

(19)日本国特許庁(JP)

## (12)特許公報(B2)

(11)特許番号

特許第7317561号

(P7317561)

(45)発行日 令和5年7月31日(2023.7.31)

(24)登録日 令和5年7月21日(2023.7.21)

(51)国際特許分類

F I

G 0 6 F 3/04886(2022.01)

G 0 6 F 3/04886

G 0 6 F 3/04842(2022.01)

G 0 6 F 3/04842

G 0 6 F 16/583(2019.01)

G 0 6 F 16/583

H 0 4 N 1/00 (2006.01)

H 0 4 N 1/00 3 5 0

G 0 6 F 16/14 (2019.01)

G 0 6 F 16/14 1 0 0

請求項の数 21 (全33頁)

(21)出願番号 特願2019-80384(P2019-80384)  
 (22)出願日 平成31年4月19日(2019.4.19)  
 (65)公開番号 特開2020-177521(P2020-177521  
 A)  
 (43)公開日 令和2年10月29日(2020.10.29)  
 審査請求日 令和4年4月15日(2022.4.15)  
 早期審査対象出願

(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 110001243  
 弁理士法人谷・阿部特許事務所  
 (72)発明者 白石 憲一  
 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 菅原 浩二

最終頁に続く

(54)【発明の名称】 タッチパネルを用いた文字入力のための画像処理装置、その制御方法及びプログラム

## (57)【特許請求の範囲】

## 【請求項1】

文書をスキャンして得られたスキャン画像を含むファイルのファイル名を設定する第1のUI画面であって、前記スキャン画像から認識された複数の文字列が連結された連結文字列を含む第1のUI画面を表示するように制御する第1の表示制御手段と、

前記第1のUI画面上での前記複数の文字列のうちの第1の文字列に関するユーザからの指示に応じて、前記複数の文字列のうちの当該第1の文字列以外の文字列の何れもが連結されていない当該第1の文字列を編集可能に表示する第2のUI画面を表示するように制御する第2の表示制御手段と、

前記第2の表示制御手段により編集可能に表示された前記第1の文字列への編集をユーザから受付ける受付手段と、  
 を有し、

前記受付手段で受け付けた前記第1の文字列への編集に基づいて編集された連結文字列が、前記ファイルのファイル名として設定される、  
 ことを特徴とする情報処理装置。

## 【請求項2】

前記第1のUI画面に含まれる連結文字列は、前記複数の文字列がユーザにより選択された種類の区切り文字を挟んで連結された連結文字列であり、  
前記第1の文字列は、前記連結文字列を構成する前記複数の文字列のうちの先頭の文字列であって、

10

20

前記第 2 の表示制御手段は、前記指示に応じて、前記複数の文字列のうちの前記第 1 の文字列以外の文字列および前記区切り文字の何れもが連結されていない前記第 1 の文字列を編集可能に表示する前記第 2 の UI 画面を表示するように制御する、  
ことを特徴とする請求項 1 に記載の情報処理装置。

【請求項 3】

前記複数の文字列は、前記スキャン画像から認識された 2 つの文字列であり、  
前記第 2 の表示制御手段は、前記 2 つの文字列のうちの前記第 1 の文字列に関するユーザからの指示に応じて、前記 2 つの文字列のうちの当該第 1 の文字列以外の文字列が連結されていない当該第 1 の文字列を編集可能に表示する前記第 2 の UI 画面を表示するように制御する、  
ことを特徴とする請求項 1 に記載の情報処理装置。

10

【請求項 4】

前記受付手段で受け付けた前記第 1 の文字列への編集に基づいて編集された連結文字列は、前記第 1 の文字列への編集を受け付けた後に前記第 1 の UI 画面上に表示されることを特徴とする請求項 1 に記載の情報処理装置。

【請求項 5】

前記第 1 の文字列は、前記指示に応じて、前記第 1 の文字列を編集するためのソフトウェアとともに前記第 2 の UI 画面上に表示されることを特徴とする請求項 1 に記載の情報処理装置。

【請求項 6】

前記情報処理装置は、スキャナを有し、  
前記スキャン画像は、前記スキャナで前記文書をスキャンして得られることを特徴とする請求項 1 に記載の情報処理装置。

20

【請求項 7】

前記情報処理装置は、サーバと通信可能であり、  
前記ファイルは、前記ファイル名とともに前記サーバへ送信され、前記サーバに保存されることを特徴とする請求項 1 に記載の情報処理装置。

【請求項 8】

前記連結文字列として連結される前記複数の文字列は、前記ファイル名に用いるように登録されている複数の項目のそれぞれに対応する文字列であって、前記複数の項目の登録されている順番に基づいて連結されることを特徴とする請求項 1 に記載の情報処理装置。

30

【請求項 9】

前記複数の項目は、前記スキャン画像から認識された複数の文字列の、前記スキャン画像内での位置に関する項目であることを特徴とする請求項 8 に記載の情報処理装置。

【請求項 10】

前記ファイルは、前記スキャン画像が予め設定されたファイル形式に従ってファイル化されることにより生成されることを特徴とする請求項 1 に記載の情報処理装置。

【請求項 11】

情報処理装置の制御方法であって、  
文書をスキャンして得られたスキャン画像を含むファイルのファイル名を設定する第 1 の UI 画面であって、前記スキャン画像から認識された複数の文字列が連結された連結文字列を含む第 1 の UI 画面を表示するように制御する第 1 の表示制御工程と、  
前記第 1 の UI 画面上での前記複数の文字列のうちの第 1 の文字列に関するユーザからの指示に応じて、前記複数の文字列のうちの当該第 1 の文字列以外の文字列の何れもが連結されていない当該第 1 の文字列を編集可能に表示する第 2 の UI 画面を表示するように制御する第 2 の表示制御工程と、  
前記第 2 の表示制御工程により編集可能に表示された前記第 1 の文字列への編集をユーザから受付ける受付工程と、  
を含み、  
前記受付工程において受け付けた前記第 1 の文字列への編集に基づいて編集された連結

40

50

文字列が、前記ファイルのファイル名として設定される、  
ことを特徴とする制御方法。

【請求項 1 2】

コンピュータに、

文書をスキャンして得られたスキャン画像を含むファイルのファイル名を設定する第 1 の UI 画面であって、前記スキャン画像から認識された複数の文字列が連結された連結文字列を含む第 1 の UI 画面を表示するように制御する第 1 の表示制御手順と、

前記第 1 の UI 画面上での前記複数の文字列のうちの第 1 の文字列に関するユーザからの指示に応じて、前記複数の文字列のうちの当該第 1 の文字列以外の文字列の何れもが連結されていない当該第 1 の文字列を編集可能に表示する第 2 の UI 画面を表示するように制御する第 2 の表示制御手順と、

10

前記第 2 の表示制御手順において編集可能に表示された前記第 1 の文字列への編集をユーザから受付ける受付手順と、

を実行させるためのプログラムであって、

前記受付手順において受け付けた前記第 1 の文字列への編集に基づいて編集された連結文字列が、前記ファイルのファイル名として設定される、  
ことを特徴とするプログラム。

【請求項 1 3】

前記第 1 の UI 画面に含まれる連結文字列は、前記複数の文字列がユーザにより選択された種類の区切り文字を挟んで連結された連結文字列であり、

20

前記第 1 の文字列は、前記連結文字列を構成する前記複数の文字列のうちの先頭の文字列であって、

前記第 2 の表示制御手順においては、前記指示に応じて、前記複数の文字列のうちの前記第 1 の文字列以外の文字列および前記区切り文字の何れもが連結されていない前記第 1 の文字列を編集可能に表示する前記第 2 の UI 画面を表示するように制御する、

ことを特徴とする請求項 1 2 に記載のプログラム。

【請求項 1 4】

前記複数の文字列は、前記スキャン画像から認識された 2 つの文字列であり、

前記第 2 の表示制御手順においては、前記 2 つの文字列のうちの前記第 1 の文字列に関するユーザからの指示に応じて、前記 2 つの文字列のうちの当該第 1 の文字列以外の文字列が連結されていない当該第 1 の文字列を編集可能に表示する前記第 2 の UI 画面を表示するように制御する、

30

ことを特徴とする請求項 1 2 に記載のプログラム。

【請求項 1 5】

前記受付手順において受け付けた前記第 1 の文字列への編集に基づいて編集された連結文字列は、前記第 1 の文字列への編集を受け付けた後に前記第 1 の UI 画面上に表示されることを特徴とする請求項 1 2 に記載のプログラム。

【請求項 1 6】

前記第 1 の文字列は、前記指示に応じて、前記第 1 の文字列を編集するためのソフトキーボードとともに前記第 2 の UI 画面上に表示されることを特徴とする請求項 1 2 に記載のプログラム。

40

【請求項 1 7】

前記スキャン画像は、スキャナで前記文書をスキャンして得られることを特徴とする請求項 1 2 に記載のプログラム。

【請求項 1 8】

前記ファイルは、前記ファイル名とともにサーバへ送信され、前記サーバに保存されることを特徴とする請求項 1 2 に記載のプログラム。

【請求項 1 9】

前記連結文字列として連結される前記複数の文字列は、前記ファイル名に用いるように登録されている複数の項目のそれぞれに対応する文字列であって、前記複数の項目の登録

50

されている順番に基づいて連結されることを特徴とする請求項 1 2 に記載のプログラム。

【請求項 2 0】

前記複数の項目は、前記スキャン画像から認識された複数の文字列の、前記スキャン画像内での位置に関する項目であることを特徴とする請求項 1 9 に記載のプログラム。

【請求項 2 1】

前記ファイルは、前記スキャン画像が予め設定されたファイル形式に従ってファイル化されることにより生成されることを特徴とする請求項 1 2 に記載のプログラム。

【発明の詳細な説明】

【技術分野】

【0 0 0 1】

本発明は、タッチパネルを用いた文字入力技術に関する。

【背景技術】

【0 0 0 2】

従来より、文書の管理手法として、文書をスキャナで読み取って得られたスキャン画像を所定フォーマットのファイルに変換し、ネットワーク上のストレージサーバに送信して保存する手法が広く利用されている。そして、スキャン画像に対してファイル名を簡単に設定する手法として、プレビュー表示されているスキャン画像上の文字領域をユーザにタッチして選択させる手法がある。特許文献 1 には、文書をスキャンして得られたスキャン画像が事前に定義した画像特徴と一致する場合、該文書内の特定位置に記載される文字列を用いてファイル名を生成して表示し、当該ファイル名へのユーザによる編集を受け付ける技術が開示されている。

【先行技術文献】

【特許文献】

【0 0 0 3】

【文献】特開 2 0 1 8 - 1 2 4 6 5 6 号公報

【発明の概要】

【発明が解決しようとする課題】

【0 0 0 4】

上記特許文献 1 の技術では、文書の特定位置から得られた複数の文字列が結合されて一続きのファイル名の状態でユーザに対して提示（UI 画面上に表示）される。そのため、ファイル名の一部である文字列毎に編集を行いたいユーザは、提示されたファイル名全体の中から、編集対象となる文字列の位置と範囲を指定してから編集する必要がある。つまり、提示されたファイル名に対し部分的な編集を行うには多くの手間を要することになっていた。

【0 0 0 5】

本発明は、上記課題に対応するべくなされたものである。すなわち、UI 画面上に表示中の未確定なファイル名等に対する部分的な編集を、ユーザが容易に行え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】

【0 0 0 6】

本開示に係る情報処理装置は、文書をスキャンして得られたスキャン画像を含むファイルのファイル名を設定する第 1 の UI 画面であって、前記スキャン画像から認識された複数の文字列が連結された連結文字列を含む第 1 の UI 画面を表示するように制御する第 1 の表示制御手段と、前記第 1 の UI 画面上での前記複数の文字列のうちの第 1 の文字列に関するユーザからの指示に応じて、前記複数の文字列のうちの当該第 1 の文字列以外の文字列の何れもが連結されていない当該第 1 の文字列を編集可能に表示する第 2 の UI 画面を表示するように制御する第 2 の表示制御手段と、前記第 2 の表示制御手段により編集可能に表示された前記第 1 の文字列への編集をユーザから受け付ける受付手段と、を有し、前記受付手段で受け付けた前記第 1 の文字列への編集に基づいて編集された連結文字列が、前記ファイルのファイル名として設定される、ことを特徴とする。

