

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第5209051号
(P5209051)

(45) 発行日 平成25年6月12日(2013.6.12)

(24) 登録日 平成25年3月1日(2013.3.1)

(51) Int. Cl. F I
G06T 11/20 (2006.01) G06T 11/20 100
G06F 17/21 (2006.01) G06F 17/21 570L

請求項の数 11 (全 13 頁)

(21) 出願番号	特願2010-514192 (P2010-514192)	(73) 特許権者	598036300
(86) (22) 出願日	平成20年6月27日 (2008.6.27)		テレフオンアクチーボラゲット エル エム エリクソン (パブル)
(65) 公表番号	特表2010-532513 (P2010-532513A)		スウェーデン国 ストックホルム エスー
(43) 公表日	平成22年10月7日 (2010.10.7)		164 83
(86) 国際出願番号	PCT/IB2008/003021	(74) 代理人	100076428
(87) 国際公開番号	W02009/001226		弁理士 大塚 康德
(87) 国際公開日	平成20年12月31日 (2008.12.31)	(74) 代理人	100112508
審査請求日	平成23年5月27日 (2011.5.27)		弁理士 高柳 司郎
(31) 優先権主張番号	60/929,465	(74) 代理人	100115071
(32) 優先日	平成19年6月28日 (2007.6.28)		弁理士 大塚 康弘
(33) 優先権主張国	米国 (US)	(74) 代理人	100116894
			弁理士 木村 秀二
		(74) 代理人	100130409
			弁理士 下山 治

最終頁に続く

(54) 【発明の名称】 データシステム及び方法

(57) 【特許請求の範囲】

【請求項 1】

ベクトルベースのコンピュータ可読言語で記述されたドキュメント内の複数の画像オブジェクトを、ビットマップベースのコンピュータ可読言語で記述されたドキュメント内の対応する画像オブジェクトに変換する方法であって、

検出手段が、前記ベクトルベースのドキュメント内の複数のオブジェクトに関連する複数のオブジェクト識別子を検出するステップと、

第1の生成手段が、検出された複数のオブジェクト識別子それぞれに対応する複数のオブジェクトのビットマップを生成する第1の生成ステップと、

ここで、前記ドキュメントの平面に垂直な軸に沿った各オブジェクトの位置に関する情報が維持され、

当該情報は前記軸に沿ったオブジェクトの位置を決定するベクトルベースドキュメント内のオブジェクトの順序を含む；

第2の生成手段が、前記ベクトルベースのドキュメントにおける前記オブジェクトの順番を保持するため、前記維持された情報に従って前記ビットマップを記述することで、前記ビットマップベースのコンピュータ可読言語で記述されたドキュメントを生成する第2の生成ステップと、

を有することを特徴とする方法。

【請求項 2】

前記第1の生成ステップは、前記オブジェクトのバウンディングボックスを決定するス

10

20

テップと、前記バウンディングボックスによって決定される前記ドキュメントの領域のラスタ化を行うステップとを有することを特徴とする請求項 1 に記載の方法。

【請求項 3】

選択手段が、1 以上の層化されたビットマップの透過属性を選択する選択ステップをさらに有することを特徴とする請求項 1 又は 2 に記載の方法。

【請求項 4】

前記ビットマップは、絶対位置決定方式である absolute positioning によって前記ビットマップベースの言語で記述された前記ドキュメント内に配置されることを特徴とする請求項 1 に記載の方法。

【請求項 5】

設定手段が、前記軸に沿った前記ビットマップの位置をセットする設定ステップをさらに有することを特徴とする請求項 1 に記載の方法。

【請求項 6】

電子プロセッサ (300) であって、

ベクトルベースのプロセッサ可読言語で記述されたドキュメントにおいて 1 以上の画像オブジェクトを、ビットマップベースのプロセッサ可読言語で記述されたドキュメント内の対応する画像オブジェクトに変換する旨の命令を実行するように構成された処理ユニット (302) と、

前記処理ユニットによって実行される命令を格納するように構成されたアプリケーションメモリ (306) とを有し、

前記処理ユニットは、以下の処理を実行する：

前記ベクトルベースのドキュメント内の 複数のオブジェクトに関連する複数のオブジェクト識別子 を検出するステップと、

検出された複数のオブジェクト識別子それぞれに対応する複数のオブジェクトのビットマップを生成する第 1 の生成ステップと、

ここで、前記ドキュメントの平面に垂直な軸に沿った各オブジェクトの位置に関する情報が維持され、

当該情報は前記軸に沿ったオブジェクトの位置を決定するベクトルベースドキュメント内のオブジェクトの順序を含む；

前記ベクトルベースのドキュメントにおける前記オブジェクトの順番を保持するため、前記維持された情報に従って前記ビットマップを記述することで、前記ビットマップベースのコンピュータ可読言語で記述されたドキュメントを生成する第 2 の生成ステップと

ことを特徴とする電子プロセッサ。

【請求項 7】

前記処理ユニットは、前記オブジェクトのバウンディングボックスを少なくとも決定するように構成され、かつ、前記バウンディングボックスによって決定された前記ドキュメントの領域のラスタ化を行うことを特徴とする請求項 6 に記載の電子プロセッサ。

