

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2011-123127

(P2011-123127A)

(43) 公開日 平成23年6月23日 (2011.6.23)

| (51) Int.Cl. | F I | テーマコード (参考) |
|-----------------------------|----------------|-------------|
| G03B 21/00 (2006.01) | G03B 21/00 E | 2K103 |
| G09G 5/00 (2006.01) | G09G 5/00 510B | 5B069 |
| G09G 5/14 (2006.01) | G09G 5/14 C | 5C058 |
| G09G 5/36 (2006.01) | G09G 5/00 510H | 5C082 |
| G09G 5/391 (2006.01) | G09G 5/36 510M | 5C164 |

審査請求 未請求 請求項の数 23 O L (全 17 頁) 最終頁に続く

(21) 出願番号 特願2009-278832 (P2009-278832)
 (22) 出願日 平成21年12月8日 (2009.12.8)

(71) 出願人 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 100090273
 弁理士 園分 孝悦
 (72) 発明者 田邊 章弘
 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
 Fターム(参考) 2K103 AA05 AA16 AB10 BB05 CA53
 CA54 CA58 CA60 CA62 CA73
 5B069 AA01 BB09 BB17 CA03 CA14
 CA17
 5C058 AA18 BA01 BA21 BA35 EA02

最終頁に続く

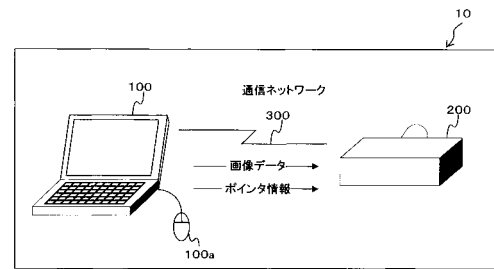
(54) 【発明の名称】 画像処理装置、画像表示装置及び画像伝送システム

(57) 【要約】

【課題】ユーザが見たい箇所の領域のフレームレートが低下してしまうのを防止できるようにする。

【解決手段】ネットワークを介して画像表示装置へ画像を伝送する画像伝送手段と、アプリケーションごとに生成される矩形領域を表示部に表示するポインタ入力手段と、前記表示部へ出力する出力画像を取得する画像取得手段と、前記表示部に出力される矩形領域を判別する矩形領域判別手段と、前記矩形領域判別手段により判別された矩形領域の中で、特定の矩形領域を判別する特定矩形領域判別手段と、前記特定矩形領域判別手段により判別された特定の矩形領域のフレームレートを変更して前記画像伝送手段により前記画像表示装置へ伝送するフレームレート制御手段とを設ける。

【選択図】 図1



【特許請求の範囲】**【請求項 1】**

ネットワークを介して画像表示装置へ画像を伝送する画像伝送手段と、
アプリケーションごとに生成される矩形領域を表示部に表示するポインタ入力手段と、
前記表示部へ出力する出力画像を取得する画像取得手段と、
前記表示部に出力される矩形領域を判別する矩形領域判別手段と、
前記矩形領域判別手段により判別された矩形領域の中で、特定の矩形領域を判別する特定矩形領域判別手段と、
前記特定矩形領域判別手段により判別された特定の矩形領域のフレームレートを変更して前記画像伝送手段により前記画像表示装置へ伝送するフレームレート制御手段とを有することを特徴とする画像処理装置。

10

【請求項 2】

前記矩形領域判別手段において判別された各矩形領域が動画を表示する矩形領域であるかを判別する動画領域判別手段を有し、
前記動画領域判別手段により判別された動画を表示する矩形領域を、前記特定矩形領域判別手段により特定の矩形領域に設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】

前記動画を表示する矩形領域において、動画が再生されたかどうかを判別する動画再生判別手段を有し、
前記動画再生判別手段により動画再生が開始されたことが判別された場合に、前記フレームレート制御手段は、前記動画を表示する矩形領域のフレームレートを上げることを特徴とする請求項 2 に記載の画像処理装置。

20

【請求項 4】

前記矩形領域判別手段において判別された各矩形領域の中で選択中の矩形領域を判別する判別手段を有し、
前記判別手段により判別された選択中の矩形領域を、前記特定矩形領域判別手段により特定の矩形領域に設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 5】

前記動画領域判別手段により動画を表示すると判別された矩形領域と、
前記判別手段により選択中の矩形領域と判別された矩形領域とが同一の矩形領域であった場合に、前記同一の矩形領域を特定矩形領域判別手段により特定の矩形領域に設定して前記特定の矩形領域のみ前記画像伝送手段で伝送することを特徴とする請求項 2 ~ 4 の何れか 1 項に記載の画像処理装置。

30

【請求項 6】

前記矩形領域判別手段において判別された各矩形領域の中で最も領域サイズの大きい矩形領域を判別する最大矩形領域判別手段を有し、
前記最大矩形領域判別手段により判別された最大矩形領域を、前記特定矩形領域判別手段により特定の矩形領域に設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 7】

前記矩形領域判別手段により判別された矩形領域のいずれかを移動する矩形領域移動手段を有し、
前記矩形領域移動手段により矩形領域が移動した際に、前記特定矩形領域判別手段により特定の矩形領域に設定された矩形領域の設定を解除し、前記画像取得手段により取得した画像を前記画像伝送手段により伝送することを特徴とする請求項 1 ~ 6 の何れか 1 項に記載の画像処理装置。

40

【請求項 8】

前記矩形領域判別手段により判別された矩形領域のいずれかの大きさを変更する矩形領域サイズ変更手段を有し、
前記矩形領域サイズ変更手段により矩形領域の大きさを変更した際に、前記特定矩形領

50

域判別手段により特定の矩形領域に設定された矩形領域の設定を解除し、前記画像取得手段により取得した画像を前記画像伝送手段により伝送することを特徴とする請求項 1 ~ 6 の何れか 1 項に記載の画像処理装置。

【請求項 9】

前記矩形領域サイズ変更手段により、特定の矩形領域に設定された矩形領域を拡大した際に、前記特定の矩形領域のフレームレートを下げることが特徴とする請求項 8 に記載の画像処理装置。

【請求項 10】

前記矩形領域サイズ変更手段により特定の矩形領域に設定された矩形領域を縮小した際に、前記特定の矩形領域のフレームレートを上げることが特徴とする請求項 8 に記載の画像処理装置。

10

【請求項 11】

ポインティングデバイスの入力を受け付けるポインタ入力手段と、

前記ポインタ入力手段により入力されるポインタの状態を検出するポインタ状態検出手段とを有し、

前記ポインタ状態検出手段により検出されたポインタ状態に応じて、前記特定矩形領域判別手段で設定された特定の矩形領域のフレームレートを変更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 12】