## 【発明の効果】

## 【 0 0 0 7 】

本開示の技術によれば、UI画面上に表示中の未確定なファイル名等に対する部分的な編集が容易にできるようになる。これにより、ファイル名の編集作業におけるユーザの操作効率が改善される。

## 【図面の簡単な説明】

## 【 0 0 0 8 】

【図 1】画像処理システムの全体構成を示す図

【図 2】MFPのハードウェア構成を示すブロック図

【図 3】MFP連携サーバ及びストレージサーバのハードウェア構成を示すブロック図

10

【図 4】画像処理システムのソフトウェア構成を示すブロック図

【図 5】画像処理システム全体の処理の流れを示すシーケンス図

【図 6】メイン画面の一例を示す図

【図 7】ログイン画面の一例を示す図

【図 8】スキャン設定画面の一例を示す図

【図 9】(a)はリクエストIDの一例を示す図、(b)及び(c)は処理状況の問合せに対するレスポンスの一例を示す図

【図 10】ファイル名設定画面の一例を示す図

【図 11】学習リクエストの一例を示す図

【図 12】MFPにおける処理の流れを示すフローチャート

20

【図 13】スキャン画像の一例を示す図

【図 14】ファイル名条件設定画面の一例を示す図

【図 15】文字認識結果の一例を示す図

【図 16】類似帳票判定結果の一例を示す図

【図 17】スキャン画像の一例を示す図

【図 18】MFPでのファイル名設定処理の流れを示すフローチャート

【図 19】MFPでのファイル名設定処理の流れを示すフローチャート

【図 20】プレビュー領域の初期表示部分を示す図

【図 21】ファイル名設定画面の一例を示す図

【図 22】ソフトキーボードの一例を示す図

30

【図 23】操作ガイダンスが表示されたファイル名設定画面の一例を示す図

【図 24】文字領域に対してタッチ操作がなされたときのファイル名設定画面の一例を示す図

【図 25】初期変倍率を調整した場合の、プレビュー領域の初期表示部分を示す図

## 【発明を実施するための形態】

## 【 0 0 0 9 】

以下、本発明を実施するための形態について図面を用いて説明する。なお、以下の実施の形態は特許請求の範囲に係る発明を限定するものでなく、また実施の形態で説明されている特徴の組み合わせの全てが発明の解決手段に必須のものとは限らない。

## 【 0 0 1 0 】

40

## [実施形態 1]

## &lt;システム構成&gt;

図 1 は、本実施形態に係る、画像処理システムの全体構成を示す図である。画像処理システムは、MFP (Multifunction Peripheral) 110 と、インターネット上でクラウドサービスを提供するサーバ装置 120 及び 130 とを含む。MFP 110 は、インターネットを介してサーバ装置 120 及び 130 と通信可能に接続されている。

## 【 0 0 1 1 】

MFP 110 は、スキャン機能を有する情報処理装置の一例である。MFP 110 は、スキャン機能に加え印刷機能やBOX保存機能といった複数の機能を有する複合機である。サーバ装置 120 及び 130 は、共にクラウドサービスを提供する情報処理装置の一例

50

である。本実施形態のサーバ装置 120 は、MFP 110 から受け取ったスキャン画像に対し画像解析を行ったり、別のサービスを提供するサーバ装置 130 に対し MFP 110 からのリクエストを転送したりするクラウドサービスを提供する。以下、サーバ装置 120 が提供するクラウドサービスを「MFP 連携サービス」と呼ぶこととする。サーバ装置 130 は、インターネットを介して送られてきたファイルを保存したり、モバイル端末（不図示）などのウェブブラウザからの要求に応じて保存ファイルを提供したりするクラウドサービス（以下、「ストレージサービス」と呼ぶ）を提供する。本実施形態では、MFP 連携サービスを提供するサーバ装置 120 を「MFP 連携サーバ」と呼び、ストレージサービスを提供するサーバ装置 130 を「ストレージサーバ」と呼ぶこととする。

#### 【0012】

図 1 に示す画像処理システム 100 の構成は一例であって、これに限定されない。例えば、MFP 連携サーバ 120 の機能を MFP 110 が兼ね備えていてもよい。また、MFP 連携サーバ 120 はインターネット上ではなく LAN (Local Area Network) 経由で MFP 110 と接続されていてもよい。また、ストレージサーバ 130 を、メール配信サービスを行うメールサーバに置き換えて、文書のスキャン画像をメールに添付し送信する場面に適用してもよい。

#### 【0013】

##### < MFP のハードウェア構成 >

図 2 は、MFP 110 のハードウェア構成を示すブロック図である。MFP 110 は、制御部 210、操作部 220、プリンタ部 221、スキャナ部 222、モデム 223 で構成される。制御部 210 は、以下の各部 211 ~ 219 で構成され、MFP 110 全体の動作を制御する。CPU 211 は、ROM 212 に記憶された様々な制御プログラム（後述のソフトウェア構成図で示す各種機能に対応するプログラム）を読み出して実行する。RAM 213 は、CPU 211 の主メモリ、ワークエリア等の一時記憶領域として用いられる。なお、本実施例では 1 つの CPU 211 が 1 つのメモリ（RAM 213 または HDD 214）を用いて後述のフローチャートに示す各処理を実行するものとするが、これに限定されない。例えば、複数の CPU や複数の RAM または HDD を協働させて各処理を実行してもよい。HDD 214 は、画像データや各種プログラムを記憶する大容量記憶部である。操作部 I/F 215 は、操作部 220 と制御部 210 とを接続するインタフェースである。操作部 220 には、タッチパネルやキーボードなどが備えられており、ユーザによる操作 / 入力 / 指示を受け付ける。なお、タッチパネルへのタッチ操作には、人の指による操作やタッチペンによる操作が含まれる。プリンタ I/F 216 は、プリンタ部 221 と制御部 210 とを接続するインタフェースである。印刷用の画像データはプリンタ I/F 216 を介して制御部 210 からプリンタ部 221 へ転送され、紙等の記録媒体上に印刷される。スキャナ I/F 217 は、スキャナ部 222 と制御部 210 とを接続するインタフェースである。スキャナ部 222 は、不図示の原稿台や ADF (Auto Document Feeder) にセットされた原稿を光学的に読み取ってスキャン画像データを生成し、スキャナ I/F 217 を介して制御部 210 に入力する。スキャナ部 222 で生成されたスキャン画像データは、プリンタ部 221 にて印刷したり（コピー出力）、HDD 214 に保存したり、LAN を介して MFP 連携サーバ 120 等の外部装置にファイル送信したりすることができる。モデム I/F 218 は、モデム 223 と制御部 210 とを接続するインタフェースである。モデム 223 は、PSTN 上のファクシミリ装置（不図示）との間で画像データをファクシミリ通信する。ネットワーク I/F 219 は、制御部 210 (MFP 110) を LAN に接続するインタフェースである。MFP 110 は、ネットワーク I/F 219 を用いて、スキャン画像データを MFP 連携サーバ 120 に送信したり、MFP 連携サーバ 120 から各種データを受信したりする。以上説明した MFP 110 のハードウェア構成は一例であり、必要に応じてその他の構成を備えるものであってもよいし、一部の構成を有していなくてもよい。

#### 【0014】

##### < サーバ装置のハードウェア構成 >

10

20

30

40

50

図3は、MFP連携サーバ120/ストレージサーバ130のハードウェア構成を示すブロック図である。MFP連携サーバ120とストレージサーバ130は共通のハードウェア構成を有し、CPU311、ROM312、RAM313、HDD314及びネットワークI/F315で構成される。CPU311は、ROM312に記憶された制御プログラムを読み出して各種処理を実行することで、全体の動作を制御する。RAM313は、CPU311の主メモリ、ワークエリア等の一時記憶領域として用いられる。HDD314は、画像データや各種プログラムを記憶する大容量記憶部である。ネットワークI/F315は、制御部310をインターネットに接続するインタフェースである。MFP連携サーバ120及びストレージサーバ130は、ネットワークI/F315を介して他の装置(MFP110など)から様々な処理のリクエストを受け、当該リクエストに応じた処理結果を返す。

10

#### 【0015】

<画像処理システムのソフトウェア構成>

図4は、本実施形態に係る、画像処理システム100のソフトウェア構成を示すブロック図である。以下、画像処理システム100を構成するMFP110及び、MFP連携サーバ120及びストレージサーバ130それぞれの役割に対応したソフトウェア構成を、順に説明する。なお、以下では、各装置が有する諸機能のうち、文書をスキャンしてファイル化し、ストレージサーバ130に保存を行うまでの処理に関わる機能に絞って説明を行うものとする。

#### 【0016】

20

MFPのソフトウェア構成

MFP110の機能モジュールは、ネイティブ機能モジュール410とアドイショナル機能モジュール420の2つに大別される。ネイティブ機能モジュール410はMFP110に標準的に備えられたアプリケーションであるのに対し、アドイショナル機能モジュール420はMFP110に追加的にインストールされたアプリケーションである。アドイショナル機能モジュール420は、Java(登録商標)をベースとしたアプリケーションであり、MFP110への機能追加を容易に実現できる。なお、MFP110には図示しない他の追加アプリケーションがインストールされていてもよい。

#### 【0017】

ネイティブ機能モジュール410は、スキャン実行部411およびスキャン画像管理部412を有する。また、アドイショナル機能モジュール420は、表示制御部421、スキャン制御部422、連携サービスリクエスト部423、画像処理部424を有する。

30

#### 【0018】

表示制御部421は、操作部220のタッチパネルに、各種のユーザ操作を受け付けるためのユーザインタフェース画面(UI画面)を表示する。各種のユーザ操作には、例えば、MFP連携サーバ120へアクセスするためのログイン認証情報の入力、スキャン設定、スキャンの開始指示、ファイル名設定、ファイルの保存指示などがある。

#### 【0019】

スキャン制御部422は、UI画面でなされたユーザ操作(例えば「スキャン開始」ボタンの押下)に応じて、スキャン設定の情報と共にスキャン実行部411に対しスキャン処理の実行を指示する。スキャン実行部411は、スキャン制御部422からのスキャン処理の実行指示に従い、スキャナI/F217を介してスキャナ部240に文書の読み取り動作を実行させ、スキャン画像データを生成する。生成したスキャン画像データは、スキャン画像管理部412によってHDD214に保存される。この際、保存されたスキャン画像データを一意に示すスキャン画像識別子の情報が、スキャン制御部422へ通知される。スキャン画像識別子は、MFP110においてスキャンした画像をユニークに識別するための番号や記号、アルファベットなどである。スキャン制御部422は、例えばファイル化する対象のスキャン画像データを上記のスキャン画像識別子を使ってスキャン画像管理部412から取得する。そして、ファイル化のために必要な処理のリクエストをMFP連携サーバ120に対して行うよう、連携サービスリクエスト部423に対して指示

40

50

する。

#### 【 0 0 2 0 】

連携サービスリクエスト部 4 2 3 は、M F P 連携サーバ 1 2 0 に対して各種処理のリクエストを行ったり、そのレスポンスを受け取ったりする。各種処理には、例えば、ログイン認証、スキャン画像の解析、スキャン画像データの送信などが含まれる。M F P 連携サーバ 1 2 0 とのやり取りは R E S T や S O A P などの通信プロトコルを使用される。

#### 【 0 0 2 1 】

画像処理部 4 2 4 は、スキャン画像データに対し所定の画像処理を行って、表示制御部 4 2 1 が表示する U I 画面で用いられる画像を生成する。所定の画像処理の詳細については後述する。