【請求項 8】

前記処理ユニットは、前記層化されたビットマップのうちの少なくとも 1 つのビットマップの透過属性を選択するようにさらに構成されることを特徴とする請求項 6 に記載の電子プロセッサ。

【請求項 9】

前記処理ユニットは、絶対位置決定方式である absolute positioning によって、前記ビットマップベースの言語で記述された前記ドキュメント内にビットマップを配置するように構成されることを特徴とする請求項 6 に記載の電子プロセッサ。

【請求項 10】

前記処理ユニットは、前記ビットマップの所定位置を前記軸線に沿ってセットするように構成されることを特徴とする請求項 6 に記載の電子プロセッサ。

【請求項 11】

10

20

30

40

50

コンピュータが読み込み実行することで、当該コンピュータを、ベクトルベースのコンピュータ可読言語で記述されたドキュメント内の複数の画像オブジェクトをビットマップベースのコンピュータ可読言語で記述されたドキュメント内の対応する複数の画像オブジェクトに変換する、請求項1乃至5のいずれか1項に記載の方法の各ステップを実行させるための命令を格納したコンピュータ可読媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、コンピュータ及びソフトウェアを用いる画像の描画(rendering)に関し、特に、ベクトルグラフィックス形式のドキュメントをビットマップドキュメントに変換するための装置及びソフトウェアコンポーネントに関する。

10

【背景技術】

【0002】

スケーラブルベクトルグラフィック(SVG)は、拡張マークアップ言語(XML)に基づくコンピュータ可読言語であり、静的及び動的ベクトルグラフィック画像を表現するために用いられる。SVGは、画像属性(サイズ、形状、色など)を決定するXMLに準拠するタグを用いて2次元画像を定義することができる。SVGは2次元ベクトルグラフィック画像の形状だけでなく、ビットマップグラフィック及びテキストも表示することができる。SVGはベクトルベースのものであり、このことはコンテンツが特定の画面解像度及び画面サイズ用として記述されるのではなく、コンテンツを容易に拡大縮小できることを意味する。SVGは、W3C推奨としてSVGバージョン1.1を採用したワールドワイドウェブコンソーシアム(W3C)によって標準化されている。

20

【0003】

ハイパーテキストマークアップ言語(HTML)は、タグを用いて記述される別のコンピュータ可読言語であり、少なくとも、かかる点でXMLに類似している。XHTMLはXMLに準拠するHTMLのバージョンである。当初から、HTMLは、テキストの論理構造を表すように意図されたものであって、ドキュメント及びウェブページのようなコンテンツ内のグラフィックを表すように意図されたものではなかった。画像はHTMLではビットマップとして表される。見出し、パラグラフなどの物理的表現はHTMLコードを提供するブラウザによって決定される。時間の経過につれて、ウェブページの視覚的外観に対するさらなる制御がHTMLに関して望まれるようになった。ウェブ開発者及び他のコンテンツ開発者がテキストの書体、サイズ、重み付け、色をHTMLページで指定できるようにするフォントタグのようなタグが導入されるようになった。ゼロ幅の境界を持つテーブル要素を用いることによって、ページコンテンツの高度のレイアウトと配置を行うことが可能とはなったが、HTMLコードは雑然となり、理解し難いものになった。今日、HTMLは、ドキュメントの構造を表すために再び記述されるようになってはいるが、プレゼンテーション関連フォーマットは、(例えばフォント、色、間隔などの)スタイルをウェブページとドキュメントに追加するためのメカニズムであるカスケディングスタイルシート(CSS)に委ねられている。

30

【0004】

近年、正規のソフトウェアアプリケーションの振舞いの模倣を試みるウェブページであるウェブアプリケーションが現れてきている。例えば、最新のブラウザは、別のコンピュータ可読言語であるJavaScriptをサポートしていると共に、HTMLドキュメントは、ウェブページ上にある画像用コンテンツ要素にアクセスするためのW3C-標準のオブジェクト指向アプリケーションプログラムインタフェース(API)であるドキュメントオブジェクトモデルを介してオブジェクトとしてアクセス可能である。JavaScriptは、ウェブアプリケーション(ウェブコンテンツ又はウェブページ)の中に挿入又は組み込まれ、かつ、ウェブページに表示されるプロセッサによって実行可能な多くのタイプの記述ロジックのうちの1つである。

40

【0005】

50

多くの電子通信システム用の仕様を普及させる第3世代パートナプロジェクト(3GPP)は、3GPPリリース5仕様のためのSVG1.1を採用し、そして、SVGは、今日画像を生成しかつ表示するためにほぼ1億台のモバイル電話ハンドセット及び他のユーザ機器(UE)によって用いられている。例えば、UEは現在、上記機器の中に含まれている一般に狭いスクリーン用の画像及び他のダウンロード済みのコンテンツを調整するブラウザアプリケーションを用いてワールドワイドウェブに定常的にアクセスしている。