前記ポインタ状態検出手段によりポインタの位置を検知し、ポインタの位置が前記特定矩形領域判別手段により判別された特定の矩形領域に存在する場合に、前記フレームレート制御手段は、特定の矩形領域のフレームレートを上げることが特徴とする請求項 11 に記載の画像処理装置。

20

【請求項 13】

前記ポインタ状態検出手段によりポインタの移動を検知し、ポインタが移動していない場合には、

前記特定矩形領域判別手段により判別された特定の矩形領域のみ前記画像伝送手段により伝送することを特徴とする請求項 11 に記載の画像処理装置。

【請求項 14】

前記ポインタ状態検出手段によりポインタの移動を検知し、ポインタが移動している場合には、前記フレームレート制御手段は、前記特定矩形領域判別手段により判別された特定の矩形領域のフレームレートを下げることが特徴とする請求項 11 に記載の画像処理装置。

30

【請求項 15】

前記ポインタ状態検出手段によりポインタが矩形領域をドラッグしていることを検知し、ポインタが矩形領域をドラッグしている場合は、前記特定矩形領域判別手段により特定の矩形領域に設定された矩形領域の設定を解除し、前記画像取得手段により取得した画像を前記画像伝送手段により伝送することを特徴とする請求項 11 に記載の画像処理装置。

【請求項 16】

前記矩形領域判別手段により判別された矩形領域の大きさが所定の大きさ以上である場合に、前記特定矩形領域判別手段により特定の矩形領域として設定することを特徴とする請求項 1 ~ 15 の何れか 1 項に記載の画像処理装置。

40

【請求項 17】

前記画像伝送手段における帯域状態を検知する帯域状態検知手段を有し、

前記帯域状態検知手段により検知された帯域状態が所定以下である場合に、前記特定矩形領域判別手段により特定の矩形領域を設定することを特徴とする請求項 1 ~ 15 の何れか 1 項に記載の画像処理装置。

【請求項 18】

前記特定矩形領域判別手段により設定された特定の矩形領域が複数存在する場合において、複数存在する特定の矩形領域の優先順位を設定する優先順位設定手段を有し、

50

前記優先順位設定手段により設定された優先順位に従い各特定の矩形領域のフレームレートを設定することを特徴とする請求項 1 ~ 15 の何れか 1 項に記載の画像処理装置。

【請求項 19】

ネットワークを介して画像処理装置から送られる画像信号を受信する信号受信手段と、前記信号受信手段により受信した画像信号を、画像表示装置に表示可能な形式へ変換する変換手段と、

前記変換手段により変換された画像信号を前記画像表示装置に表示する画像表示制御手段とを有することを特徴とする画像表示装置。

【請求項 20】

前記画像表示装置は、入力画像を投射する液晶パネル、前記液晶パネルを駆動する駆動ドライバ、レンズ、駆動系及び光源で構成される投影部を有するプロジェクタであることを特徴とする請求項 19 に記載の画像表示装置。

10

【請求項 21】

請求項 1 ~ 18 の何れか 1 項に記載の画像処理装置と、請求項 19 または 20 に記載の画像表示装置とからなることを特徴とする画像伝送システム。

【請求項 22】

ネットワークを介して画像表示装置へ画像を伝送する画像伝送工程と、

アプリケーションごとに生成される矩形領域を表示部に表示するポインタ入力工程と、

前記表示部へ出力する出力画像を取得する画像取得工程と、

前記表示部に出力される矩形領域を判別する矩形領域判別工程と、

20

前記矩形領域判別工程により判別された矩形領域の中で、特定の矩形領域を判別する特定矩形領域判別工程と、

前記特定矩形領域判別工程により判別された特定の矩形領域のフレームレートを変更して前記画像伝送工程により前記画像表示装置へ伝送するフレームレート制御工程とを有することを特徴とする画像処理方法。

【請求項 23】

ネットワークを介して画像処理装置から送られる画像信号を受信する信号受信工程と、

前記信号受信工程により受信した画像信号を、画像表示装置に表示可能な形式へ変換する変換工程と、

前記変換工程により変換された画像信号を前記画像表示装置に表示する画像表示制御工程とを有することを特徴とする画像表示方法。

30

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、画像処理装置、画像表示装置及び画像伝送システムに関し、特に、パーソナルコンピュータ（以下、「PC」と表記する。）等の情報処理装置により生成した画像をプロジェクタなどの表示装置に送信するために用いて好適な技術に関する。

【背景技術】

【0002】

近年、会議や研修会等でプレゼンテーションを行う際は、PCから受信した画像信号をスクリーンに投射させるプロジェクタを用いることが多い。通常、PCとプロジェクタとはアナログRGBケーブルを用いたアナログ接続によって画像信号の送受信を行っていた。

40

【0003】

しかしながら、アナログ接続では、接続に手間がかかる、遠隔地からの接続が困難、複数台のPCと同時に接続したい、といった理由からネットワークを介してPCとプロジェクタとを接続する画像転送システムが注目されている。

【0004】

この画像転送システムを実現するために、従来では、PCなどの画像処理装置に表示された画面について定期的に画面キャプチャを行い、画面キャプチャによって取得された画

50

像データをプロジェクタなどの画像表示装置に送信している。

【0005】

前述のような画像転送システムにおいて、PCなどの画像処理装置に表示された画面の全てを常にキャプチャして送る場合には処理時間が多くかかってしまう。このため、フレームレートが落ちてしまい、プロジェクタなどの表示装置で滑らかな表示を行うことができなかった。

【0006】

このような不都合を解消するために、特許文献1では、入力画像を複数個のタイル画像に分割し、タイル画像ごとに、画像フレームを参照して、差分値を検出する。そして、この差分値が閾値以上の場合に、このタイル画像を更新する必要があると判断し、符号化圧縮部およびフレームメモリに送り、画像を更新する方法が開示されている。

10

【先行技術文献】

【特許文献】

【0007】

【特許文献1】特開2001-061066号公報

【発明の概要】

【発明が解決しようとする課題】

【0008】

しかしながら、タイル画像ごとに差分をチェックして送る場合、変化があった場所を常に送ることになり、ユーザが着目していない領域の画像情報も送られてしまう。そのため、結果的にユーザが見たい箇所の領域までフレームレートが落ちてしまい、ユーザにとって違和感のある画像表示になってしまうという問題があった。