#### 【 0 0 2 2 】

##### サーバ装置のソフトウェア構成

まず、M F P 連携サーバ 1 2 0 のソフトウェア構成について説明する。M F P 連携サーバ 1 2 0 は、リクエスト制御部 4 3 1、画像処理部 4 3 2、ストレージサーバアクセス部 4 3 3、データ管理部 4 3 4、表示制御部 4 3 5 を有する。リクエスト制御部 4 3 1 は、外部装置からのリクエストを受信できる状態で待機しており、受信したリクエスト内容に応じて、画像処理部 4 3 2、ストレージサーバアクセス部 4 3 3、データ管理部 4 3 4 に対し所定の処理の実行を指示する。画像処理部 4 3 2 は、M F P 1 1 0 から送られてくるスキャン画像データに対して、文字領域の検出処理、文字認識処理、類似文書の判定処理といった画像解析処理の他、回転や傾き補正といった画像加工処理を行う。ストレージサーバアクセス部 4 3 3 は、ストレージサーバ 1 3 0 に対する処理のリクエストを行う。クラウドサービスでは、R E S T や S O A P などのプロトコルを用いてストレージサーバにファイルを保存したり、保存したファイルを取得したりするための様々なインタフェースを公開している。ストレージサーバアクセス部 4 3 3 は、公開されたインタフェースを使用して、ストレージサーバ 1 3 0 に対するリクエストを行う。データ管理部 4 3 4 は、M F P 連携サーバ 1 2 0 で管理するユーザ情報や各種設定データ等を保持・管理する。表示制御部 4 3 5 は、インターネット経由で接続された P C やモバイル端末（いずれも不図示）上で動作しているウェブブラウザからのリクエストを受けて、画面表示に必要な画面構成情報（H T M L、C S S 等）を返す。ユーザは、ウェブブラウザで表示される画面経由で、登録されているユーザ情報を確認したり、スキャン設定を変更したりできる。

#### 【 0 0 2 3 】

次に、ストレージサーバ 1 3 0 のソフトウェア構成について説明する。ストレージサーバ 1 3 0 は、リクエスト制御部 4 4 1、ファイル管理部 4 4 2、表示制御部 4 4 3 を有する。リクエスト制御部 4 4 1 は、外部装置からのリクエストを受信できる状態で待機しており、本実施形態においては M F P 連携サーバ 1 2 0 からのリクエストに応じて、受信したファイルの保存や保存ファイルの読み出しをファイル管理部 4 4 2 に指示する。そして、リクエストに応じたレスポンスを M F P 連携サーバ 1 2 0 に返す。表示制御部 4 4 3 は、インターネット経由で接続された P C やモバイル端末（いずれも不図示）上で動作しているウェブブラウザからのリクエストを受けて、画面表示に必要な画面構成情報（H T M L、C S S 等）を返す。ユーザは、ウェブブラウザで表示される画面経由で、保存ファイルを確認したり取得したりすることができる。

#### 【 0 0 2 4 】

##### < 画像処理システム全体の処理の流れ >

図 5 は、M F P 1 1 0 で文書をスキャンし、得られたスキャン画像をファイル化してストレージサーバに保存する際の、装置間の処理の流れを示すシーケンス図である。図 6 は M F P 1 1 0 の起動時に表示されるメインメニューの U I 画面（以下、「メイン画面」と表記）の一例を示す図である。文書をスキャンしてファイル化し、クラウドストレージサービスの利用に必要な専用のアプリケーションを M F P 1 1 0 にインストールすることで、メイン画面 6 0 0 上に「スキャンしてクラウドストレージに保存」ボタン 6 0 1 が表示されるようになる。そして、ユーザがメイン画面 6 0 0 内に表示されたメニューボタンの

10

20

30

40

50

中から「スキャンしてクラウドストレージに保存」ボタン601を押下すると、図5のシーケンス図で示される一連の処理が開始する。以下、図5のシーケンス図に沿って、装置間のやり取りを時系列に説明する。なお、以下の説明において記号「S」はステップを表す。

【0025】

MFP110は、MFP連携サーバ120にアクセスするためのログイン認証の情報を入力するUI画面（以下、「ログイン画面」と表記）を操作部220に表示する（S501）。図7にログイン画面の一例を示す。ユーザが、予め登録されているユーザIDとパスワードを、ログイン画面700上の入力欄702及び703にそれぞれ入力し「ログイン」ボタン701を押下すると、ログイン認証のリクエストがMFP連携サーバ120に送信される（S502）。

10

【0026】

ログイン認証のリクエストを受けたMFP連携サーバ120は、当該リクエストに含まれるユーザIDとパスワードを用いて認証処理を行う（S503）。認証処理の結果、正規のユーザであることが確認されれば、MFP連携サーバ120は、アクセストークンをMFP110に返す。以後、MFP110からMFP連携サーバ120に対して行う各種リクエストの際にこのアクセストークンを一緒に送ることで、ログイン中のユーザが特定される。本実施形態では、MFP連携サーバ120へのログインの完了によって、ストレージサーバ130へのログインも同時に完了するものとする。このためにユーザは、インターネット上のPC（不図示）のウェブブラウザ等を介して、MFP連携サービスを利用するためのユーザIDとストレージサービスを利用するためのユーザIDとの紐づけを予め行っておく。これにより、MFP連携サーバ120へのログイン認証に成功すれば同時にストレージサーバ130へのログイン認証も完了し、ストレージサーバ130にログインするための操作を省略できる。そして、MFP連携サーバ120においては、自装置にログインしたユーザからのストレージサービスに関するリクエストにも対応可能となる。なお、ログイン認証の方法は一般的に公知な手法（Basic認証、Digest認証、OAuthを用いた認可等）を用いて行えばよい。

20

【0027】

ログインが完了すると、MFP110は、スキャン設定用のUI画面（以下、「スキャン設定画面」と表記）が操作部220に表示される（S507）。図8にスキャン設定画面の一例を示す。スキャン設定画面800には、「スキャン開始」ボタン801、カラー設定欄802、解像度設定欄803が存在する。「スキャン開始」ボタン801は、原稿台にセットした文書（本実施形態では見積書や請求書といった帳票を想定）に対するスキャン処理の開始を指示するためのボタンである。カラー設定欄802では、スキャン時のカラーモードを設定する。例えばフルカラーやモノクロといった選択肢の中から指定できるようになっている。解像度設定欄803では、スキャン時の解像度を設定する。例えば600dpiや1200dpiといった選択肢の中から指定できるようになっている。なお、カラーモードと解像度は設定項目の一例であって、これらすべてが存在しなくてもよいし、これら以外の設定項目が存在してもよい。また、カラーモードや解像度に関する選択肢を、ストレージサービスの要求する設定値のみに限定したりしてもよい。ログインユーザは、このようなスキャン設定画面800を介してスキャン処理についての詳細な条件設定を行なう。スキャン設定を終えたログインユーザが、MFP110の原稿台にスキャン対象の文書をセットし、「スキャン開始」ボタン801を押下するとスキャンが実行される（S505）。これにより、紙文書を電子化した画像データが生成される。スキャンの完了後、MFP110は、スキャンによって得られた画像データを、その解析リクエストと共にMFP連携サーバ120に送信する（S506）。

30

40

【0028】

スキャン画像の解析リクエストを受けたMFP連携サーバ120では、リクエスト制御部431が画像処理部432に対し、画像解析処理の実行を指示する（S507）。その際、リクエスト制御部431は、受信した解析リクエストを一意に特定可能なリクエスト

50

ＩＤをＭＦＰ１１０に返す。図９（ａ）にリクエストＩＤの一例を示す。一方、解析処理の実行指示を受けた画像処理部４３２は、スキャン画像に対する解析処理を実行する（Ｓ５０８）。この画像解析処理では、まずスキャン画像内に存在する文字領域を検出する処理を行う。文字領域の検出は、例えば、ある閾値で２値化を行った画像から文字と推測される矩形領域を抽出する方法等、既知の方法を適用すればよい。次に、文字領域検出処理によって見つかった各文字領域に対して、文字認識処理（ＯＣＲ：Optical Character Recognition）を行う。そして、今回のスキャン対象の帳票が、過去のスキャン対象の帳票と類似しているかどうかを判定する処理（類似文書判定処理）を行う。この類似帳票の判定では、スキャン画像内に存在する各文字領域が当該スキャン画像のどこに位置するかを示す配置情報を用いる。具体的には、過去のスキャン画像についての配置情報と今回のスキャン画像についての配置情報とを比較し、文字領域の配置が類似しているかどうかを判定する。これは、文字領域の配置が類似していれば、それらは同一のフォーマットを用いて作成された帳票であると推測できることに基づいている。この類似帳票の判定で使用する過去のスキャン画像の配置情報は、後述の「帳票学習処理」によって蓄積される。これら画像解析処理によって得られた結果は、リクエスト制御部４３１に渡される。なお、本実施形態では、文字領域の配置の類似度合いのみに基づき帳票の類否を判定しているが、例えば、ＯＣＲ結果から帳票の種別（見積書、請求書など）を特定し、得られた種別の情報を加味して類否を判定してもよい。

#### 【００２９】

上記画像解析処理が行われている間、ＭＦＰ１１０は、上述のリクエストＩＤを使用して、ＭＦＰ連携サーバ１２０に対して定期的（例えば数百ミリ秒から数ミリ秒程度毎）に処理状況の問合せを行う（Ｓ５０９～Ｓ５０９'）。この問合せは、ＭＦＰ連携サーバ１２０からの画像解析処理の完了レスポンス（Ｓ５１０）が取得できるまで繰り返し実行される。ＭＦＰ連携サーバ１２０は、処理状況の問合せを受けると、リクエストＩＤに対応する画像解析処理の進行状況を確認し、完了していない場合は処理中を表すレスポンス（図９（ｂ）を参照）を返す。また、完了していた場合は完了を表すレスポンス（図９（ｃ）を参照）を返す。このレスポンスの“status”には現在の処理状況を示す文字列、具体的には、ＭＦＰ連携サーバ１２０で処理が行われている最中である場合には“processing”が入り、処理が完了している場合には“completed”が入る。なお、処理が失敗した場合の“failed”など、他のステータスを表す文字列が入ることもある。図９（ｃ）に示すとおり、処理完了時のレスポンスには、ステータス情報に加え、スキャン画像の解析結果に関する情報などを含む。

#### 【００３０】

処理完了レスポンスを受信した後、ＭＦＰ１１０は、当該レスポンスに含まれる、画像解析結果の格納先を示すＵＲＬを用いて、画像解析処理の結果を、ＭＦＰ連携サーバ１２０に対してリクエストする（Ｓ５１１）。この場合のＵＲＬには、“ocrResultUrl”と“matchingResultUrl”がある。リクエストを受けてＭＦＰ連携サーバ１２０のリクエスト制御部４３１は、画像解析処理の結果情報を返す。

#### 【００３１】

そして、ＭＦＰ１１０は、取得した結果情報を使用して、ファイル名を設定するためのＵＩ画面（以下、「ファイル名設定画面」と表記）を表示する（Ｓ５１２）。図１０に、ファイル名設定画面の一例を示す。ユーザが、ファイル名を設定して送信ボタン１００７を押下すると、ＭＦＰ１１０は、まず、当該ファイル名を設定する際にユーザが行った入力操作に関する情報（設定時入力情報）を含んだ学習リクエストをＭＦＰ連携サーバ１２０に送信する（Ｓ５１３）。図１１に、学習リクエストの一例を示す。“learningContent”において学習内容が特定され、その中には、ファイル名に使用された文字領域に関する“rectInfoArray”とメタデータに関する“metadataArray”がある。“rectInfoArray”には、ファイル名設定時に使用した文字領域の座標情報が入る。“metadataArray”には、ファイル名に使用した文字列に対応する文字領域の情報と、当該文字列に付随する区切り文字（セパレータ）の情報が入る。図１１の例では、先頭が“fileRegion0”の文字領域の文字

列、その次が区切り文字、そして末尾が“fileRegion1”の文字領域の文字列、の順に並ぶファイル名構成であることを示している。

【0032】

学習リクエストを受信したMFP連携サーバ120では、リクエスト制御部431が、画像処理部432に対し帳票学習処理の実行を指示する(S514)。この指示を受けて画像処理部432は、スキャン画像に存在する各文字領域の配置情報と、S513で受け取った学習リクエストに含まれるファイル名設定時の入力情報(ファイル名の構成情報)を保存する(S515)。

【0033】

その後、MFP110は、スキャン画像のデータとそのファイル化の際に設定するファイル名等の情報と共に、当該スキャン画像の保存リクエストをMFP連携サーバ120に送信する(S516)。保存リクエストを受信したMFP連携サーバ120では、リクエスト制御部431が、ファイル生成処理を開始すると共に保存リクエストを正常に受けたことを示すレスポンスをMFP110に返す。そのレスポンスを受け取ったMFP110は、処理を終了し、スキャン設定画面の表示状態(S504)に戻る。一方、MFP連携サーバ120では、事前に登録されたスキャン設定からファイルフォーマットの情報を取得し、当該ファイルフォーマットに従ってスキャン画像をファイル化する(S517)。この際、生成されたファイルには保存リクエストにて指定されたファイル名が付されることになる。こうして生成されたスキャン画像ファイルは、ストレージサーバ130に送信され、保存される(S518)。