【0006】

SVG Tinyの仕様1.2はSVGバージョン1.2用としてW3Cにより現在開発されているところである。SVG Tinyは特にモバイル通信装置用として設計されたものであるが、典型的には限られたメモリと処理リソースとを有している。SVG Tiny 1.2はリリース6仕様用として3GPPによっても採用されている。オーディオ及びビデオの完全な制御を含む種々の新たなマルチメディア機能に対するサポートがマイクロDOM(μ DOM)及びスクリプトと共に含まれている。 μ DOMとはサブセットのドキュメントオブジェクトモデル(DOM)である。SVG属性のなかには、多くの種類のアニメーションを可能にする μ DOMスクリプトによって操作できるものもある。オブジェクトを追加したり、取り除いたりして、動的コンテンツをページ上に生み出すようにすることが可能である。

【0007】

特に、SVG及び他のベクトルベースの表現によってアニメーションが容易になる。なぜなら、前景オブジェクトによって視野から現在隠されている後景オブジェクトに関する情報が失われていないからである。したがって、異なる画像内の異なる位置に前景オブジェクトを出現させるとき、その異なる位置の画像の中に以前に隠された(単複の)後景オブジェクトを含むことが可能となる。特に、SVGでは、描画(rendering)の「ペインタモデル」が用いられ、このモデルでは「塗り潰し(paint)」が連続処理で適用される。換言すれば、一連の「塗り潰し」処理を行うことによって電子プロセッサが出力装置やディスプレイ上にオブジェクトを描画することになる。個々の処理によってプロセッサは出力ディスプレイの対応する領域にわたって塗り潰しを行うと共に、塗り潰された領域が以前に塗り潰された領域に重なり合うと、新たな塗り潰しは部分的に又は完全に古い塗り潰しを覆い隠すことになる。

【0008】

HTMLの中に定義されている画像はビットマップであるため、前景オブジェクトによって隠された後景オブジェクトの外観に関する情報は存在しない。この情報不足のために、前景オブジェクトを異なる位置に出現させるとき、プロセッサは以前の後景オブジェクトを描画することが困難となる。

【0009】

SVG又は等価のベクトルベースの画像をビットマップ画像に変換することは簡単ではあるが、そうすることにはいくつかの欠点がある。例えば、動きを示すために画像を変更することは、ひとたび、例えば画像がSVGからビットマップ画像へ変換されてしまった場合、非常に困難となる(不可能でさえある)。上述したように、前景オブジェクトによって隠された後景オブジェクトの外観であって、SVG又は他のベクトルベースの表現の中に存在していた外観に関する情報は、HTML又は他の非ベクトルベースの表現の中には存在していない。さらに、SVG Tiny 1.2がまだ完全には展開されていないという理由のために、SVG内の高度のアプリケーションは多くのUEによって実行可能ではなく、さらに、すべてのタイプのUEにおいてSVGがサポートされているわけではない。

米国特許公開番号US 2003/0222883A1は、混合メディアストリームに及び、演算処理量を最小限にして、プレゼンテーションワークステーションにて混合メディアを描画する方法及びシステムの記載を開示している。木構造(tree structure)は、ノードを含んでいる。個々のノードはサブ木へのポイントとなりえ、且つ、それぞれは幾何学的なオブジェクト、幾何学的なオブジェクトに適用する変換、又は、時間変化に表現を指

10

20

30

40

50

示する。シーンは、そのシーン中の各要素のノードを含む。また、これらとは別に、例えばテキストや形状のような、これら要素の外観を記述するものでも良い。ルートノードは、シーンの各オブジェクトの記述をつなぐ。演算量の最適化は、木のノードの選択の工程と、各サブ木を如何にして描画するか演算工程を含む。可能であれば、情報の描画は、ユーザに見せることになるオブジェクトの部分についてのみ演算されることである。

D i l l o r i o , A . 等による “ベクトルグラフィックスにおけるXMLベースの言語間の動的変換(Dynamic Conversion between XML-Based languages for Vector Graphics)”, Internet Imaging VII, Proc. SPIE-IS&T Electronic Imaging, SPIE vol. 6061, pp. 60610N-1 thru 60610N-9, 18 January 2006 には、2つのベクトルフォーマットであるSVG及びVMLと、ラスターフォーマットであるGIFとの間の変換を、簡単に、しかも、自動的に、無理なく良好に行うツールが記載されている。このツールは、変換ルール、近似性及び経験則を適用させることで、異なる機能性及び表現性で言語間の良好な変換を行っている。

【0010】

SVG及び等価のベクトルベースの画像をビットマップ及び等価の非ベクトルベースの画像に変換できることが望ましい。なぜなら、このことはSVGに源を発するコンテンツの表示及び他のベクトルグラフィックコンテンツをさらに多くのタイプのUE及び他の表示装置上に表示することを可能にするからである。