20

本発明は前述の問題点に鑑み、ユーザが見たい箇所の領域のフレームレートが低下してしまうのを防止できるようにすることを目的とする。

【課題を解決するための手段】

【0009】

本発明の画像処理装置は、ネットワークを介して画像表示装置へ画像を伝送する画像伝送手段と、アプリケーションごとに生成される矩形領域を表示部に表示するポインタ入力手段と、前記表示部へ出力する出力画像を取得する画像取得手段と、前記表示部に出力される矩形領域を判別する矩形領域判別手段と、前記矩形領域判別手段により判別された矩形領域の中で、特定の矩形領域を判別する特定矩形領域判別手段と、前記特定矩形領域判別手段により判別された特定の矩形領域のフレームレートを変更して前記画像伝送手段により前記画像表示装置へ伝送するフレームレート制御手段とを有することを特徴とする。

30

【発明の効果】

【0010】

本発明によれば、ユーザが着目している領域のフレームレートを上げることで、ユーザにとって違和感の無い画像表示を実現することが可能となる。

【図面の簡単な説明】

【0011】

【図1】PCとプロジェクタとのシステム構成を説明するブロック図である。

40

【図2】表示システムの概略構成を説明するブロック図である。

【図3】優先ウィンドウを設定する際の処理を説明するフローチャートである。

【図4】画像をプロジェクタに転送する処理を説明するフローチャートである。

【図5】優先ウィンドウ送信プログラムの処理を説明するフローチャートである。

【図6】プロジェクタの処理を説明するフローチャートである。

【図7】ウィンドウ状態が変化した場合の処理を説明するフローチャートである。

【図8】マウス状態が変化した場合の処理を説明するフローチャートである。

【図9】ウィンドウ画面の一例を示す図である。

【図10】ウィンドウの属性情報を示す図である。

【図11】優先ウィンドウの情報を示す図である。

50

【発明を実施するための形態】

【0012】

以下、図面を参照して本発明の実施形態を詳細に説明する。

図1は、本実施形態の画像処理装置を用いた画像伝送システムの一実施形態を示す説明図である。本実施形態の画像伝送システム10は、PC100とプロジェクタ200とを備えており、PC100とプロジェクタ200とはLAN（ネットワーク）300のような回線を通して通信ネットワークを形成している。

【0013】

各部の簡単な役割を説明すると、PC100は、ディスプレイに表示されている画像信号を、通信ネットワークを介して画像表示装置に送信する。本実施形態においては、画像表示装置としてプロジェクタ200を例示している。プロジェクタ200は、PC100から送られた画像信号を受信して、出力できる形への変換を行い表示する。このために、ネットワーク300を介してPC100から送られる画像信号を受信する信号受信手段と、受信手段により受信した画像信号を、画像表示装置に表示可能な形式へ変換するデコーダ204（変換手段）とを有する。また、204により変換された画像信号を画像表示装置に表示する投射画像制御部207（画像表示制御手段）を有する。

【0014】

またこのとき、マウス100aによってPC100におけるウィンドウシステム上のポインタが操作されるが、ポインタ表示位置は、画像データとは別にプロジェクタ200に送信する。なお、ここでいうウィンドウシステムとは、例えばマイクロソフト社のオペレーティングシステム（以下OS）であるWindows（登録商標）によって実現される、Graphical User Interface（GUI）のことである。GUIにより、複数のタスクに固有の領域を割当てて画面出力を多重化することが可能になる。

【0015】

(第1の実施形態)

本実施形態では、図1のPC100とプロジェクタ200とによる表示システム400の概略ブロック構成を図2に示す。

図2に示すように、PC100は、CPU101と、CPU101の処理手順を記述した制御プログラムを記憶するためのROM102と、ワークメモリとして一時的に制御プログラムやデータを格納するRAM103を持つ。

【0016】

そして、アプリケーションやOSなどの各プログラムとデータを格納するためのハードディスクドライブ（HDD）104と、PC100の表示する画像データを圧縮するためのエンコーダ105を備える。更に、LAN等を経由してプロジェクタ200やサーバなどと通信するためのインタフェースであるネットワークIF106と、キーボードやポインティングデバイス（マウス等のポインタ入力手段）からのユーザ入力を処理するユーザIF107を備える。

【0017】

そして、PC100の表示する画像が記憶されているVRAM108と、画像処理部109と、表示制御部110と、液晶ディスプレイなどの表示部111と、これら各要素を接続する内部バス112を備えている。

【0018】

プロジェクタ200は、CPU201と、CPU201の処理手順を記述した制御プログラムを記憶するためのROM202と、ワークメモリとして一時的に制御プログラムやデータを格納するRAM203を備える。そして、所定の圧縮方式で圧縮されたデータをデコードするためのデコーダ204を持つ。更に、PC100との接続用インタフェースであるネットワークIF205と、各種の操作ボタンや操作リモコンなどからのユーザ入力を処理するユーザIF206を備える。

【0019】

そして、画像を出力するための投射画像制御部207と、入力画像を投射する液晶パネ

10

20

30

40

50

ル及びその駆動ドライバ、レンズ及びその駆動系及び光源で構成される投影部 208 と、これら各要素を接続する内部バス 209 と、を備えている。

【0020】

次に、本実施形態における PC 100 の処理の流れを、図 3 ~ 図 5 を用いて説明する。

図 3 に、複数のウィンドウから優先ウィンドウを設定する際の処理の例を示す。

PC 100 は CPU 101 からの命令により優先ウィンドウ判定用のプログラムを HDD 104 から読み込み処理が開始される。

【0021】

プログラムが開始されると、表示部 111 の最前面に表示されているアクティブなウィンドウを Windows の Application Programming Interface (以下、API) である関数 GetActiveWindow で取得する。そして、RAM 103 に保持する (ステップ S001)。次に、関数 GetWindowText により、各ウィンドウのテキスト情報を読み取り、ROM 102 に格納されているアプリケーション情報と比較し、各ウィンドウにおけるアプリケーションを判別する。そして、判別した各ウィンドウのアプリケーション情報は RAM 103 に格納する (ステップ S002)。

10

【0022】

次に、API である関数 GetWindowRect を用いて各ウィンドウにおける左上の原点座標 (X1、Y1)、X 軸座標上のウィンドウサイズ、Y 軸座標上のウィンドウサイズ情報から、アプリケーションごとに生成される矩形領域を取得する。そして、RAM 103 に保持する (ステップ S003)。各ウィンドウのハンドル情報とレイヤ情報は GetNextWindow で取得し、取得した各ウィンドウのサイズ情報と、アクティブなウィンドウサイズ情報よりレイヤの順番を把握する。レイヤの順番と各ウィンドウの座標とサイズ情報により、表示部 111 に表示されている、各ウィンドウの表示されている面積を算出し RAM 103 に保持する (ステップ S004)。