【0034】

以上が、画像処理システム全体の処理の流れである。

【0035】

<MFPの処理詳細>

続いて、MFP110における動作に着目して、上述の画像処理システムでストレージサービスを利用する場合の処理について詳しく説明する。図12は、MFP110における処理の流れを示すフローチャートである。この一連の処理は、制御部210において、CPU211がHDD214に記憶された制御プログラムを実行することにより実現され、前述のメイン画面600での「スキャンしてクラウドストレージに保存」ボタン601の押下に応答して開始される。以下、図12に示すフローチャートに沿って詳しく説明する。なお、各処理の冒頭における記号「S」はステップを意味する。

【0036】

S1201において、表示制御部421は、前述のログイン画面700を表示する。ログイン画面700上でユーザ名とパスワードが入力欄702及び703にそれぞれ入力され、「ログイン」ボタン701の押下が検知されると、S1202に進む。

【0037】

S1202において、連携サービスリクエスト部423は、ログイン認証リクエストをMFP連携サーバ120に送信する。MFP連携サーバ120でのログイン認証によって正規のユーザであることが確認されると、連携サービスリクエスト部423は、アクセストークンをMFP連携サーバ120から受信する。

【0038】

S1203において、アクセストークンを受信したMFP110では、表示制御部421が、前述のスキャン設定画面800を操作部220に表示する。ログインユーザによって原稿台に文書がセットされ、「スキャン開始」ボタン801の押下が検知されると、表示制御部421は、スキャン制御部422に対してその旨を通知する。

【0039】

S1204において、上記通知を受けたスキャン制御部422は、スキャン実行部411に対してスキャン処理の実行を指示する。当該指示を受けてスキャン実行部411は、原稿台に置かれた文書をスキャンする。本フローの説明においては、スキャン対象として帳票(より具体的には見積書)を例に説明を行うものとする。これにより、図13に示す

10

20

30

40

50

ような見積書のスキャン画像が得られる。スキャンによって生成されたスキャン画像データは、画像データ保存部 4 1 2 に保存され、当該スキャン画像を特定可能な識別子がスキャン制御部 4 2 1 に通知される。この際、スキャンの実行中であることを示すメッセージ画面（不図示）が操作部 2 2 0 上に表示されるようにしてもよい。

#### 【 0 0 4 0 】

次に、S 1 2 0 5 において、連携サービスリクエスト部 4 2 3 は、スキャン制御部 4 2 1 を介してスキャン画像データを取得し、M F P 連携サーバ 1 2 0 にその解析リクエストと共に送信する。当該解析リクエストに基づき、M F P 連携サーバ 1 2 0 では、受信したスキャン画像データに対して、前述の画像解析処理が実行される。その際、M F P 連携サーバ 1 2 0 のリクエスト制御部 4 3 1 からリクエスト I D が返される。

10

#### 【 0 0 4 1 】

S 1 2 0 6 において、リクエスト I D を取得した連携サービスリクエスト部 4 2 3 は、処理ステータスの取得リクエストをリクエスト I D と共に M F P 連携サーバ 1 2 0 に送信する。当該取得リクエストに基づき、M F P 連携サーバ 1 2 0 は、リクエスト I D で特定される画像解析処理の進行状況に応じたレスポンスを M F P 1 1 0 に返す。

#### 【 0 0 4 2 】

S 1 2 0 7 において、連携サービスリクエスト部 4 2 3 は、受け取ったレスポンスにおけるステータスが処理中を示す“processing”であった場合は S 1 2 0 6 に戻る。そして、所定時間の経過を待って、再び処理ステータスの取得リクエストを M F P 連携サーバ 1 2 0 に送信する。受け取ったレスポンスにおけるステータスが完了を示す“completed”であった場合は、S 1 2 0 8 に進む。ここで、本実施形態における処理完了時のレスポンスに含まれる画像解析結果の情報について、前述の図 9（c）を参照しつつ説明する。画像解析結果の情報には、“ocrResultUrl”、“matchingResultUrl”、“formKeys”の 3 種類の情報がある。“ocrResultUrl”は、画像解析処理のうち O C R 処理の結果を取得するための U R L の情報である。“matchingResultUrl”は、画像解析処理のうち類似帳票判定処理の結果を取得するための U R L の情報である。“formKeys”は、ファイル名、メタデータ、フォルダパスといったプロパティをスキャン画像に設定するための設定値の情報であり、M F P 連携サーバ 1 2 0 において事前設定されている。“formKeys”の中には以下の情報が含まれる。

- ・ key：スキャン画像に設定する設定値を一意に示す値 30
- ・ keyType：key の設定値の種別を示す値
- ・ value：設定値の初期値（例えば、スキャン画像であることを示す“scan”）
- ・ type：設定値に入力する値の種類を示す値  
（例えば、文字列なら“string”、数値なら“number”、日付なら“date”など）
- ・ displayName：M F P のタッチパネル上に設定画面を表示するときの表示名
- ・ required：その設定値を入力することが必須かどうかを示す値
- ・ multiSelect：ファイル名に複数の文字領域の文字列を使用するかどうかを示す値
- ・ separator：複数の文字領域の文字列を使用する場合の文字列間を繋ぐ区切り文字の内容

- ・ autoInput：初期ファイル名の自動入力を行うかどうかを示す値 40

#### 【 0 0 4 3 】

図 9（c）には、スキャン画像にファイル名を設定する場合の“formKeys”が示されている。この図 9（c）の場合を例に、“value”、“displayName”、“required”、“multiSelect”の各値が、次の S 1 2 0 9 で表示されるファイル名設定画面（図 1 0）にどのような影響をあたえるかを説明する。いま、“value”の内容は空である。この場合、ファイル名設定画面 1 0 0 0 のファイル名入力欄 1 0 0 1 には初期値が表示されないことになる。また、“displayName”は「ファイル名」となっている。この場合、図 1 0 のファイル名設定画面 1 0 0 0 のように設定のタイトルやラベルを示す「ファイル名」の文字列が画面上部に（この例では白抜き文字で）表示されることになる。また、“required”は「true」となっている。この場合、ファイル名入力欄 1 0 0 1 に文字列が入っていない空の状態では 50

送信ボタン 1007 を押下することができなくなる。また、“multiSelect”が「true」となっている。この場合、スキャン画像から検出された複数の文字領域を選択することが可能になり、これによって各文字領域に対応した文字列を繋げたファイル名を設定できることになる。これら“formKeys”に含まれる上記の内容は、MFP 連携サーバ 120 の表示制御部 435 が表示するファイル名条件設定画面を介してユーザが設定する。図 14 にファイル名条件設定画面の一例を示す。ファイル名条件設定画面 1400 内のチェックボックス 1401 は、以前にスキャン画像のファイル名設定時に用いた文字領域の情報を使用して自動的にファイル名を入力するか否かをユーザが選択するために用いる。設定欄 1402 には、ファイル名に複数の文字領域に対応する複数の文字列を使用する場合に、文字列間に挿入する区切り文字に何を使用するかをプルダウン表示される選択肢（アンダースコア、ハイフンなど）の中から設定する。ここでは、アンダースコアが選択されている。設定欄 1403 には、ファイル名設定画面（図 10）の初期表示の際に予め入力状態にしておき、ユーザにファイル名の候補として提示する文字列（初期ファイル名）を設定する。このファイル名条件設定画面 1400 で設定した内容はデータ管理部 434 で保持され、処理完了時のレスポンスを作成する際に参照される。

10

#### 【0044】

S1208において、連携サービスリクエスト部 423 は、処理完了時のレスポンスに含まれるURLにアクセスして、画像解析結果を取得する。図9(c)の例では“ocrResultUrl”と“matchingResultUrl”に情報が格納されているので、この2つのURLにアクセスして画像解析結果を取得する。以下の表1は、図13のスキャン画像から検出された文字領域の座標と、各文字領域についての文字認識結果（認識された文字列）を示している。

20

#### 【0045】

30

40

50

【表 1】

| 番号 | 領域の X 座標 | 領域の Y 座標 | 領域の幅 | 領域の高さ | 領域内文字列       |
|----|----------|----------|------|-------|--------------|
| 1  | 1019     | 303      | 489  | 95    | 見積書          |
| 2  | 261      | 446      | 243  | 43    | 〒100-9999    |
| 3  | 1584     | 446      | 262  | 36    | 見積番号:        |
| 4  | 1874     | 443      | 230  | 47    | R12-3456     |
| 5  | 255      | 531      | 798  | 47    | 東京都港区B-B-B   |
| 6  | 1671     | 530      | 165  | 44    | 発行日:         |
| 7  | 1872     | 527      | 259  | 54    | 2017/09/10   |
| 8  | 406      | 626      | 594  | 71    | 品川株式会社       |
| 9  | 1164     | 636      | 112  | 55    | 御中           |
| 10 | 456      | 1186     | 105  | 55    | 品目           |
| 11 | 1102     | 1188     | 100  | 55    | 単価           |
| 12 | 1510     | 1188     | 101  | 55    | 数量           |
| 13 | 2043     | 1187     | 113  | 57    | 金額           |
| 14 | 400      | 1356     | 230  | 45    | GF-1544      |
| 15 | 1023     | 1358     | 180  | 45    | 11,550       |
| 16 | 1562     | 1358     | 27   | 46    | 3            |
| 17 | 2023     | 1358     | 182  | 51    | 34,650       |
| 18 | 398      | 1447     | 230  | 54    | EF24-70      |
| 19 | 1003     | 1448     | 203  | 47    | 134,100      |
| 20 | 1572     | 1448     | 10   | 47    | 1            |
| 21 | 2003     | 1448     | 204  | 47    | 134,100      |
| 22 | 1517     | 2038     | 128  | 63    | 合計           |
| 23 | 1950     | 2042     | 250  | 58    | 168,750      |
| 24 | 972      | 2770     | 581  | 72    | 川崎株式会社       |
| 25 | 1167     | 2925     | 615  | 43    | 神奈川県横浜市2-2-2 |

## 【0046】

そして、図15は、“ocrResultUrl”にアクセスして取得される、図13のスキャン画像についての文字認識結果を示している。“imageWidth”は、解析対象のスキャン画像のX方向（横方向）のピクセル数を示す。“imageHeight”は、解析対象のスキャン画像のY方向（縦方向）のピクセル数を示す。“regions”は、スキャン画像から検出した文字領域の座標情報（“rect”）と、その文字領域に対応する文字列情報（“Text”）が含まれる。“rect”には、検出した文字領域を特定する情報が含まれ、“x”は文字領域の左上隅のX座標、“y”は文字領域の左上隅のY座標、“width”は文字領域のX（幅）方向のピクセル数、“height”は文字領域のY（高さ）方向のピクセル数を示す。“text”は、“rect”で特定される文字領域をOCR処理して得られた文字列の情報を示す。この“rect”と“text”の情報が、解析対象のスキャン画像内で検出された全ての文字領域分だけ含まれる。ただし、図15ではその一部を省略している。

## 【0047】

そして、図16は、“matchingResultUrl”にアクセスして取得される、図13のスキャ

ン画像についての類似帳票判定結果を示している。“matched”には、前述の類似帳票判定処理によって、今回解析対象のスキャン画像と類似しているものが過去のスキャン画像の中から見つかったかどうかを示す値として“true”又は“false”が入る。値が“true”であれば、今回のスキャン対象の帳票が以前にスキャンした帳票と類似していることを示し、“false”であれば類似していないことを表す。“formId”には、類似する帳票が見つかった場合は、過去の類似帳票のスキャン画像を一意に示す値（ID値）が入る。一方、類似する帳票が見つからなかった場合は、今回スキャンを行った帳票のスキャン画像を一意に示す新たに作成されたID値が入る。そして、この新たに作成されたID値は、過去に類似帳票が存在しなかった帳票のスキャン画像についてのファイル名設定時の入力情報（ファイル名の構成情報）を学習させるときに使用される。“matchingScore”には、類似する帳票があった場合に一致度合いを示す値が入る。本実施形態の類似度は、過去のスキャン画像における文字領域の配置情報と今回のスキャン画像における文字領域の配置情報との一致度合いを0～1までの実数値で表したものであり、その値が大きいほど類似した帳票であることを示す。“rectInfoArray”には、過去の類似帳票のスキャン画像に対するファイル名設定時に使用された文字領域に対応する、今回解析対象のスキャン画像における文字領域を示す情報が入る。