【発明の概要】

【発明が解決しようとする課題】

【0011】

SVG画像(アプリケーション又はドキュメント)は、ビットマップ画像に変換された複数の主たるコンポーネントに分割され、かつ、対応するHTMLページに配置され得る。このようにして、SVGドキュメントのダイナミクスが保持が可能となる。DOMなどのSVGと共通の機能が同じ方法で行われるように何らかの方法でHTMLページを構成することが可能である。例えば、SVGドキュメント内のコンポーネントから得られるID属性は、対応するHTMLドキュメント内の同じ変換済みのコンポーネントであってもよい。本発明はSVGとHTML以外のフォーマットに対しても適用可能であると共に、例えば、SVGからHTMLへ、並びに、HTMLからSVGへの両方向に実施することが可能である。

【課題を解決するための手段】

【0012】

本発明の諸側面によれば、ベクトルベースのコンピュータ可読言語で記述されたドキュメント内の1以上の画像オブジェクトを、ビットマップベースのコンピュータ可読言語で記述されたドキュメント内の対応する画像オブジェクトに変換する方法が提供される。本方法は、ベクトルベースのドキュメント内の1以上のそれぞれのオブジェクトに関連づけられた1以上のオブジェクト識別子を検出するステップと、検出された1以上のオブジェクト識別子に対応する1以上のオブジェクトのビットマップを生成するステップとを含む。生成されたビットマップは、ドキュメントの平面に垂直な軸線に沿ったビットマップの位置に関する情報をベクトルベースのドキュメントから保持する。

【0013】

本発明の他の側面によれば、プロセッサが、ベクトルベースのプロセッサ可読言語で記述されたドキュメント内の1以上の画像オブジェクトを、ビットマップベースのプロセッサ可読言語で記述されたドキュメント内の対応する画像オブジェクトに変換する旨の命令を実行するように構成された処理ユニットを含むプログラム可能な電子プロセッサと、処理ユニットによって実行される命令を格納するように構成されたアプリケーションメモリとが提供される。処理ユニットは、次のアクション、すなわち、ベクトルベースのドキュメント内の1以上のそれぞれのオブジェクトに関連づけられた1以上のオブジェクト識別子を検出するステップと、検出された1以上のオブジェクト識別子に対応する1以上のオブジェクトのビットマップを生成するステップとを実行するように構成される。生成され

10

20

30

40

50

たビットマップは、ドキュメントの平面に垂直な軸線に沿ったビットマップの位置に関する情報をベクトルベースのドキュメントから保持する。

【 0 0 1 4 】

本発明のさらなる諸側面によれば、ベクトルベースのコンピュータ可読言語で記述されたドキュメント内の1以上の画像オブジェクトを、ビットマップベースのコンピュータ可読言語で記述されたドキュメント内の対応する画像オブジェクトに変換する方法を、実行時にコンピュータに実行させる格納済みの命令を有するコンピュータ可読媒体が提供される。

【 0 0 1 5 】

本発明の様々な特徴、目的及び効果は、本発明を実施するための形態を図面と共に読むと明らかとなるであろう。

【図面の簡単な説明】

【 0 0 1 6 】

【図1A】ベクトルベースのフォーマットのドキュメント及び画像をドキュメント及びビットマップフォーマットの画像に変換する方法を示すフローチャートである。

【図1B】HTMLのような非ベクトルベースのフォーマットからSVGのようなベクトルベースのフォーマットへ変換を行う方法を示すフローチャートである。

【図2】変換可能な画像の描画を描く図である。

【図3】ベクトルベースのフォーマットのドキュメント及び画像をビットマップフォーマットのドキュメント及び画像に変換する電子プロセッサのブロック図である。

【図4】例示のモバイル通信装置を描く図である。

【発明を実施するための形態】

【 0 0 1 7 】

本願発明者らは、アドビフラッシュ (Adobe Flash) のようなSVG又は他のベクトルベースの画像が、オリジナルのSVGコードで表されるオブジェクトを検出するために、描画済みSVG画像の構文解析を行うことによって (HTMLドキュメント用の) 等価のビットマップ又は他の非ベクトルベースの画像に変換され得ることを認識した。個々の検出済みオブジェクトに対して、オブジェクトの別個のビットマップが作成され、次いで、画像がZ軸に沿ってどこに (すなわち、どの塗り潰し層に) 存在するかに関する情報が保持される。個々のオブジェクトに対して別個のビットマップ画像が作成され、次いで、別個のビットマップ画像が上下に層化される。すなわちこれら層化されたビットマップ画像はそのZ軸座標に基づいてZ軸に沿って配列される。透過的な又は不可視の画像としてビットマップ画像からなる様々な画像を選択的に描画することによって、画像のアニメーションを実現することが可能となる。

【 0 0 1 8 】

当業者であれば透過性と可視性との区別を理解するであろう。ビットマップ画像内の透過性によって、正方形や長方形の形状を持たないオブジェクトの表現が可能になる。例えば、空 (すなわち別のビットマップ画像によって表される背景) が光線の間から「見える」「光線」を伴った円として太陽を描くことができる。透過性を用いて、複数のビットマップ画像を上下に積重ねることにより「完全な」画像の構成が可能となる。可視性とは、ビットマップが目に見える形で又は隠された状態で描画されることを意味する用語である。このような可視属性を用いて、例えばオブジェクトがスクリーン上に現れるとき、オブジェクトの属性を変更することが可能となる。例えば、青い円を隠し、それによってディスプレイ上の同じ位置に赤い円が見えるようにすることが可能となる。