20

【0023】

ここで、図 9 にウィンドウによる画面構成の一例を示す。

表示部 111 に表示されている画像 900 の中に、アクティブな Window 901、その下に配置されている Window 902、更にその下に配置されている Window 903、マウス 904 で構成されている。

30

【0024】

例えば、図 9 に示されるようにアクティブな Window 901 においては、Window 901 のウィンドウサイズが面積となる。Window 902 においては、Window 901 によって隠れている領域を減算した領域である。また、Window 903 の面積は Window 901 と Window 902 によって隠れている領域を減算した領域である。以上、ステップ S001 ~ ステップ S004 によりウィンドウの構成を知ることができる。

【0025】

ここで、以上のステップにより各ウィンドウの情報を格納した際に RAM 103 に格納されるテーブル情報の例を図 10 に示す。

40

図 10 に示すように、ウィンドウの属性情報 1001 の項目には、各ウィンドウ ID 1002 の各ウィンドウ 001 ~ 004 に対応して、アプリケーション 1003、アクティブウィンドウ 1004、面積 1005 のパラメータが存在する。ステップ S001 によりアクティブウィンドウ 1004 がウィンドウ ID 002 のウィンドウであることが分かる。

【0026】

また、ステップ S002 により各ウィンドウのアプリケーション 1003 が分かり、図 10 に示すようにウィンドウ 001 が表計算、ウィンドウ 002 が動画プレイヤー、ウィンドウ 003 がピクチャー、ウィンドウ 004 がブラウザとして登録される。また、ステップ S003、ステップ S004 により各ウィンドウの面積 1005 が計算される。これ

50

により、ウィンドウ 0 0 1 は 2 4 0 0 ピクセル、ウィンドウ 0 0 2 は 1 6 0 0 ピクセル、ウィンドウ 0 0 3 は 4 8 0 0 ピクセル、ウィンドウ 0 0 4 は 6 0 0 ピクセルであることが分かる。

【 0 0 2 7 】

次に、ステップ S 0 0 5 ~ ステップ S 0 1 1 により各ウィンドウが優先ウィンドウに該当するかどうかを判別していく。

まず、動画ウィンドウが、矩形領域が動画を表示する矩形領域であるかを判別する動画領域判別を行う。ステップ S 0 0 2 により各ウィンドウのアプリケーションが判別可能であるので、あらかじめ R O M 1 0 2 に登録されている動画アプリケーションであるかを判別する (ステップ S 0 0 5)。この判別の結果、動画アプリケーションであれば (Y E S)、該当するウィンドウを優先ウィンドウ (動画) に設定して R A M 1 0 3 に優先ウィンドウ情報として保持し (ステップ S 0 0 8)、ステップ S 0 0 6 の処理に進む。

10

【 0 0 2 8 】

一方、ステップ S 0 0 5 において、動画ウィンドウでなければ (N O)、ステップ S 0 0 6 の処理に進む。ステップ S 0 0 6 では、ステップ S 0 0 1 で取得した R A M 1 0 3 に保持されているアクティブウィンドウ情報から、該当するウィンドウがアクティブウィンドウであるかを判定する (ステップ S 0 0 6)。

【 0 0 2 9 】

この判定の結果、アクティブウィンドウであれば (Y E S)、該当するウィンドウを優先ウィンドウ (アクティブ) に設定して R A M 1 0 3 に優先ウィンドウ情報として保持して (ステップ S 0 0 9)、ステップ S 0 0 7 の処理に進む。

20

一方、ステップ S 0 0 6 において、アクティブウィンドウ情報でなければ (N O)、ステップ S 0 0 7 の処理に進む。

【 0 0 3 0 】

ステップ S 0 0 7 では、ステップ S 0 0 3、ステップ S 0 0 4 で取得した各ウィンドウの表示面積情報より、該当するウィンドウが最大領域サイズのウィンドウであるかを判別 (最大矩形領域判別) する (ステップ S 0 0 7)。この判定の結果、最大サイズウィンドウであれば (Y E S)、該当するウィンドウを優先ウィンドウ (最大サイズ) に設定して R A M 1 0 3 に優先ウィンドウ情報として設定して (ステップ S 0 1 0)、ステップ S 0 1 1 の処理に進む。すなわち、矩形領域サイズ変更を行う。

30

【 0 0 3 1 】

一方、ステップ S 0 0 7 において、最大サイズウィンドウ情報でなければ (N O)、ステップ S 0 1 1 の処理に進む。ステップ S 0 1 1 の処理では全ウィンドウについてステップ S 0 0 5 ~ ステップ S 0 1 0 の処理を実施したかどうかをチェックする (ステップ S 0 1 1)。全てのウィンドウに対して処理が終わっていなければ (N O) ステップ S 0 0 5 に戻り処理を継続し、全てのウィンドウに対して処理が終わっていれば (Y E S) 処理を終える。

【 0 0 3 2 】

ここで、以上のステップにより優先ウィンドウの情報を格納した際に R A M 1 0 3 に格納されるテーブル情報の例を図 1 1 に示す。

40

図 1 1 に示すように、優先ウィンドウ 1 1 0 1 の項目には各ウィンドウ I D 1 1 0 2 のウィンドウ 0 0 1 ~ 0 0 4 に対して、動画 1 1 0 3、アクティブ 1 1 0 4、最大サイズ 1 1 0 5 が設定されている。ステップ S 0 0 8 により設定された動画の優先ウィンドウはウィンドウ 0 0 2 に設定されている。また、ステップ S 0 0 9 により設定されたアクティブの優先ウィンドウはウィンドウ 0 0 2 に設定されている。また、ステップ S 0 1 0 により設定された最大サイズの優先ウィンドウはウィンドウ 0 0 3 に設定されていることが分かる。

【 0 0 3 3 】

ここで、例えばウィンドウ 0 0 2 のように、ウィンドウによっては複数の条件を満たし、ステップ S 0 0 8、ステップ S 0 0 9、ステップ S 0 1 0 で優先ウィンドウに設定する

50

ウィンドウが存在する場合がある。この場合、各優先ウィンドウの情報を保持するものとする。

以上、説明した図3における優先ウィンドウの設定プログラムは、以降述べるプログラムよりも早い周期で独立して常時動作しているものとする。

【0034】

次に、図4のフローチャートを参照しながら、PC100が表示部111に表示している画像をプロジェクタ200に転送する際の処理の流れを示す。

PC100は、CPU101からの命令により画像転送用のプログラムをHDD104から読み込み処理が開始される。このプログラムは、図3で説明した優先ウィンドウ判定用のプログラムと同時に開始される。本実施形態においては、PC100は、画像取得手段、矩形領域判別手段、特定矩形領域判別手段、及びフレームレート制御手段を有する。