#### 【0048】

図16の例の場合は、以前に、見積書をスキャンして得られた図13のスキャン画像に対して「見積書」と「品川株式会社」の2つの文字列を使用してファイル名を設定し、その入力情報の学習を行っている。その後、同一フォーマットで作成した見積書をスキャンして図17に示すスキャン画像を生成し、当該スキャン画像について類似帳票判定処理を行った結果、図13のスキャン画像と類似していると判定されている。そして、図13のスキャン画像に対するファイル名の設定時に使用された文字領域の情報が、自動入力の対象となる文字領域の情報として格納された状態を、図16の例は示している。以下、図16に示す類似帳票判定処理の結果が得られるまでの過程を説明する。

#### 【0049】

前提として、前述の図11の学習リクエストに基づく学習結果が、データ管理部434に保存されているものとする。そして、類似帳票判定処理では、以前のファイル名設定時に用いた「見積書」と「品川株式会社」の文字領域の座標情報を用いて、今回のスキャン画像に含まれる文字領域の中の当該座標情報で示された文字領域とその一部が重なる文字領域が特定される。そして、一部が重なると特定された文字領域の座標情報と当該文字領域の文字列を“rectInfoArray”の“text”に格納される。“key”には、ファイル名の自動入力に使用する文字領域を一意に示す値が入る。“region”には“key”が一意に示す文字領域の座標情報と当該文字領域に対するOCR処理によって得られた文字列が格納される。そして、“rect”には、“key”が一意に示す文字領域を特定する情報が含まれる。この場合において、“x”は文字領域の左上隅のX座標、“y”は文字領域の左上隅のY座標、“width”は文字領域のX（幅）方向のピクセル数、“height”は文字領域のY（高さ）方向のピクセル数を示す。そして、“rect”で特定される文字領域をOCR処理して得られた文字列の情報が“text”に入っている。“metadataArray”には、ファイル名を自動入力する際に使用する文字領域の順番と区切り文字がどこに入るかを示す情報が格納される。ファイル名以外にメタデータなどのプロパティ情報が設定されている場合は、“rectInfoArray”や“metadataArray”において、必要な情報が追加される。“key”には、スキャン画像に設定する設定値を一意に示す値が格納される。“keyType”には、“key”の設定値の種別を示す値が格納される。ファイル名に使用する場合は、“key”には“filename”が入り、“keyType”には“filename”が入る。“value”には、“key”の値に使用する文字領域と区切り文字の情報が格納される。図16の例では、“rectInfoArray”の中の“fileRegion0”の“key”を持つ文字領域の文字列が1番目、区切り文字が2番目、“fileRegion1”の“key”を持つ文字領域の文字列が3番目となっている。そして、この順番でファイル名入力欄1001に自動入力され、初期ファイル名として表示されることになる。

#### 【0050】

S 1 2 0 9において、表示制御部 4 2 1 が、前述のファイル名設定画面 1 0 0 0 を操作部 2 2 0 に表示して、ユーザが、スキャン画像に対するファイル名の設定を行えるようにする。このファイル名設定処理の詳細については後述する。

【 0 0 5 1 】

S 1 2 1 0において、連携サービスリクエスト部 4 2 3 が、S 1 2 0 6 の処理ステータスのリクエストに対するレスポンスを参照して、ファイル名の自動入力を行う設定となっているか否かを判定し、次の処理を決定する。具体的には、前述の図 9 ( c ) で示した処理完了時のレスポンスに含まれる“autoInput”の値を参照し、自動入力を行う設定を意味する“true”の場合はS 1 2 1 1へ進む。一方、自動入力を行わない設定を意味する“false”の場合はS 1 2 1 4へ進む。

10

【 0 0 5 2 】

S 1 2 1 1において、連携サービスリクエスト部 4 2 3 は、今回のスキャン画像に類似するスキャン画像が過去に存在していたかどうかを、S 1 2 0 8 で取得した画像解析結果に基づき判定する。具体的には、前述の図 1 6 で示した類似帳票判定処理の結果に含まれる“matched”の値を参照し、過去のスキャン画像の中に類似したものが存在することを示す“true”の場合はS 1 2 1 2へ進む。一方、類似したものが存在しないことを示す“false”の場合はS 1 2 1 4へ進む。

【 0 0 5 3 】

S 1 2 1 2において、連携サービスリクエスト部 4 2 3 は、S 1 2 0 9 のファイル名設定処理において、ファイル名入力欄 1 0 0 1 に自動入力されていた初期ファイル名に対して、ユーザが変更を加えたかどうかを判定する。この場合の変更には、例えば今回のファイル名に使用する文字領域を、初期ファイル名で使用されている文字領域と異なるものにしたたり、区切り文字を追加したり或いは削除したりすることを含む。そして、この判定は、図 1 6 の類似帳票判定処理の結果に含まれる“metadataArray”の“value”の値と、今回のスキャン画像に対するファイル名設定を行った後のファイル名入力欄 1 0 0 1 への入力情報との比較により行う。比較の結果、差異があれば初期ファイル名に対して変更がなされたと判断してS 1 2 1 3へ進み、一致していれば初期ファイル名に対して変更がなされていないと判断してS 1 2 1 4へ進む。なお、“formKeys”の“value”によって自動入力されていたファイル名の先頭文字列については、ユーザによって削除されていても入力情報が変わったとみなさずにS 1 2 1 4に進んでもよい。

20

30

【 0 0 5 4 】

S 1 2 1 3では、連携サービスリクエスト部 4 2 3 が、今回のスキャン画像に対して設定されたファイル名の構成を表す設定時入力情報の学習リクエスト（前述の図 1 1 を参照）を、MFP連携サーバ 1 2 0 へ送信する。この場合の学習リクエストに含まれる“formId”には、S 1 2 0 8 で取得した画像解析結果の類似判定処理結果に含まれる“formId”の値が使用される。図 1 6 の例では、“formId”の値として“aaaaaaaa-ffff-49ab-acf8-55558888eeee”が入っているので、これが学習リクエストに格納されることになる。また、“learningContent”には、ファイル名設定時に使用した入力情報の内容が格納される。まず、“rectInfoArray”には、ファイル名に使用した文字列が属する文字領域の座標情報が、その文字領域の数分だけ格納される。そして、“metadataArray”には、ファイル名の設定に用いた文字領域と区切り文字の情報が格納される。前述の表 1 を参照して、このときのユーザが行った操作を説明すると、ファイル名設定画面 1 0 0 0 のプレビュー領域 1 0 0 2 において、最初に番号 1 の「見積書」をタッチする。その次に、番号 8 の「品川株式会社」をタッチし、最後に送信ボタン 1 0 0 7 を押下している。学習リクエストを受けたMFP連携サーバ 1 2 0 は、このようなユーザ操作に基づく入力情報を蓄積し、当該蓄積した入力情報を次回以降の画像解析処理から使用することになる。学習リクエストを送信後、当該学習リクエストを正常に処理した旨のレスポンスをMFP連携サーバ 1 2 0 から受け取ると、S 1 2 1 4に進む。

40

【 0 0 5 5 】

S 1 2 1 4では、連携サービスリクエスト部 4 2 3 が、処理対象のスキャン画像データ

50

と S 1 2 0 9 で設定されたファイル名とを対応付けて、M F P 連携サーバ 1 2 0 に送信する。この送信完了によって本処理が終了する。なお、本実施形態ではスキャン画像データとそのファイル名の送信を最後に行っているが、スキャン処理が終わり、スキャン画像管理部 4 1 2 によるスキャン画像データの保存の時点で並行して、M F P 連携サーバ 1 2 0 に送信しておいてもよい。

#### 【 0 0 5 6 】

以上が、M F P 1 1 0 における動作に着目したストレージサービスを利用する場合の処理の流れである。なお、上記の説明では、1 枚の見積書をスキャンして得られた 1 つのスキャン画像をファイル化する際のファイル名設定を行う場面を想定していた。しかし、本実施形態はこれに限定されるものではなく、複数の文書をスキャンして得られた複数のスキャン画像をファイル化する際のファイル名設定を行う場面にも同様に適用可能である。この場合、ファイル名設定画面 1 0 0 0 内にページめくりをするためのボタンを設け、2 ページ目や 3 ページ目のスキャン画像の文字領域からもファイル名の設定を行えるようにしてもよい。

#### 【 0 0 5 7 】

( ファイル名設定処理 )

続いて、前述の S 1 2 0 9 にて表示制御部 4 2 1 が実行するファイル名設定処理について、図 1 8 及び図 1 9 に示すフローチャートに沿って詳しく説明する。

#### 【 0 0 5 8 】

S 1 8 0 1 では、前述のファイル名設定画面 1 0 0 0 のファイル名入力欄 1 0 0 1 に初期状態で表示されるファイル名 ( 初期ファイル名 ) が生成される。初期ファイル名は、以下の手順で生成される。まず、前述の処理完了時のレスポンスを参照し、“formKeys”の“value”に何らかの値が入っていれば、当該値をファイル名の先頭に設定する。例えば、“formKeys”の“value”として“scan”の文字列が入っていた場合は、初期ファイル名の先頭に“scan”を設定する。次に、“matchingResultUrl”にアクセスして取得した類似帳票判定結果に従って所定の文字列を設定する。すなわち、“matched”の値が“true”であり、かつ、“matchingScore”が一定以上の数値 ( 例えば 0.6 以上 ) であると、“metadataArray”の“value”で指定されるファイル名構成に従って文字列を設定する。具体的には、“value”において指定された順番に従って“rectInfoArray”で特定される文字領域の文字列を設定する。図 1 6 の類似帳票判定結果の例では、“matched”の値が“true”であり、かつ、“matchingScore”が“0.74・・・”なので一定以上の数値である。そして、“metadataArray”の“value”にて、最初が“fileRegion0”、次が“separator”、最後が“fileRegion1”の並びであることが指定されている。よって、“fileRegion0”の“text”の値である「見積書」、処理完了レスポンスに含まれる“separator”の値である「\_(アンダースコア)」、 “fileRegion1”の“text”の値である「下丸子株式会社」が順に連結される。こうして「見積書\_下丸子株式会社」が、初期ファイル名として生成されることになる。もし、“formKeys”の“value”に何らかの値 ( 例えば“scan\_” ) が入っていた場合は、その文字を先頭に追加した「scan\_見積書\_下丸子株式会社」が初期ファイル名となる。類似帳票判定結果において、“matched”の値が“false”で、“matchingScore”が一定未満の数値の場合は、“formKeys”の“value”に文字列が入っていればその文字列がそのまま初期ファイル名となる。また、“formKeys”の“value”に文字列が入っていなければ初期ファイル名なし ( 空欄 ) となる。なお、本実施形態では、“matched”の値と“matchingScore”の値の 2 つを、“metadataArray”の“value”の順番に従って“rectInfoArray”で指定された情報を初期ファイル名生成の条件とした。しかし、これに限定されず、いずれか一方の結果のみを条件としてもよい。例えば、“matched”の値が“true”であることのみを条件に初期ファイル名を生成してもよいし、類似度合いを示す値が閾値以上であることのみを条件にファイル名を生成してもよい。

#### 【 0 0 5 9 】

S 1 8 0 2 では、ファイル名設定画面 1 0 0 0 のプレビュー領域 1 0 0 2 に初期状態で表示する際にスキャン画像をどの程度縮小して表示するか ( 以下、「初期変倍率」と呼ぶ

