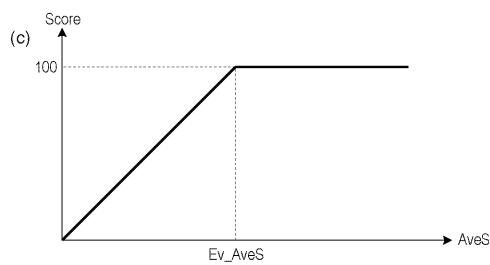
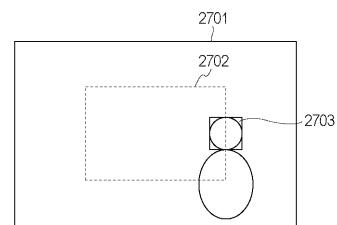
【図18】



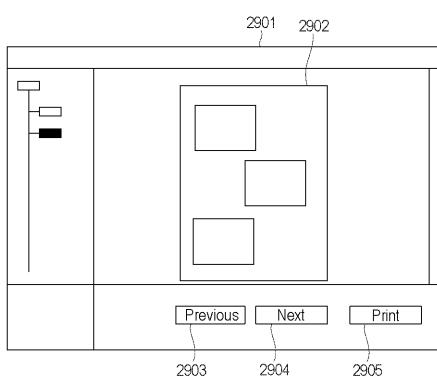
【図19】



【図20】



【図21】



【図22】

```
<?xml version="1.0" encoding="utf-8" ?>
<LayoutInfo>
  <BASIC>
    <Theme>growth</Theme>
    <PageSize>A4</PageSize>
    <Resolution>300</Resolution>
    <MainGroup>son, father</MainGroup>
    <SubGroup>son, father</SubGroup>
  </BASIC>
  <ImageSb>
    <ID>0</ID>
    <POSITION>
      <leftTop x=400, y=900/>
      <leftBottom x=400, y=1500/>
      <rightTop x=1300, y=900/>
      <rightBottom x=1300, y=1500/>
    </POSITION>
    <Shape>Rectangle</Shape>
    <PersonGroup>SubGroup</PersonGroup>
  </ImageSb>
  <ImageSb>
    <ID>1</ID>
    <POSITION>
      <leftTop x=700, y=100/>
      <leftBottom x=700, y=500/>
      <rightTop x=1100, y=100/>
      <rightBottom x=1100, y=500/>
    </POSITION>
    <Shape>ellipse</Shape>
    <PersonGroup>Main Group</PersonGroup>
  </ImageSb>
  <ImageSb>
    <ID>2</ID>
    <POSITION>
      <leftTop x=1100, y=200/>
      <leftBottom x=1100, y=600/>
      <rightTop x=1450, y=200/>
      <rightBottom x=1450, y=600/>
    </POSITION>
    <Shape>ellipse</Shape>
    <PersonGroup>Main Group</PersonGroup>
  </ImageSb>
  ...
</LayoutInfo>
```

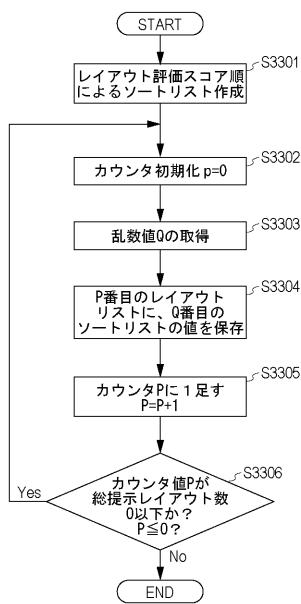
【図23】

```
<?xml version="1.0" encoding="utf-8" ?>
<LayoutInfo>
    <BASIC>
        <Theme>travel</Theme>
        <PageSize>A4</PageSize>
        <Resolution>300</Resolution>
        <MainGroup>son, mother, father</MainGroup>
    </BASIC>
    <ImageSlot>
        <ID>0</ID>
        <POSITION>
            <LeftTop x=100, y=200>
            <LeftBottom x=100, y=800>
            <RightTop x=1000, y=200>
            <RightBottom x=1000, y=800>
        </POSITION>
        <Shape>Rectangle</Shape>
        <PersonGroup>Main Group</PersonGroup>
    </ImageSlot>
    <ImageSlot>
        <ID>1</ID>
        <POSITION>
            <LeftTop x=800, y=700>
            <LeftBottom x=800, y=1500>
            <RightTop x=1700, y=700>
            <RightBottom x=1700, y=1500>
        </POSITION>
        <Shape>Rectangle</Shape>
        <PersonGroup>Main Group</PersonGroup>
    </ImageSlot>
    <ImageSlot>
        <ID>2</ID>
        <POSITION>
            <LeftTop x=300, y=1700>
            <LeftBottom x=300, y=2300>
            <RightTop x=1200, y=1700>
            <RightBottom x=1200, y=2300>
        </POSITION>
        <Shape>Rectangle</Shape>
        <PersonGroup>Main Group</PersonGroup>
    </ImageSlot>
</LayoutInfo>
```