【0035】

画像取得手段は、表示部111へ出力する出力画像を取得する。矩形領域判別手段は、表示部111に出力される矩形領域を判別する。特定矩形領域判別手段は、矩形領域判別手段により判別された矩形領域の中で、特定の矩形領域を判別する。フレームレート制御手段は、特定矩形領域判別手段により判別された特定の矩形領域のフレームレートを変更してプロジェクタ200へ伝送する。本実施形態においては、動画再生が開始されたことが判別された場合に、動画を表示する矩形領域のフレームレートを上げるようにしている。

【0036】

プログラムが開始されると、優先ウィンドウが設定されているかどうかをRAM103の格納データにより判別する(ステップS101)。この判別の結果、優先ウィンドウが設定されている場合(YES)、次のステップS102に進み、そうでなければ(NOの場合)ステップS106に進む。なお、プログラム起動時には優先ウィンドウは設定されていないので、自動的にステップS106の処理に進む。

【0037】

優先ウィンドウが設定されている場合は、設定されている優先ウィンドウが所定の大きさ以上で無い場合には重要なウィンドウではないため所定の大きさ以上であるか判定する(ステップS102)。ステップS102で判定する所定の大きさは、ROM102に工場出荷時に格納されている情報であってもよいし、ユーザIF107を介して設定された値であってもよい。この判定の結果、所定の大きさなければ(NOの場合)、ステップS106に進み、所定の大きさ以上であれば(YES)次のステップS103に進む。

【0038】

次に、ネットワークIF106で利用できる帯域状態が所定の値以上であるか、帯域状態を検知する帯域状態検知を行う(ステップS103)。所定の帯域値はROM102に工場出荷時に格納されている情報であってもよいし、ユーザIF107を介して設定された値であってもよい。所定の帯域値以上であれば全画面を転送しても問題ないので、ステップS106に進む。

【0039】

一方、帯域状態が所定以下である場合(YESの場合)には、全画面を常に送る帯域が確保できていないため、設定されている優先のウィンドウを送信する優先ウィンドウ送信プログラムを起動する(ステップS104)。そして、前述した特定矩形領域判別手段により特定の矩形領域を設定する。また、設定された特定の矩形領域と同一の矩形領域が複数存在する場合においては、複数存在する特定の矩形領域の優先順位設定を行う。優先ウィンドウ送信プログラムの処理の流れについては後述する。

【0040】

優先ウィンドウ送信プログラムを起動したあとは、優先ウィンドウプログラムによって優先ウィンドウの画像がプロジェクタ200に転送されたかどうかを判別する(ステップS105)。この判別の結果、優先ウィンドウの画像送信が完了していなければ(NOの場合)、画像送信が完了するまで待ち続け、完了したら次のステップS106に進む(Y

10

20

30

40

50

ES)。

【0041】

次に、全画面転送の処理をスキップするフラグ情報がRAM103に格納されているかどうかを判定する(ステップS106)。この判定の結果、スキップするフラグ情報がRAM103に格納されていれば全画面の転送は省略し(YES)、格納されていなければ次のステップS107に進む(NO)。

【0042】

次に、PC100の表示制御部110を介して表示部111に出力されている画面をキャプチャする。キャプチャは、WindowsのAPIである関数BitBlt(bit block transfer)を用いてVRAM108に保持する画面データを取得し、RAM103に保持する(ステップS107)。キャプチャした後、エンコーダ105により、例えばJPEGなどの所定の圧縮方式によりステップS107で取得した画像データをエンコードする(ステップS108)。この際、圧縮方式はどのような圧縮方式であってもかまわない。圧縮した画像データはネットワークIF106を介してプロジェクタ200に転送される(ステップS109)。

10

【0043】

図4におけるステップS104で起動した優先ウィンドウ送信プログラムの処理の流れに関して図5を用いて説明する。

ある1つの優先ウィンドウが動画であるかどうかをRAM103に格納されている優先ウィンドウ情報により判定する(ステップS201)。優先ウィンドウが動画であれば(YES)、その優先ウィンドウが更にアクティブであるかを判定する(ステップS202)。この判定の結果、優先ウィンドウが動画でなければ(NOの場合)、その優先ウィンドウがアクティブであるかを判別する(ステップS204)。

20

【0044】

一方、ステップS202において、優先ウィンドウが動画であり更にアクティブでもある場合(YES)、該当する優先ウィンドウが最も優先されるウィンドウであると判別される。この場合は、ステップS212に進んで処理を行う。ステップS212では、PC100の表示制御部110を介して表示部111に出力されている画面をキャプチャする。キャプチャは関数BitBltを用いてVRAM108に保持する画面データを取得し、RAM103に保持する(ステップS212)。

30

【0045】

キャプチャした選択中の画面は、エンコーダ105により例えばJPEGなどの所定の圧縮方式によりステップS212で取得した画像データの中から優先ウィンドウの領域をエンコードする(ステップS213)。この際、圧縮方式はどのような圧縮方式であってもかまわない。圧縮した画像データはネットワークIF106を介してプロジェクタ200に転送される(ステップS214)。

【0046】

次に、RAM103に格納されている優先ウィンドウ情報に変化がないかどうかを判別し(ステップS215)、変化がなければ(NO)ステップS212~S214の処理を継続する。また、優先ウィンドウ情報に変化がある場合には(YES)処理を終えて優先ウィンドウの処理が完了したことを図4で説明した画像転送用のプログラムに通知して処理を終える(ステップS216)。

40

【0047】

ここで、ステップS202において、優先ウィンドウが動画であるがアクティブではない場合(NO)、ステップ203に進んで優先ウィンドウの動画が再生中であるかどうかを判別する(動画再生判別を行う)。再生しているかどうかは直接動画アプリケーションに問い合わせるか、もしくは数コマ画面をキャプチャして変化を検知することで実現することが可能である。動画が再生中でなければ(NO)ステップS207に進み、動画が再生中であれば(YES)該当する優先ウィンドウの送信回数を設定し、RAM103に保持する(ステップS206)。

50

【 0 0 4 8 】

送信回数は、ウィンドウのサイズや、各優先ウィンドウの優先度から設定される。優先度は、例えば“ 動画 > アクティブ > 最大サイズ ”といったように、ROM 102 に格納されている優先度情報に従って順位付けを行う。優先ウィンドウの送信回数を設定した後はステップ S 207 に進む。

【 0 0 4 9 】

ステップ S 204 では、優先ウィンドウがアクティブであるかどうかを判別し、優先ウィンドウであれば (YES) ステップ S 206 に進み、そうでなければ (NO) 優先ウィンドウが最大サイズウィンドウであるかを判別する (ステップ S 205)。最大サイズウィンドウであれば (YES) ステップ S 206 に進み、そうでなければ (NO) ステップ S 207 に進む。