【 0 0 1 9 】

オブジェクトを識別できるようにSVGドキュメントにはラベルが付けられる。例えば、画像又はドキュメント内の個々のオブジェクトは、オブジェクト名のような対応するオブジェクト識別子を有し、そして識別子はベクトル形式から作成された抽象化層に存在していると考えられることができる。DOMツリー内のオブジェクトの所在位置を突止めるためには一般に識別子が用いられる。したがって、スクリプト可能に (すなわち記述ロジック

10

20

30

40

50

によって実行可能に)作成される任意のSVGドキュメントは通常(単複の)好適な識別子を有することになる。SVGドキュメントは通常インクスケイプ(Inkscape)のような好適な描画ツールによって作成される。インクスケイプとは、SVGファイル形式を用いるベクトルグラフィックエディタである。インクスケイプに関する情報はインターネット上でwww.inkscape.orgから入手可能である。

【0020】

一般に、オブジェクトとは、HTMLで表されるオブジェクトとして表現されるべき、SVGドキュメント又は他のベクトル・グラフィックドキュメントの明確な部分である。オブジェクトは一体にグループ化された単一の画像要素又は複数の要素であってもよく、このグループはある特定のラベルを有している。したがって、1つの可能性として、SVG要素がオブジェクトであることが考えられる。但し、オブジェクトのグループ化も可能であり、このグループ化は好適な描画ツールにより行うことができる。このようなグループ化は、マイクロソフト社の「パワーポイント(PowerPoint)」及び「ワード(Word)」のようなアプリケーションにおける非SVG描画オブジェクトのための「グループ」機能を示唆するものである。オブジェクト名は、オブジェクトを表すコンピュータ可読コードで識別子(ID)属性により好便に表される。上述したように、HTMLだけでなくSVG及び同等の言語によってもID属性が提供される。

10

【0021】

したがって、ドキュメント又はベクトルベースのコードで記述された画像をビットマップコードで記述されたドキュメント又は画像に変換する方法は、フローチャートによって図1Aに示される次のステップを含んでよい。ステップ102において、SVG又は他のベクトル・グラフィックドキュメント内のオブジェクト(特に画像オブジェクト)に関連するオブジェクト識別子は、例えばオブジェクト識別子を表すドキュメントSVGコードを走査することによって検出される。このようなオブジェクト検出を実行するために、コンピュータを適切にプログラムしてもよいし、あるいは、必要な画像以外の多くのビットマップ画像が作成されないように、スクリプト開発者がドキュメントをレビューして、「関心対象の」識別子の位置をピンポイントで正確に指定できるようにしてもよい。このオブジェクト検出ステップは、個々の画像要素をグループ化するステップと、(要素の動き又はアニメーションなどの)グループ化された要素の振舞いが様々なグループ化に対応して変化するかどうかを判定するステップとを含んでよい。SVGドキュメントのすべての主要部分のビットマップ画像を作成することは一般に可能であるが、そうすることは必ずしも実際的であるとは限らない。

20

30

【0022】

オブジェクト識別子の検出時に、ビットマップのような対応するオブジェクトの非ベクトル・グラフィックバージョンが生成される。このバージョンのオブジェクトは、対応する画像がz平面のどこに存在しているかに関するベクトルグラフィックコード情報から保持される(ステップ104)。オブジェクトのバウンディングボックスを決定し、次いで、当該特定領域をラスタライズしてビットマップに変えることにより好適なビットマップの生成が可能となる。バウンディングボックスとは、一般に、オブジェクト及びその派生オブジェクトを完全に囲むオブジェクトの座標系の軸と位置合わせされる一番ぴったりフィットする(tightest-fitting)矩形である。オブジェクトがラスタライズされると、他のすべてのオブジェクトは不可視にされ、それによってこれら他のオブジェクトは上記ビットマップの中に現れなくなる。当業者であれば理解できるように、すべてのオブジェクトの適切なプロセッサのメモリ内にあるSVGドキュメントを描画することによって、ドキュメント内のすべてのオブジェクトのバウンディングボックスが認知されると共に、バウンディングボックスの描画済みコンテンツの簡単な取得に關与するラスタ化がよく知られている。

40

【0023】

さらなる検出済みオブジェクトが存在すれば(ステップ106のイエス)、反復ステップ102及び104によって当該オブジェクトの非ベクトル・グラフィックバージョンが

50

生成される。検出済みオブジェクトがそれ以上存在しなければ（ステップ106のノー）、SVGコードで検出された1以上のオブジェクトに対応する1以上の別個のビットマップ画像が、原SVGドキュメントをHTMLに変換する際に該画像のそれぞれのz平面座標に基づいて上下に層化される（ステップ108）。

【0024】

これらのビットマップは、好適には絶対位置決定方式を用いて、SVGドキュメントの場合のようにHTMLドキュメント内の同じ場所に配置される。オブジェクトが現れる順序は2つのドキュメントに関して同じであるため、これらのオブジェクトがZ軸上に現れる順序、すなわち、ドキュメントのビューイング平面に対して垂直な方向は同じになる。上記とは別に、Z軸上のオブジェクトの位置が明示的に設定されるようにドキュメント用HTMLコードを記述することができる。