10

20

50

）が決定される。この初期変倍率の決定には、“ocrResultUrl”にアクセスして取得した文字認識結果を用いる。まず、スキャン画像内で検出された全ての文字領域のうち最も左にある文字領域と最も右にある文字領域をそれぞれ求める。この場合において、最も左の文字領域となるのは、全文字領域の中で、x座標の値が最小値の文字領域である。また、最も右の文字領域となるのは、全文字領域の中で、x座標の値と文字領域幅（width）の値とを足した合計値が最大となる文字領域である。こうして決定した最も左にある文字領域の左辺がプレビュー領域1002の左辺と、最も右にある文字領域の右辺がプレビュー領域1002の右辺とそれぞれ一致するような変倍率を、初期変倍率として決定する。なお、決定した文字領域の端とプレビュー領域の端とを完全に一致させてしまうと、当該文字領域が小さい場合にユーザが選択（押下）しづらくなることがあるため、一定量のマージンを持たせて初期変倍率を決定してもよい。例えば、プレビュー領域1002内で最も左に表示されることになる文字領域の左端と、最も右に表示されることになる文字領域の右端からそれぞれ10ピクセル分の余白が確保されるように初期変倍率を決定するといった具合である。また、プレビュー領域1002上に各種操作ボタン等のUIコンポーネントが重畳表示される場合には、当該UIコンポーネントに文字領域が重ならないように初期変倍率を決定してもよい。具体的には、UIコンポーネントの幅分をプレビュー領域1002の幅から差し引いた上で、最も左の文字領域と最も右の文字領域とが残りの領域の幅内に収まるように初期変倍率を決定してもよい。前述の図10のファイル名設定画面1000の例では、プレビュー領域1002内に3つの操作ボタン（拡大ボタン1004、縮小ボタン1005、復帰ボタン1006）が存在しているので、これらのボタンに文字領域が重ならないように初期変倍率を決定する。また、スキャン対象となる帳票の上下左右の端部付近には印刷処理に関わるヘッダ情報等が印刷されていることがあり、このようなヘッダ情報等に対応する文字領域の文字列がファイル名の一部に採択される可能性は低い。そこで、スキャン画像の上下左右の端から一定量（例えば50ピクセル）以内に存在する文字領域は、上記最も左及び最も右の文字領域を求める際に除外するようにしてもよい。

#### 【0060】

S1803では、S1802で決定した初期変倍率による拡張後のスキャン画像のどの部分を、ファイル名設定画面1000のプレビュー領域1002内に初期状態で表示するかが決定される。上記S1802で決定された変倍率にて拡張した後のスキャン画像においては、最も左の文字領域と最も右の文字領域とが収まる幅が、プレビュー領域1002の幅と略一致する。したがって、まず、左右方向についてはプレビュー領域1002の中にこれら左右の文字領域がちょうど収まるように、例えば最も左の文字領域の左辺がプレビュー領域1002の左辺と一致するように初期状態で表示する部分（以下、「初期表示部分」と呼ぶ。）が決まる。その上で、最も上にある文字領域がプレビュー領域1002の上端付近に位置するように、上下方向の初期表示部分を決定する。このとき、初期変倍率の決定時にマージンを考慮していた場合は、同様にマージン分を考慮して、初期表示部分を決定すればよい。例えば、スキャン画像の上端から一定量（例えば50ピクセル）以内に存在する文字領域は表示対象から除外し、それ以外の文字領域の中で最も上にある文字領域が、プレビュー領域1002の上端に表示されるように決定してもよい。図20は、図13のスキャン画像について決定された初期表示部分を示す図である。図20における線の枠が、プレビュー領域1002に初期状態で表示されるスキャン画像の範囲を示している。なお、上述のようにして初期表示部分を決定した結果、類似帳票が存在して初期ファイル名がファイル名入力欄1001に自動で入力された場合における当該初期ファイル名に使用された文字列の文字領域が、プレビュー領域1002に表示されないケースがある。この場合には、例えば、初期ファイル名に使用した文字列の文字領域がプレビュー領域1002の中央に表示されるように、初期表示部分を決定すればよい。この時、ファイル名入力欄1001に自動入力された初期ファイル名を構成する文字列と当該初期ファイル名に使用された文字列の文字領域とを、それぞれ強調表示（例えば色を変更するなど）してもよい。図21は、図13のスキャン画像において文字列「川崎株式会社」が初期ファイル名に自動入力されている場合のスキャン設定画面1000の例である。ここでは

10

20

30

40

50

、文字列「川崎株式会社」に対応する文字領域がプレビュー領域 1 0 0 2 の中央に位置するように、さらに、当該文字領域とファイル名入力欄 1 0 0 1 に入力されている文字列「川崎株式会社」の色を変更して表示している。この時、強調表示する文字領域の文字サイズが小さい場合には、適切な文字サイズとなるように初期変倍率を調整してもよい。この調整後の初期変倍率は、例えば、操作部 2 2 0 のタッチパネルのサイズとプレビュー領域 1 0 0 2 に初期状態で表示する範囲とに基づき得ることができる。具体的には、初期ファイル名に使用している文字列の文字領域の高さを基に、文字サイズが何ポイントかを求め、求めたポイントが閾値以上（所定サイズ以上）になるようにプレビューの初期変倍率を調整すればよい。

#### 【 0 0 6 1 】

S 1 8 0 4 では、S 1 8 0 1 ~ S 1 8 0 3 までの処理結果を用いて、前述のファイル名設定画面 1 0 0 0 が操作部 2 2 0 に表示される。ユーザは、図 1 0 に例示したファイル名設定画面 1 0 0 0 を用いて、ストレージサーバ 1 3 0 に保存するスキャン画像についてのファイル名を設定する。以下、詳しく説明する。

#### 【 0 0 6 2 】

##### 1 ) ファイル名入力欄

ユーザが、ファイル名入力欄 1 0 0 1 をタッチすると、操作部 2 2 0 のタッチパネルの表示がソフトキーボードに切り替わり、任意の文字を入力したり自動入力された文字列を編集したりすることができる。例えば、前述の図 2 1 の例のように「川崎株式会社」が自動入力された状態のファイル名入力欄 1 0 0 1 の文字列部分をタッチすると、図 2 2 に示すソフトキーボード 2 2 0 0 が表示される。この際、図 2 2 に示すように、入力エリア 2 2 0 1 にタッチされた文字列が入力された状態で表示される。これにより、ユーザは、ソフトキーボード 2 2 0 0 を操作して、タッチした「川崎株式会社」の文字列を修正することができる。また、図 1 0 に示すような文字列が未入力状態のファイル名入力欄 1 0 0 1 がタッチされた場合は、入力エリア 2 2 0 1 が未入力状態のソフトキーボード 2 2 0 0 が表示され、ユーザはこれを操作して任意の文字列をファイル名入力欄 1 0 0 1 に入力することができる。

#### 【 0 0 6 3 】

##### 2 ) プレビュー領域

プレビュー領域 1 0 0 2 には、前述のとおり、変倍処理が施されたスキャン画像の特定部分が表示される。そして、プレビュー領域 1 0 0 2 内の任意の文字領域をタッチして選択すると、当該選択した文字領域の文字列がファイル名入力欄 1 0 0 1 に追加される。この際、選択したことが分かるよう、選択に係る文字領域に対して下線を付したり、枠で囲んだり、色を付すなどして強調表示してもよい。また、選択した文字領域が中央になるよう表示部分を変更したり或いは表示倍率を変更したりしてもよい。さらに、複数の文字領域が選択された場合においては、それぞれの文字領域に対して異なる色を付す等、文字領域毎に表示態様を異ならせてもよい。そして、ユーザは、一度選択した文字領域を再度タッチすると、選択状態が解除（すなわち、文字領域に付した線や色を削除し元の状態に戻す。）され、ファイル名入力欄 1 0 0 2 内の当該文字領域に対応する文字列も削除される。なお、本実施形態では、文字領域が未選択の状態では、プレビュー領域 1 0 0 2 に各文字領域を示す枠等は表示しないこととしている。しかし、ユーザがタッチ可能な部分を認識できるよう各文字領域を示す枠等を予め表示しておき、選択された際にはその色を変える等してもよい。また、各文字領域が分かるようにする表示を行うかどうかをユーザがボタン等で切り替えられるようにしてもよい。そして、ユーザは、プレビュー領域 1 0 0 2 においてスワイプ操作を行うことで、表示される画像部分を移動することができる。

#### 【 0 0 6 4 】

##### 3 ) 各種操作ボタン

「×」マークのボタン 1 0 0 3 は、ファイル名入力欄 1 0 1 に入力中の文字列のうち末尾の文字列を削除するための削除ボタンである。「+」マークのボタン 1 0 0 4 は、プレビュー領域 1 0 0 2 に表示中の画像を拡大して表示させるための拡大ボタンであり、各文

10

20

30

40

50

字領域をより大きく表示させたいときなどに用いる。「-」マークのボタン 1005 は、プレビュー領域 1002 に表示中の画像を縮小して表示させるための縮小ボタンであり、より広い範囲を表示させたいときなどに用いる。拡大や縮小を行う時には、プレビュー領域 1002 の中央の座標が拡大や縮小を行う前と同一となるように表示位置の調整がなされる。「Uターン矢印」マークのボタン 1006 は、スワイプ操作による表示位置の移動や、拡大ボタン 1004 や縮小ボタン 1005 の押下による表示倍率の変更を行っていた場合に、プレビュー領域 1002 を初期表示状態に戻すための復帰ボタンである。「送信」ボタン 1007 は、ファイル名設定画面 1000 で設定したファイル名にてスキャン画像をファイル化してストレージサーバ 130 に保存させるためのボタンである。この「送信」ボタン 1007 の押下に応答して、設定されたファイル名の情報と共にスキャン画像データが MFP 連携サーバ 120 へ送信される。

10

#### 【0065】

S1805 では、初期表示状態のファイル名設定画面 1000 内に、操作ガイダンスが表示される。この操作ガイダンスによって、初めて操作するユーザでも、プレビュー領域 1002 内の文字領域をタッチして選択することでファイル名が設定できるということを容易に理解することができる。図 23 は、図 10 に示すファイル名設定画面 1000 におけるプレビュー領域 1002 の上部に、操作ガイダンスとしてのメッセージ 2301 を重畳表示した状態を示している。このような操作ガイダンスは、操作方法を既に知っているユーザにとっては不要となるため、タッチ操作やスワイプ操作など何らかのユーザ操作を検知した時点で、当該操作ガイダンスは非表示にされる。また、一般的な文書の上部には例えば文書タイトル、会社名、帳票番号、日付といった、当該文書にとって重要な意味を持つ文字や数字が存在することが多い。よって、これら重要な意味を持つ文字列の文字領域と操作ガイダンスとが重ならないよう、文字列の属性に応じて、プレビュー領域 1002 に表示する画像部分を下にずらす等してもよい。また、いずれかの文字領域に操作ガイダンスが重なってしまったり或いは文字領域から一定距離以内に入ってしまうときは、重なったり近づきすぎたりしないように操表示位置を決定すればよい。このように、操作ガイダンスの表示位置を固定せずに柔軟に決定することで、ユーザが文字領域を選択しやすくなる。

20

#### 【0066】

S1806 は、ユーザ操作の有無の監視処理である。所定間隔でユーザ操作の有無をチェックし、何らかのユーザ操作が検知されれば S1807 に進む。そして、S1807 では、前述のとおり、操作ガイダンスが非表示にされる。

30

#### 【0067】

S1808 ~ S1815 は、検知されたユーザ操作の内容に応じて次の処理を決定する判定処理である。なお、検知されたユーザ操作に対応する処理が完了すると S1806 に戻り、次のユーザ操作が検知されるのを待つ。

#### 【0068】

S1808 では、検知されたユーザ操作が、プレビュー領域 1002 内のいずれかの文字領域へのタッチ操作であったかが判定される。具体的には、まず、タッチされた位置の座標をスキャン画像の座標に対応付けて変換する。そして、変換後の座標が、前述の文字認識結果（図 15 を参照）の“regions”で特定される各文字領域のうちいずれかの領域内に含まれれば、文字領域へのタッチであると判定する。判定の結果、文字領域へのタッチであれば S1816 に進み、文字領域へのタッチでなければ S1809 に進む。文字領域へのタッチであった場合の S1816 では、タッチされた文字領域が、既に選択されているかが判定される。タッチされたのが未選択の文字領域の場合は S1817 に進み、タッチされたのが選択済みの文字領域の場合は S1818 に進む。

40

#### 【0069】

S1817 では、タッチされた文字領域の文字列がファイル名入力欄 1001 に追加され、当該文字領域に対して、選択されたことを示す強調表示処理がなされる。この際、ファイル名入力欄 1001 に他の文字列が入力されていないければ、そのまま文字列が追加さ