10

【 0 0 5 0 】

ステップ S 207 では、全ての優先ウィンドウに対してチェックが終わったかを判定し、チェックが終わっていなければ (NO) ステップ S 201 の処理に戻り、チェックが終わっていれば (YES) ステップ S 208 に進む。

【 0 0 5 1 】

ステップ S 208 では、PC 100 の表示制御部 110 を介して表示部 111 に出力されている画面をキャプチャする。キャプチャは、関数 BitBlit を用いて VRAM 108 に保持する画面データを取得し、RAM 103 に保持する。キャプチャした後、エンコーダ 105 により、例えば JPEG などの所定の圧縮方式によりステップ S 208 で取得した画像データの中でステップ S 206 で設定された優先ウィンドウの領域をエンコードする (ステップ S 209)。

20

【 0 0 5 2 】

ステップ S 206 で設定された回数は優先ウィンドウごとに異なるので、規定回数に達している優先ウィンドウは、ステップ S 209 における圧縮領域から順次除外して設定を解除する。なお、圧縮方式はどのような圧縮方式であってもかまわないものとする。圧縮した優先ウィンドウの画像データは、ネットワーク IF 106 を介してプロジェクタ 200 に転送される (ステップ S 210)。

【 0 0 5 3 】

次に、ステップ S 206 で設定した送信回数分送信したかどうかを判定する (ステップ S 211)。設定した送信回数分、全ての優先ウィンドウを送信していれば (YES) ステップ S 216 に進み、そうでなければ (NO) ステップ S 208 ~ S 210 の処理を継続する。前述したようにステップ S 206 では優先ウィンドウの処理が完了したことを図 4 で説明した画像転送用のプログラムに通知して処理を終える。

30

【 0 0 5 4 】

次に、図 6 を用いて、プロジェクタ 200 の処理の流れを説明する。

図 6 に示すように PC 100 からネットワーク IF 106 を介して送信される画像データをネットワーク IF 205 で受信して RAM 203 に画像データを格納する (ステップ S 301)。次に、受信して RAM 203 に格納されている画像データをデコーダ 204 によりデコード処理を行い (ステップ S 302)、RAM 203 にデコード後の画像データを保存する。次に全ての画像データについてデータ受信が完了したかどうかを判別し (ステップ S 303)、全ての画像データについてデータ受信が完了していない場合にはステップ S 301 に戻る (NO)。全ての画像データについてデータ受信が完了している場合には、投射画像制御部 207 により投影部 208 に画像を表示する (ステップ S 304)。

40

【 0 0 5 5 】

以前述べてきた実施形態により、PC 100 の特定のウィンドウを優先ウィンドウとして設定することで、ユーザにとって重要な画面情報を違和感なくプロジェクタ 200 で再生することができる。

【 0 0 5 6 】

50

(第2の実施形態)

ウィンドウの状態によって優先ウィンドウを変更する

本実施形態では、ウィンドウの状態が変化した場合における実施形態について述べる。

本実施形態の表示システムの概略ブロック構成は、第1の実施形態と同様に図2に示した構成とする。また、PC100の処理の流れは第1の実施形態で説明した処理と同様とし、プロジェクトについても同様である。

【0057】

次に、図7を用いて本実施形態におけるPC100の処理の流れについて説明する。

図3のステップS003～S004間で取得したウィンドウ情報が格納されているRAM103からの情報により、ウィンドウサイズが変化したかどうかを判別する(ステップS401)。ウィンドウサイズが変化した場合には(YES)ステップS403に移行する。

10

【0058】

一方、ウィンドウサイズが変化していなければ(NO)、ステップS003～S004で取得したウィンドウの位置情報によりウィンドウが移動したかどうかを判別する(ステップS402)。ステップS402の判別の結果、矩形領域移動が行われていなければステップS401の処理に戻る(NO)。一方、矩形領域移動が行われていた場合には、次のステップS403に進む(YES)。

【0059】

次に、変化があったウィンドウが優先ウィンドウであるかを判別する(ステップS403)。優先ウィンドウ情報は図3におけるステップS005～S011により設定されるウィンドウであり、RAM103に優先ウィンドウの情報が格納されている。該当するウィンドウが優先ウィンドウで無い(NO)場合には、優先ウィンドウの送信回数の設定を0に設定してRAM103に記録する(ステップS405)。

20

【0060】

RAM103に保持される優先ウィンドウの送信回数が0に設定されたことで、図5に示したステップS206における送信回数を0に設定するものとする。従って、図5におけるステップS206ではステップS201～S205の条件に関わらず送信回数を0に設定し、優先ウィンドウの送信処理をすぐに終了させる。

【0061】

一方、ステップS403において該当するウィンドウが優先ウィンドウであった場合には(YES)、優先ウィンドウの面積が拡大されているかどうかを判別する(ステップS404)。優先ウィンドウの面積が拡大されている場合には(YES)、画像サイズが大きくなるのでフレームレートを落とすため優先ウィンドウの送信回数の減算値をRAM103に記録する。RAM103に記録された減算値に従って、図5におけるステップS206ではステップS201～S205の条件により設定された送信回数から、設定された減算値を減算して送信回数を設定する。そして、設定された送信回数に従って優先ウィンドウを送信する(ステップS406)。

30

【0062】

一方、ステップS404において、優先ウィンドウの面積が縮小されている場合には(NO)、画像サイズが小さくなるのでフレームレートを上げるため優先ウィンドウの送信回数の加算値をRAM103に記録する。RAM103に記録された加算値に従って、図5におけるステップS206ではステップS201～S205の条件により設定された送信回数から、設定された加算値を加算して送信回数を設定する。そして、設定された送信回数に従って優先ウィンドウを送信する(ステップS407)。

40

【0063】

以上、述べてきた実施形態により、PC100のウィンドウのフレームレートを動的に変えることで、ユーザにとって重要な画面情報を違和感なくプロジェクト200で再生することができる。

【0064】

50

(第3の実施形態)

本実施形態では、マウスの状態(ポインタ状態)によって優先ウィンドウを変化させる場合の実施形態について述べる。本実施形態における概略ブロック構成は、第1の実施形態と同様に図2に示した構成とする。また、PC100の処理の流れは第1の実施形態で説明した処理と同様とし、プロジェクトについても同様である。