10

【0025】

オプションの追加ステップ110として、画像が透過的になるように、1以上のビットマップベースの画像の透過属性が選択される。この結果、変換済みの画像が非ベクトル・グラフィックフォーマットを有しているにもかかわらず、この画像を容易にアニメーション化することが可能となる。当業者であれば理解できるように、例えばプログラム可能な電子プロセッサのコンポーネントを用いるためにステップ102～106の順序を変更してもよい。

【0026】

SVGドキュメント及び他のベクトルベースのドキュメントは埋め込まれた画像をサポートする。そのため、HTMLのような非ベクトルベースのフォーマットから、SVGのようなベクトルベースのフォーマットへの変換が行われる逆方向の方法は簡単である。但し、これらのベクトル表現が非ベクトルベースの元のベクトル表現の中に含まれていないのでオブジェクトのベクトル表現を利用することはできない。

20

【0027】

図1Bは例示の逆方向の方法を示すフローチャートであり、この図では、ステップ122において、HTMLで表されるビットマップ画像に関連づけられた1以上のオブジェクト識別子が、例えばオブジェクト識別子用のHTMLコードを走査することによって検出される。ステップ124において、検出済みのビットマップ画像が該画像のそれぞれの座標でSVGドキュメントの中に埋め込まれる。別の検出オブジェクトが存在すれば（ステップ126のイエス）、当該オブジェクトのビットマップは反復ステップ122及び124によってSVGドキュメントの中に埋め込まれる。検出済みオブジェクトがそれ以上存在しなくなれば（ステップ126のノー）、ドキュメントの変換は完了し、処理フローは他のタスクを続行する（ステップ128）。

30

【0028】

図2は、正方形によって覆われる黒塗りの青色の円を含む画像オブジェクトの描画である。（特許出願書式の制限に起因して図2の円は図を明瞭にするために曖昧にされず、黒になっている。）図2に描かれている画像オブジェクトはSVGで以下のように記述することができる：

<svg>

<circle cx="100" cy="50" r="30" style="fill:blue"/>

</svg>

そして、このSVGコードでは、ex、cy、rの動画化は可能ではあるが引き継ぐことはできないSVG属性として認識される。すなわち、属性ex、cy（円の中心の座標）、r（円の半径）を記述することは可能であるが、これらの属性をHTMLドキュメントへ直接引き継ぐことは不可能である。円の半径や色のような他の属性を変えるためには、2つの（またはそれ以上の）円が作成され、1つの円は目に見える状態にされ、同時に1つの円（またはそれ以上の円）は隠された状態にされる。

40

【0029】

当業者であれば理解できるように、明確なZ軸座標はSVGでは存在せず、ドキュメン

50

ト内の要素の順序によって要素のZ軸位置が決定されることになる。z座標が指定されていない場合、上記はHTMLの場合にもあてはまることである。HTMLドキュメントにおいて、これらの要素を別の何らかの順序で配置することを選択した場合、SVGドキュメント内のz位置が追跡され、例えばこれらのz位置を好適なメモリに格納することによって、z座標をHTMLドキュメントに適宜セットすることが可能となる。

【0030】

図2に描かれている画像オブジェクトはHTMLでは以下のように記述することができる：

```
<html>
```

```
  <div style= "position: absolute ; left : 70px; top: 20px;">
```

```
    
```

```
  </div>
```

```
</html>
```

SVGのような言語で表される単一のDOMコマンドを用いて、図2のカバー用正方形を削除し、円を可視にすることが可能となる。例えば、上述のようなラスタによって、図2の画像が典型的なビットマップに変換された場合、円はもはや画像内の別個のオブジェクトとしては提示されなくなる。換言すれば、典型的ビットマップは単一層のみを有することになる。

【0031】

上述のように、SVG画像(ドキュメント)は複数のさらに小さな画像に変換される。この変換は個々のSVGオブジェクトを表す1つの画像を用いて実行可能であるか、SVGオブジェクトのグループ化を行うことが可能である。上述した変換及びその他のステップは1以上のスクリプトにより容易に実行可能である。これらのスクリプトは好適にはSVG及びHTMLに共通のものであることが望ましい。すなわち、位置、名称などに対する参照のみがスクリプトで用いられる

本発明を用いて、同じ(又は非常に類似した)外見及び機能の達成が同じクライアントの2以上のバージョンにおいて可能となる。本発明は、異なるバージョンが例えばDOM APIなどを有するという共通APIの利点を利用するものである。これによって共通スクリプトの利用が可能となる。スクリプトとHTML DOMとの組み合わせは、アニメーション、対話機能、及び表現効果のために利用できる「動的HTML」と呼ぶことができる。