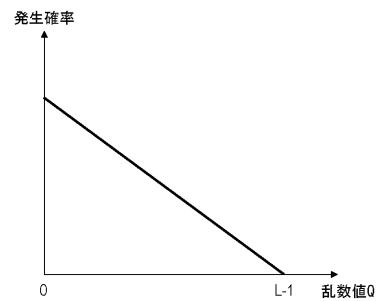
【図24】

```
<?xml version="1.0" encoding="utf-8"?>
<LayoutInfo>
    <BASIC>
        <Theme>growth</Theme>
        <PageSize>A4</PageSize>
        <Resolution>300</Resolution>
        <MainGroup>son</MainGroup>
        <SubGroup>son, father</SubGroup>
    </BASIC>
    <ImageSlot>
        <ID>0</ID>
        <POSITION>
            <LeftTop x=400 y=900>
            <LeftBottom x=400 y=1500>
            <RightTop x=1300 y=900>
            <RightBottom x=1300 y=1500>
        </POSITION>
        <Shape>Rectangle</Shape>
        <PersonGroup>Sub Group</Person Group>
        <ImageID>0x00000001</ImageID>
        <TrimmingRatio>50.0</TrimmingRatio>
    </ImageSlot>
    <ImageSlot>
        <ID>1</ID>
        <POSITION>
            <LeftTop x=700 y=100>
            <LeftBottom x=700 y=500>
            <RightTop x=1100 y=100>
            <RightBottom x=1100 y=500>
        </POSITION>
        <Shape>Ellipse</Shape>
        <PersonGroup>Main Group</Person Group>
        <ImageID>0x00000089</ImageID>
        <TrimmingRatio>38.0</TrimmingRatio>
    </ImageSlot>
    <ImageSlot>
        <ID>2</ID>
        <POSITION>
            <LeftTop x=1100 y=200>
            <LeftBottom x=1100 y=600>
            <RightTop x=1450 y=200>
            <RightBottom x=1450 y=600>
        </POSITION>
        <Shape>Ellipse</Shape>
        <PersonGroup>Main Group</Person Group>
        <ImageID>0x00000129</ImageID>
        <TrimmingRatio>53.0</TrimmingRatio>
    </ImageSlot>
    * * * *
</LayoutInfo>
```

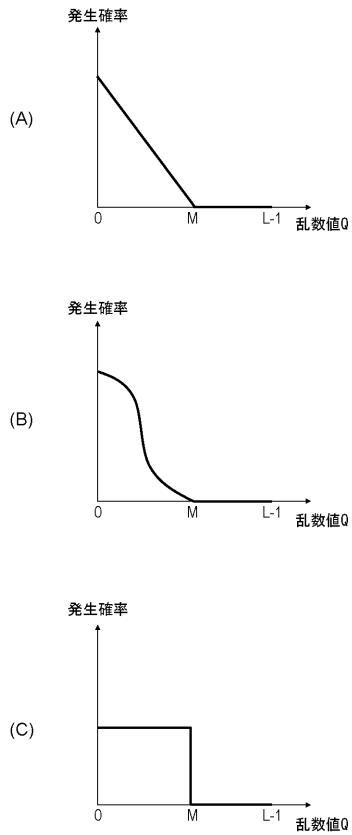
【図25】



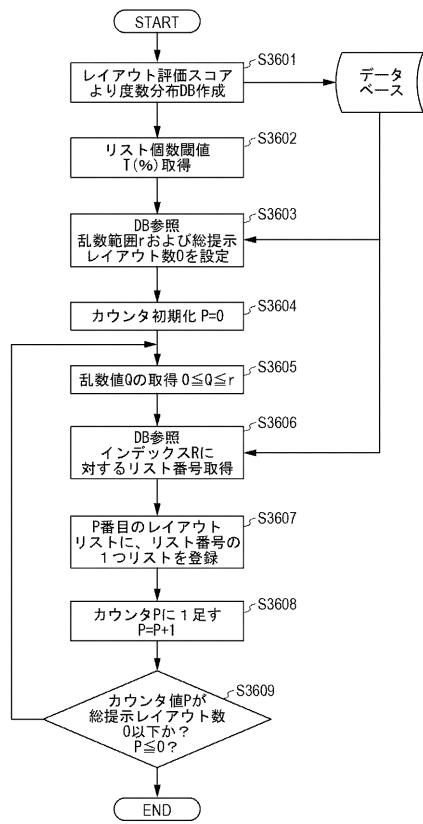
【図26】



【図27】



【図28】



---

フロントページの続き

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

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

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

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

審査官 鈴木 明

(56)参考文献 特開2013-046188(JP,A)  
特開2009-017328(JP,A)  
特開2002-063066(JP,A)

(58)調査した分野(Int.Cl., DB名)

H 04 N	1 / 387
G 06 T	11 / 60
H 04 N	5 / 232
G 06 T	3 / 00
G 06 T	1 / 00