50

れる。そして、当該文字領域をファイル名設定に使用したことが、ファイル名設定時の入力情報として記憶される。一方、ファイル名入力欄 1 0 0 1 に既に他の文字領域の文字列が入力されていた場合は、区切り文字が追加された後、タッチされた文字領域の文字列が追加される。この場合、当該文字領域をファイル名設定に使用したこと事実に加え、「第 1 の文字領域」、「区切り文字」、「第 2 の文字領域」といったように、文字列間に区切り文字を挿入したことも記憶される。図 2 4 は、「見積書」と文字認識された文字領域が最初に選択され、続いて「下丸子株式会社」と文字認識された文字領域が選択された場合のファイル設定画面 1 0 0 0 を示した図である。ファイル名入力欄 1 0 0 1 には、タッチした文字領域の順に対応する 2 つの文字列が入力されており、当該 2 つの文字列の間にセパレータとしてのアンダースコアが入力されている。

10

#### 【 0 0 7 0 】

S 1 8 1 8 では、タッチされた文字領域の文字列がファイル名入力欄 1 0 0 1 から削除され、当該文字領域についての選択されたことを示す強調表示が取り消され、選択されていないことを示す通常表示の状態に戻る。また、その文字列に区切り文字が付いていた場合は、当該区切り文字も一緒に削除される。例えば、初期ファイル名の生成に用いたファイル名構成が[「第 1 の文字領域」、「区切り文字」、「第 2 の文字領域」]であったとする。この場合において、「第 1 の文字領域」となっている文字領域をユーザがタッチした場合は、当該第 1 の文字領域の文字列と一緒に区切り文字も削除され、ファイル名入力欄 1 0 0 1 には「第 2 の文字領域」の文字列のみが残ることになる。

#### 【 0 0 7 1 】

20

検知されたユーザ操作が文字領域へのタッチでなかった場合の S 1 8 0 9 では、検知されたユーザ操作が、プレビュー領域 1 0 0 2 内でのスワイプ操作であったかどうか判定される。スワイプ操作が検知された場合は S 1 8 1 9 に進み、そうでなければ S 1 8 1 0 に進む。

#### 【 0 0 7 2 】

S 1 8 1 9 では、プレビュー領域 1 0 0 2 に表示する画像部分をスワイプされた方向に移動させる処理が実行される。具体的には、ユーザがタッチパネル上で指を動かした移動量に応じて、プレビュー領域 1 0 0 2 に表示する画像部分を更新する。この処理は、プレビュー領域 1 0 0 2 内の文字領域以外の領域でのタッチ操作を検知した場合に限定して行ってもよい。また、文字領域へのタッチ操作であった場合もその後のユーザが指を移動させた量に応じて文字領域へのタッチとして扱わずにスワイプ操作として扱ってもよい。

30

#### 【 0 0 7 3 】

S 1 8 1 0 では、検知されたユーザ操作が、拡大ボタン 1 0 0 4 又は縮小ボタン 1 0 0 5 の押下であったかどうか判定される。拡大ボタン 1 0 0 4 又は縮小ボタン 1 0 0 5 の押下であった場合は S 1 8 2 0 に進み、そうでなければ S 1 8 1 1 に進む。

#### 【 0 0 7 4 】

S 1 8 2 0 では、プレビュー領域 1 0 0 2 に表示中の画像部分を、拡大又は縮小する処理が実行される。具体的には、拡大ボタン 1 0 0 4 が押下されていた場合はスキャン画像の変倍率を一定量増やし、縮小ボタン 1 0 0 5 が押下されていた場合はスキャン画像の変倍率を一定量減らす処理がなされる。ここでの一定量は任意であるが、例えば現状の倍率を基準として 1 0 % 程度である。この際、一定量の拡大又は縮小を行うまでの間に、初期変倍率が存在したり、スキャン画像の幅 / 高さ と プレビュー領域 1 0 0 2 の幅 / 高さ と が一致する倍率が存在したときは、その倍率で留めてもよい。なお、プレビュー領域 1 0 0 2 内にスキャン画像全体が収まる倍率（最小倍率）未满是縮小されず、最小の文字領域が一定サイズの大きくなる倍率（最大倍率）以上には拡大できないように制御される。

40

#### 【 0 0 7 5 】

S 1 8 1 1 では、検知されたユーザ操作が、復帰ボタン 1 0 0 6 の押下であったかどうか判定される。復帰ボタン 1 0 0 6 の押下であった場合は S 1 8 2 1 に進み、そうでなければ S 1 8 1 2 に進む。

#### 【 0 0 7 6 】

50

S 1 8 2 1では、プレビュー領域 1 0 0 2を初期表示状態に戻す処理が実行される。具体的には、プレビュー領域 1 0 0 2に表示するスキャン画像の変倍率をS 1 8 0 2で決定した初期変倍率に戻し、さらに、プレビュー領域 1 0 0 2への表示部分をS 1 8 0 3で決定した初期表示部分に戻す処理がなされる。この際、初期変倍率や初期表示部分の調整がなされていた場合には、当該調整後の初期変倍率や初期表示部分に戻すのが望ましい。

【 0 0 7 7 】

S 1 8 1 2では、検知されたユーザ操作が、ファイル名入力欄 1 0 0 1に入力されている文字列へのタッチ操作であったかが判定される。この際、ファイル名入力欄 1 0 0 1に入力されている文字列の後ろの空白領域に対してのタッチ操作が検知された場合は、最後尾の文字列へのタッチ操作と扱えばよい。ファイル名入力欄 1 0 0 1に入力されている文字列へのタッチ操作であればS 1 8 2 2に進み、そうでなければS 1 8 1 3に進む。

【 0 0 7 8 】

S 1 8 2 2では、タッチ操作がなされた文字列が入力エリア 2 2 0 1に入力された状態の編集用ソフトキーボードが操作部 2 2 0に表示される。いま、ファイル名設定画面 1 0 0 0が図 2 4に示す状態であったと仮定する。この状態下で、ファイル名入力欄 1 0 0 1の「見積書」の文字列へのタッチ操作が検知された場合は、ソフトキーボードの入力エリア 2 2 0 1に「見積書」の文字列が入力された状態のソフトキーボード 2 2 0 0が表示されることになる。また、「下丸子株式会社」の文字列へのタッチ操作或いはその後ろの空白領域へのタッチ操作が検知された場合は、末尾の文字列である「\_下丸子株式会社」の文字が入力エリア 2 2 0 1に入力された状態のソフトキーボード 2 2 0 0が表示されることになる。なお、ソフトキーボード 2 2 0 0の操作は一般的なキーボードの操作と同じであるためその説明を省略する。続くS 1 8 2 3は、ユーザ操作の有無の監視処理である。ソフトキーボード 2 2 0 0におけるOKボタン 2 2 0 2の押下の有無を所定間隔でチェックし、OKボタン 2 2 0 2の押下が検知されればS 1 8 2 4に進む。なお、OKボタン以外のキーが押下された場合は、当該押下されたキーに応じて、入力エリア 2 2 0 1に表示されている文字列の編集が行なわれることになる。例えば、ユーザは、ソフトキーボード 2 2 0 0を用いて、文字認識処理で誤認識していた文字列の修正を行うことができる。

【 0 0 7 9 】

S 1 8 2 4では、区切り文字を削除する変更が行われたかが判定される。本実施形態ではこの判定は、S 1 8 1 2でタッチされた文字列がS 1 8 2 2で区切り文字とともに表示された場合にのみ行なわれ、区切り文字とともに表示されなかった文字列については、区切り文字の削除が行われなかったものとして扱うものとする。すなわち、S 1 8 2 2で文字列の先頭に区切り文字が付与された状態で表示された場合、OKボタン 2 2 0 2が押下された時点で、文字列の先頭に区切り文字が存在するかどうかに基づいて、区切り文字を削除するような文字列の編集が行われたか判定する。例えば、S 1 8 1 2で「下丸子株式会社」の文字列へのタッチ操作がなされると、上述のとおりソフトキーボード 2 2 0 0の入力エリア 2 2 0 1には区切り文字が先頭に付された「\_下丸子株式会社」の状態が表示される。そこで、先頭に区切り文字が残っているかどうかをチェックすることで、区切り文字の削除が行われたかどうかを判定する。また、S 1 8 1 2で「見積書」の文字列へのタッチ操作がなされると、上述のとおり入力エリア 2 2 0 1には「見積書」が表示されるので、この場合は、区切り文字を削除する変更は行われなかったものとして扱う。

なお、上述した例では、S 1 8 1 2でタッチされた文字列の前に区切り文字が存在する場合に、S 1 8 2 2で、区切り文字が先頭に付された文字列が入力エリア 2 2 0 1に表示されるものとして説明したが、これに限るものではない。例えば、S 1 8 1 2でタッチされた文字列の後ろに区切り文字が存在する場合に、S 1 8 2 2で、当該タッチされた文字列の後ろに区切り文字を付して入力エリア 2 2 0 1に表示するように構成した場合も、S 1 8 2 4で同様の判定を行うことができる。すなわち、図 2 4の例で「見積書」の文字列へのタッチ操作に応じてソフトキーボード 2 2 0 0の入力エリア 2 2 0 1に「見積書\_」のように末尾に区切り文字を付して表示するように構成した場合について説明する。この場合は、本ステップS 1 8 2 4において末尾に区切り文字が残っているかどうかをチェックす

10

20

30

40

50

ることで、区切り文字の削除が行われたかどうかを判定すればよい。

【 0 0 8 0 】

S 1 8 2 4 で区切り文字の削除が行われたと判定された場合は S 1 8 2 5 に進む。そして、S 1 8 2 5 では、区切り文字がファイル名から削除されたことが記憶される。例えば、初期ファイル名の生成に用いたファイル名構成が [「第 1 の文字領域」、「区切り文字」、「第 2 の文字領域」] であったとする。この時に、ファイル名入力欄 1 0 0 1 における「第 2 の文字領域」に対応する文字列をタッチし、ソフトキーボード 2 2 0 0 を使用して区切り文字を削除すると、変更後のファイル名構成は [「第 1 の文字領域」、「第 2 の文字領域」] となる。そして、この変更後のファイル名構成の情報が保持されることになる。続く S 1 8 2 6 では、ファイル名に区切り文字を追加する変更が行われたかが判定される。区切り文字の追加が行われたと判定された場合は S 1 8 2 7 に進む。そして、S 1 8 2 7 では、区切り文字がファイル名に追加されたことが記憶される。例えば、初期ファイル名の生成に用いたファイル名構成が [「第 1 の文字領域」、「第 2 の文字領域」] であったとする。この時に、ファイル名入力欄 1 0 0 1 における「第 2 の文字領域」に対応する文字列をタッチし、ソフトキーボード 2 2 0 0 を使用して区切り文字を先頭に追加すると、変更後のファイル名構成は [「第 1 の文字領域」、「区切り文字」、「第 2 の文字領域」] となる。そして、この変更後のファイル名構成の情報が保持されることになる。

10

【 0 0 8 1 】

そして、S 1 8 2 8 では、操作部 2 2 0 の表示画面が、ソフトキーボード 2 2 0 0 からファイル名設定画面 1 0 0 0 に戻された上で、変更後の内容にファイル名が修正される。この際、修正を行った文字列に対応する文字領域がプレビュー領域 1 0 0 2 の中央にくるようにスキャン画像の表示部分を変更してもよい。なお、ファイル名に使用されている文字列の一部が修正されても、ファイル名構成が変わっていなければ、ファイル名構成の情報が新たに保持されることはない。ただし、ある文字列に含まれる全ての文字を削除する修正を行った場合はファイル名構成も変わる。よってこの場合にも（フローチャートでは省略しているが）、変更後のファイル名の構成情報が保持されることになる。例えば、初期ファイル名の生成に用いたファイル名構成が [「第 1 の文字領域」、「区切り文字」、「第 2 の文字領域」] であったとする。この時に、ファイル名入力欄 1 0 0 1 における「第 2 の文字領域」に対応する文字列をタッチし、ソフトキーボード 2 2 0 0 を使用してすべての文字を削除したとする。この場合、不要となった区切り文字も削除されて変更後のファイル名構成は [「第 1 の文字領域」] に変わるので、当該変更後のファイル名構成の情報が保持されることになる。

20

30

【 0 0 8 2 】

S 1 8 1 3 では、検知されたユーザ操作が、文字列が入力されていない状態の空欄のファイル名入力欄 1 0 0 1 へのタッチ操作であったかどうか判定される。空欄のファイル名入力欄 1 0 0 1 へのタッチ操作であれば S 1 8 2 9 に進み、そうでなければ S 1 8 1 4 に進む。