【0065】

図8に、マウスの状態に応じて送信回数を変更する処理の流れを説明するフローチャートを示す。

APIである関数GetCursorInfo、GetCursorPosなどによりマウスの位置としての座標情報を取得する(ステップS501)。

10

【0066】

次に、ステップS501で取得した座標が、図3により取得した優先ウィンドウの座標情報からマウスの位置の状態(ポインタ状態)が優先ウィンドウの中に存在するかどうかを判断する(ステップS502)。このポインタ状態検出の結果、優先ウィンドウ内に存在しない場合(NOの場合)は、ステップS503に進んでマウスがウィンドウをドラッグしているかどうかを判別する。

【0067】

ステップS503の判別の結果、マウスがウィンドウをドラッグしている場合(YESの場合)には、ステップS505に進み、優先ウィンドウの送信回数の設定を0に設定してRAM103に記録する。RAM103に保持される優先ウィンドウの送信回数が0に設定されたことで、図5に示したステップS206における送信回数を0に設定するものとする。したがって、図5におけるステップS206ではステップS201~S205の条件に関わらず送信回数を0に設定し、優先ウィンドウの送信処理をすぐに終了させる。

20

【0068】

一方、ステップS503の判別の結果、マウスがウィンドウをドラッグしていない場合には(NOの場合)、ステップS504に進み、マウスが動いていないか判別する。この判別の結果、マウスが動いていない場合には(NO)ステップS507の処理に進む。一方マウスが動いている場合には(YES)、優先ウィンドウの送信回数の減算値をRAM103に記録する(ステップS506)。

【0069】

RAM103に記録された減算値に従って、図5におけるステップS206ではステップS201~S205の条件により設定された送信回数から、設定された減算値を減算して送信回数を設定し、設定された送信回数に従って優先ウィンドウを送信する。

30

【0070】

ステップS502において、優先ウィンドウ内に存在する場合は(YES)、ステップS507の処理に進む。ステップS507ではPC100の表示制御部110を介して表示部111に出力されている画面をキャプチャする。キャプチャは関数BitBltを用いてVRAM108に保持する画面データを取得し、RAM103に保持する。

【0071】

キャプチャした後、エンコーダ105により、例えばJPEGなどの所定の圧縮方式によりステップS507で取得した画像データの中から優先ウィンドウの領域をエンコードする(ステップS508)。この際、圧縮方式はどのような圧縮方式であってもかまわない。圧縮した画像データはネットワークIF106を介してプロジェクト200に転送される(ステップS509)。

40

【0072】

次に、マウスの位置としての座標情報を取得し(ステップS510)、マウスが動作しているかどうかを判断する(ステップS511)。マウスが動作している場合には(YES)処理を終え、マウスが動作していない場合にはステップS507に戻り、優先ウィンドウを送信し続ける。なお、ステップS507~ステップS511の処理を行う場合には、図5における優先ウィンドウの送信プログラムは終了してもよい。

50

以上、述べてきた実施形態によりPC 100のマウスの状態に応じて適切にフレームレートを変えることで、ユーザにとって重要な画面情報を違和感なくプロジェクタ200で再生することができる。

【0073】

(その他の実施形態)

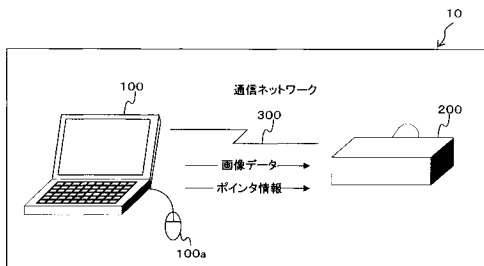
また、本発明は、以下の処理を実行することによっても実現される。即ち、上述した実施形態の機能を実現するソフトウェア(コンピュータプログラム)を、ネットワーク又は各種のコンピュータ読み取り可能な記憶媒体を介してシステム或いは装置に供給する。そして、そのシステム或いは装置のコンピュータ(またはCPUやMPU等)がプログラムを読み出して実行する処理である。