【0032】

このような変換方法は多くのプログラム可能な電子プロセッサによって好適に実行することが可能である。図3は上述した方法を実行できる適切なプロセッサ300のブロック図である。プロセッサ300は処理ユニット302を含み、この処理ユニット302は1又は2以上のソフトウェアモジュール及びアプリケーションを実行し、本願において説明する方法を実行する。キーボード又は他の装置を介してプロセッサ300へのユーザ入力提供され、ユーザに対してディスプレイ304に情報を示すことができる。キーボード/ディスプレイ304はオプションと見なされ、したがって破線によって示されている。これは上述した方法のなかにはユーザの対話処理を必要としないで実行できるものもあるという理由による。上記方法を定義するソフトウェア命令は好適なアプリケーションメモリ306に保存できると共に、プロセッサは所望情報のダウンロードとキャッシュの少なくともいずれかを好適なメモリ308において実行することも可能である。プロセッサ300は、他のプロセッサ、通信装置などのような別のコンポーネントをプロセッサ300と接続するために用いることができるインタフェース310を含んでもよい。

【0033】

図4はプロセッサ300を含んでよいモバイル電話機のようなUE400を描く図である。UE400は、ジョイスティック又は他のナビゲーション装置410及びディスプレイ420を含んでもよい。ウェブブラウザアプリケーション430はユーザ機器400においてプロセッサ300により実行することができる(図4には図示せず)。ウェブアプ

10

20

30

40

50

リケーション 440 はダウンロードされることができると共に、ブラウザ 430 において実行されることができ。ウェブアプリケーションはユーザには不可視な 1 組のリンク 443 又はディスプレイ 420 上で閲覧可能にすることができるアプリケーションを含んでもよい。ウェブアプリケーション 440 は、本願に記載の方法を実行するためのスクリプト用ロジック 445 を含むこともできる。

【0034】

当業者であれば理解できるように、本願は、SVG 及び HTML に焦点を当てるものである。これら SVG と HTML の双方は、HTML よりもさらに自由度の高い SVG を有するマークアップ言語であると共に、このマークアップ言語は単に便宜を図るためのものであって、本発明を限定するためのものではない。マークアップ言語間においてドキュメントを変換するための従来の技術は典型的には、マークアップ(コード)を変換するステップを含むものである。このような従来の技術の一例として拡張可能なスタイルシート言語変換(XSLT)があり、これはXML文書を他のXMLや他のドキュメントに変換するために用いることができるW3CXMLベースの言語である。逆に、本願に基づく技術は、ディスプレイ上に表示されている(又は表示されるべき)特筆すべきスクリプト可能な/動画化可能な属性を保持する技術である。この技術は、SVG要素のグループ化を行い、それによって1ビットマップ画像又は(オブジェクトがスクリーン上に現れる際の)オブジェクトが、変換済み(HTML)ドキュメントのスクリプト化/動画化を行う所望の方法に応じて、積層された(グループ化された)(透過的な)ビットマップ層から構成される等価の画像に要素を変換するステップを含むことができる。

【0035】

例えば、時と共に変化するドキュメントあるいは別様に変動するドキュメントに対応するために、上述の手続が必要に応じて繰り返し実行される、ということは了解されるであろう。変換方法はオフラインで実行することができるが、これらの変換方法は、例えばウェブブラウザリンク上でのユーザのクリックにตอบสนองしてモバイル電話機や他のUEなどにおいてリアルタイムで実行することも可能である。理解を簡単にするために、プログラム可能なコンピュータシステムのコンポーネントが実行できる行為のシーケンスなどに関して本発明の多くの側面を説明した。専用回路(例えば専用機能を実行するように相互接続された離散論理ゲートや特定用途向け集積回路)、1又は2以上のプロセッサが実行するプログラム命令、又はこれらの組合せによって様々な行為が、実行可能であることは認識されるであろう。

【0036】

さらに、適当な命令セット(この命令セットは、コンピュータ・ベースのシステム、プロセッサ内蔵システム、または媒体から命令をフェッチしてその命令を実行できるその他のシステムなどの命令実行システム、装置又はデバイスによって、またはこれらと結合して使用される)を格納した任意の形式のコンピュータ可読記憶媒体内において、本発明は、完全に具体化されるものであるとさらに考えることができる。本明細書での使用では、「コンピュータ可読媒体」とは、命令実行システム、装置、またはデバイスによって、またはそれと結合して使用するための、プログラムを装備、格納、通信、伝搬または転送できる任意の手段であってよい。コンピュータ可読媒体は、例えば、電子的、磁氣的、光学的、電磁的、赤外線、半導体のシステム、機器、装置、伝播媒体であってもよいが、これらに限定されない。コンピュータ可読媒体のより詳細な例(網羅的なリストではないが)は、1つ以上の結線を含む電気的な接続、可搬型コンピュータ・ディスク、ランダムアクセスメモリ(RAM)、リードオンリメモリ(ROM)、消去可能でプログラム可能な読み出し専用メモリ(EPROMまたはフラッシュメモリ)、及び光ファイバを含む。

【0037】

以上に述べたように、本発明は、多くの異なる形式(必ずしもその全てについて上記では説明されていないが)で具体化されてもよく、全てのそのような形式は本発明の範囲に含まれると思量するものである。本発明の多様な形態の各々について、任意のそのような形式は、説明の動作を実行する「ように構成された論理」、または別案として、説明の

10

20

30

40

50

動作を実行する「ところの論理」と呼ばれてもよい。

【0038】

用語「含む」及び「備える」は、本出願での使用では、述べられた特徴、整数、工程、またはコンポーネントが存在することを指定するものであり、その他の1つ以上の特徴、整数、工程、コンポーネントまたはそれらのグループの存在または付加を排除するものではないことに重々留意されたい。

【0039】

以上で説明した個々の実施形態は、単に説明のためのものであって、いかなる手法であれ本発明を限定するものとは考えないようにされたい。本発明の範囲は、添付の特許の範囲によって決定されるものであり、特許の範囲に帰着する全ての派生形態及び等価案は本発明に包含されることを意図するものである。

【図1A】

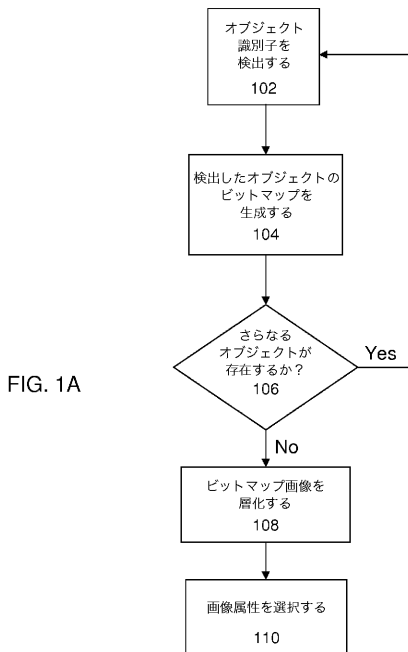


FIG. 1A

【図1B】

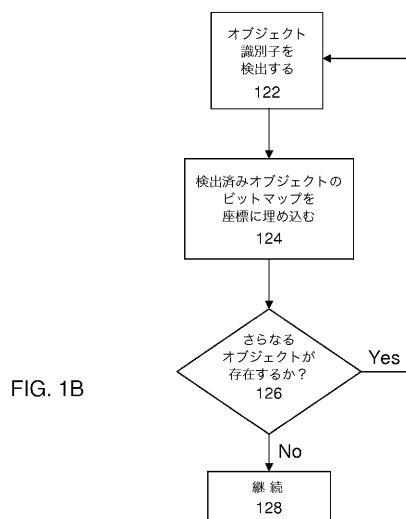


FIG. 1B

【図2】

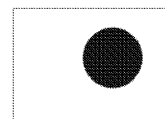
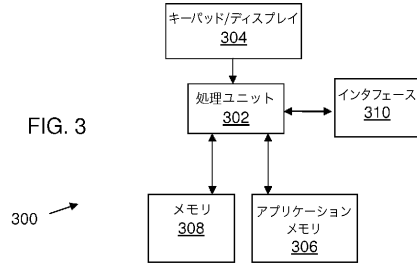
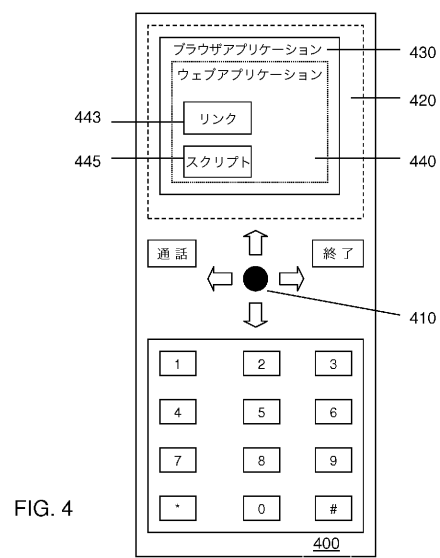


FIG. 2

【 図 3 】



【 図 4 】



フロントページの続き

- (72)発明者 ブリドル, クリントン
スウェーデン国 ウプランズ ヴェサビー 19471, リッタルガタン 278
- (72)発明者 オルド, ステファン
スウェーデン国 エンスケデ 12246, ラバンクスグレンド 4
- (72)発明者 プロディン, ペル-エリク
スウェーデン国 ソレンテユナ エス-191 44, スクユートバネヴェーゲン 47

審査官 千葉 久博

- (56)参考文献 特開2007-072601(JP,A)
特開2005-301422(JP,A)
特開2004-164208(JP,A)
特表2002-521774(JP,A)
米国特許出願公開第2003/0222883(US,A1)
国際公開第2006/040925(WO,A1)
青柳龍也, "XMLベースの新世代グラフィックス言語 SVGの実力を知る", Java WORLD, 日本, (株)IDGジャパン, 2001年 2月 1日, 第5巻, 第2号, p.131-142
DI IORIO A. ET AL, "Dynamic conversion between XML-based languages for vector graphics", INTERNET IMAGING VII., 2006年 1月18日, Vol.6061

(58)調査した分野(Int.Cl., DB名)

G06T 11/00 - 11/40
G06F 17/21
G06T 11/60, 11/80
G09G 5/00 - 5/42