【符号の説明】

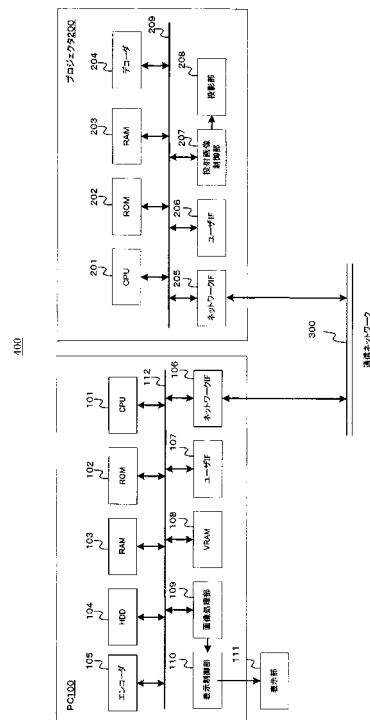
【0074】

100 PC、200 プロジェクタ、100a マウス、300 ネットワーク

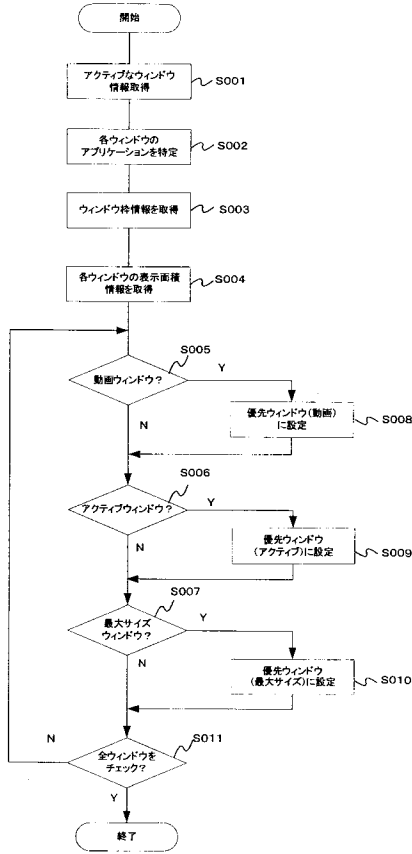
【図1】



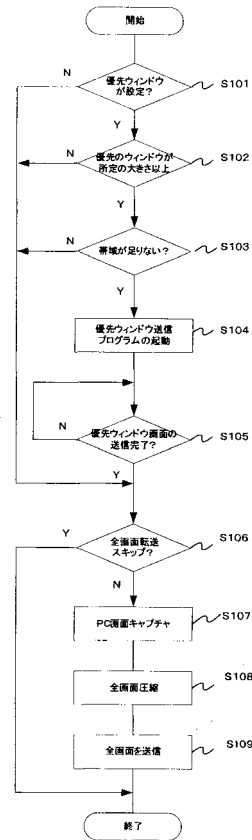
【図2】



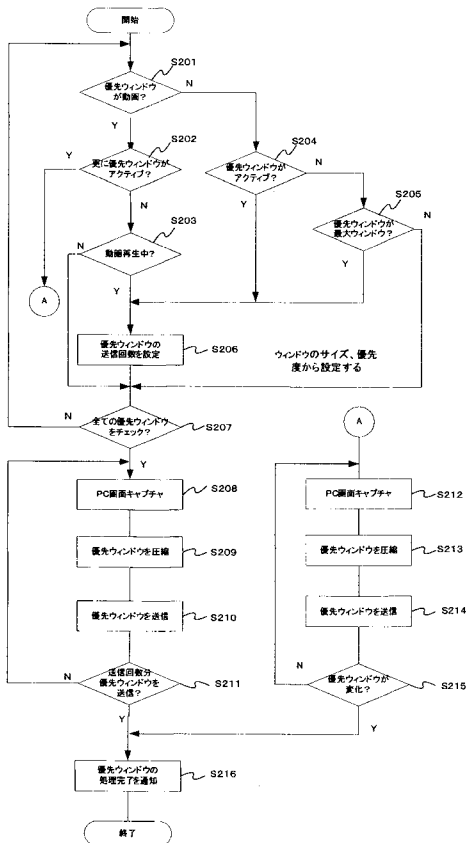
【図3】



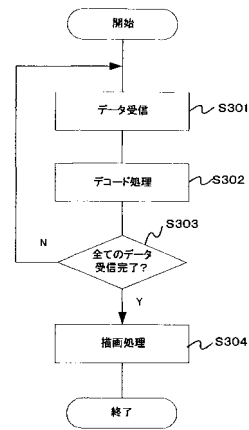
【図4】



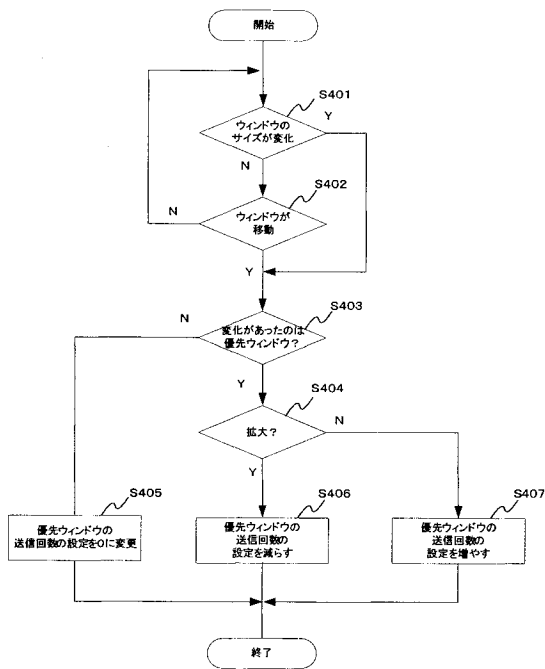
【図5】



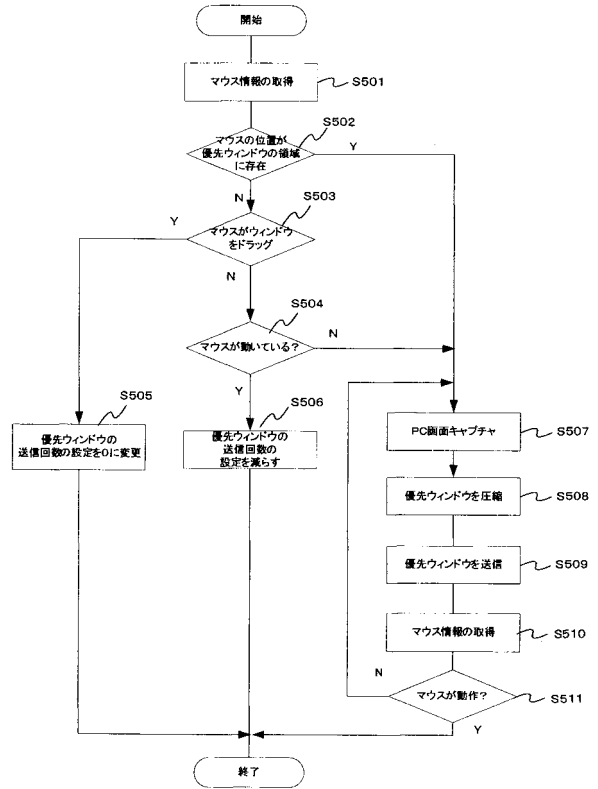
【図6】



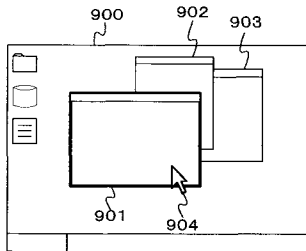
【 図 7 】



【 図 8 】



【 図 9 】



【 図 1 1 】

| ウィンドウID (1102) | 優先ウィンドウ(1101) | | |
|-------------------|---------------|-----------------|-----------------|
| | 動画 (1103) | アクティブ (1104) | 最大サイズ (1105) |
| 001 | | | |
| 002 | ○ | ○ | |
| 003 | | | ○ |
| 004 | | | |

【 図 1 0 】

| ウィンドウID (1002) | ウィンドウ属性情報(1001) | | |
|-------------------|--------------------|-----------------|--------------|
| | アプリケーション (1003) | アクティブ (1004) | 面積 (1005) |
| 001 | 表計算 | | 2400pixel |
| 002 | 動画プレイヤー | ○ | 1600pixel |
| 003 | ピクチャー | | 4800pixel |
| 004 | ブラウザ | | 800pixel |

 フロントページの続き

| (51)Int.Cl. | F I | テーマコード(参考) |
|--------------------------------|-----------------------|------------|
| H 0 4 N 5/74 (2006.01) | G 0 9 G 5/00 5 2 0 V | |
| G 0 6 F 3/153 (2006.01) | G 0 9 G 5/00 5 5 5 D | |
| H 0 4 N 7/173 (2011.01) | G 0 9 G 5/00 5 5 0 C | |
| | G 0 9 G 5/36 5 2 0 E | |
| | H 0 4 N 5/74 Z | |
| | G 0 6 F 3/153 3 3 0 A | |
| | H 0 4 N 7/173 6 3 0 | |

Fターム(参考) 5C082 AA03 AA21 BA27 BA41 BB01 BC19 BD02 CA33 CA34 CA52
 CA54 CA84 CB05 MM10
 5C164 FA10 MC05P UA02P UB81P