【 0 0 8 3 】

S 1 8 2 9 では、入力エリア 2 2 0 1 に何も入力されていない状態の文字列追加用ソフトキーボードが操作部 2 2 0 に表示される。続く S 1 8 3 0 は、ユーザ操作の有無の監視処理である。ソフトキーボード 2 2 0 0 における OK ボタン 2 2 0 2 の押下の有無を所定間隔でチェックし、OK ボタン 2 2 0 2 の押下が検知されれば S 1 8 3 1 に進む。そして、S 1 8 3 1 では、操作部 2 2 0 の表示画面が、ソフトキーボード 2 2 0 0 からファイル名設定画面 1 0 0 0 に戻された上で、入力された文字列がファイル名に追加される。

40

【 0 0 8 4 】

S 1 8 1 4 では、検知されたユーザ操作が、削除ボタン 1 0 0 3 の押下であったかどうか判定される。削除ボタン 1 0 0 3 の押下であった場合は S 1 8 3 2 に進み、そうでなければ S 1 8 1 5 に進む。S 1 8 3 2 では、ファイル名入力欄 1 0 0 1 に入力されている文字列のうち末尾の文字列が削除される。例えば、初期ファイル名の生成に用いたファイル名構成が [「第 1 の文字領域」、「区切り文字」、「第 2 の文字領域」] であれば、削除ボ

50

タン 1 0 0 3 の押下によって「第 2 の文字領域」に対応する文字列が削除される。この結果、削除後のファイル名構成は[「第 1 の文字領域 1」]となり、ファイル名は「第 1 の文字領域」に対応した文字列のみとなる。この際、削除された文字列に対応する文字領域については、プレビュー領域 1 0 0 2 において非選択を示す状態に戻される。そして、変更後のファイル名構成の情報が保持されることになる。

【 0 0 8 5 】

S 1 8 1 5 では、検知されたユーザ操作が、送信ボタン 1 0 0 7 の押下であったかどうか判定される。送信ボタン 1 0 0 7 の押下であった場合は、本処理を終了し、図 1 2 のフローチャートの S 1 2 1 0 に進む。

【 0 0 8 6 】

以上が、S 1 2 0 9 にて実行されるファイル名設定処理の内容である。

【 0 0 8 7 】

< 変形例 1 >

上述の S 1 8 0 2 ( 初期変倍率の決定 ) において、最も左の文字領域と最も右の文字領域とに基づき決定した初期変倍率に従うと、プレビュー表示される各文字領域の高さ平均が所定の閾値未満となる場合は、所定の閾値以上となるように初期変倍率を変更してもよい。この場合において、「高さの平均」は、全文字領域についての平均でもよいし、一部の文字領域についての平均でもよい。一部の文字領域についての平均の例としては、決定された初期変倍率にてプレビュー領域 1 0 0 2 に表示されることになる文字領域の高さの平均や、スキャン画像の特定部分 ( 例えば上半分 ) の文字領域の高さの平均などが挙げられる。具体例を説明する。いま、最も左の文字領域の左端と最も右の文字領域の右端との間隔 ( 幅 ) が 2000px、プレビュー領域 1 0 0 2 の幅が 800px、所定の閾値が 15px であったとする。ここで、全文字領域の高さの平均が 60px であった場合に算出される初期変倍率は  $0.4 (=800 \div 2000)$  であり、全文字領域の高さの平均は  $24px (=60 \times 0.4)$  で所定の閾値未満ではないため 0.4 がそのまま初期変倍率となる。一方、全文字領域の高さの平均が 30px であった場合には、全文字領域の高さの平均は  $12px (=30 \times 0.4)$  となる。この場合、所定の閾値未満となるため、全文字領域の高さの平均が所定の閾値以上 ( ここでは閾値と同じ値 ) になるように、初期変倍率を  $0.5 (=0.4 \times 15 \div 12)$  に変更する。図 2 5 に、文字領域の高さの平均が閾値を満たせずに初期変倍率を調整した場合の、スキャン設定画面 1 0 0 0 の初期表示の一例を示す。この例では、初期変倍率の値を大きくする調整を行った後、S 1 8 0 3 において最も左の文字領域の左端と最も上の文字領域の上端に合わせて表示部分が決定されている。その結果、帳票の右側に相当する領域が表示部分に含まれなくなるが、プレビュー領域 1 0 0 2 内に存在する各文字領域については視認性が確保されることになる。

【 0 0 8 8 】

< 変形例 2 >

上述の S 1 8 0 3 ( 初期表示部分の決定 ) において、初期ファイル名に使用した文字領域が複数ある場合は、使用した文字領域のうち一番上にある文字領域がプレビュー領域 1 0 0 2 に表示されるようにしてもよい。また、各文字領域における文字列の意味種別を判断し、ファイル名に使用する可能性の高い文字領域がプレビュー領域 1 0 0 2 内に表示されるように表示部分を調整してもよい。また、予め指定した文字領域の数分の文字領域がプレビュー領域 1 0 0 2 に表示されるように、表示部分を決定してもよい。さらに、初期ファイル名に使用する文字列の文字領域のうち、一番上にある文字領域と一番下にある文字領域の中間地点が、プレビュー領域 1 0 0 2 の中央にくるように表示部分を決定してもよい。

【 0 0 8 9 】

< 変形例 3 >

上述の S 1 8 0 5 ( 操作ガイダンスの表示 ) において、類似帳票のファイル名構成に従った初期ファイル名がファイル名入力欄 1 0 0 1 に自動入力されている場合、当該初期ファイル名に使用されている文字列の文字領域に重ならないように表示してもよい。この際

10

20

30

40

50

、初期ファイル名に使用されている文字列が複数存在し、それらの文字領域に対して重ならないように表示することが困難な場合も考えられる。このような場合には、初期ファイル名の先頭にある文字列の文字領域を優先し、末尾にある文字列の文字領域の一部又は全部に重ねて表示してもよい。また、初期ファイル名以外の文書プロパティ、例えばメタデータなどを設定する場合には、その文書プロパティのタイプに応じて操作ガイダンスの重なりを避ける文字領域を判定し、操作ガイダンスの表示位置を決めてもよい。例えば、タイプが日付型の文書プロパティに値をセットする場合であれば、対象文書における日付の文字領域には重ならないように操作ガイダンスを表示する。このとき、各文字領域に対応する文字列が日付かどうかは、スキャン画像から抽出された文字列が特定の文字配列に当て嵌るかによって判断する。例えば、yy/mm/ddの形式における、「yy」、「mm」、「dd」のそれぞれに数字が入る場合は、日付であると判断することができる。

10

【0090】

<変形例4>

上述のS1822（編集用ソフトキーボードの表示）において、入力エリア2201内に、タッチされた文字列を入力状態にして表示するのに代えて、ファイル名全体を入力状態にして表示してもよい。この場合、編集対象をユーザが誤認しないよう、他の文字列と区別する強調表示（色を変える、枠を付すなど）を行ったり、カーソルをその末尾に表示するなどし、編集対象の範囲が分かるように表示する。また、ファイル名入力欄1001に入力されている文字列に対するタッチ操作にตอบสนองして編集用ソフトキーボードを表示するのに代えて、さらに追加的なユーザ操作（例えば不図示の専用ボタンの押下）を待って編集用ソフトキーボードを表示するようにしてもよい。

20

【0091】

<変形例5>

また、本実施形態の説明では、ユーザによる文字領域へのタッチ操作が検出された場合には、タッチされた文字領域の文字列をファイル名に追加或いは削除する処理（S1817、S1818）が実行される。しかしながら、システムの応答が遅く、タッチ操作を検知してからその対象文字列の追加や削除がファイル名に反映されるまでに時間が掛かることがある。この場合、ユーザは自身のタッチ操作にシステムが反応していないと勘違いし、何度も同じ文字領域に対してタッチ操作を繰り返すことが考えられる。こうなると、同じ文字領域の文字列の追加と削除が繰り返し行われて、ユーザが所望する状態で止めることができない場合がありうる。例えば、削除対象の文字列を複数回タッチすると、削除対象の文字列が再入力されたりといったことが起こり得る。そこで、追加・削除処理中に同じ文字領域に対するタッチ操作が繰り返し行われないよう、追加・削除処理が完了するまでは同じ文字領域に対する新たな追加・削除のためのタッチ操作を受け付けないようにしてもよい。この際、処理中であることを示すメッセージの表示を併せて行うようにしてもよい。これにより、ファイル名設定時の入力操作において、ユーザが意図しない文字列の追加や削除が実行されてしまうのを防ぐことができる。

30

【0092】

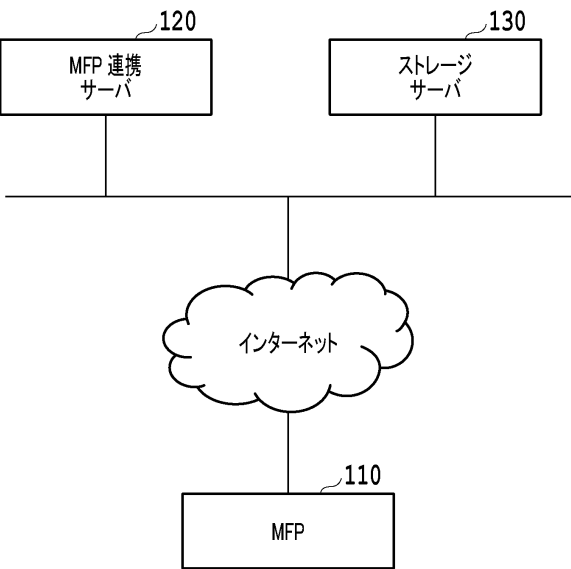
（その他の実施例）

本発明は、上述の実施形態の1以上の機能を実現するプログラムを、ネットワーク又は記憶媒体を介してシステム又は装置に供給し、そのシステム又は装置のコンピュータにおける1つ以上のプロセッサがプログラムを読み出し実行する処理でも実現可能である。また、1以上の機能を実現する回路（例えば、ASIC）によっても実現可能である。

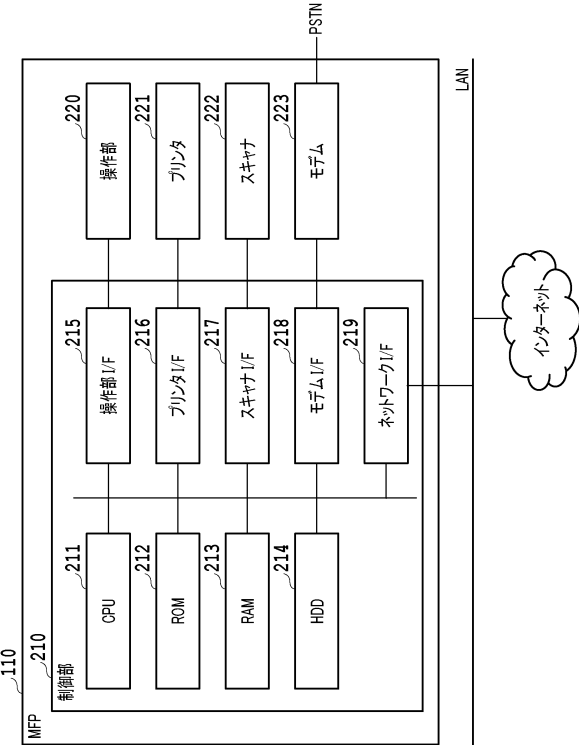
40

【図面】

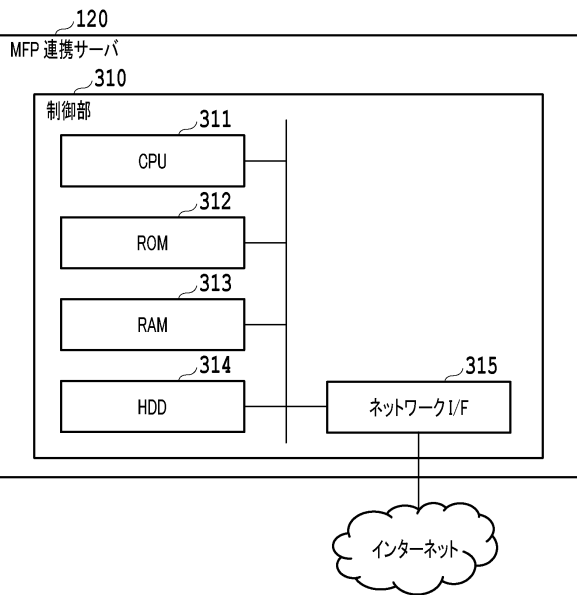
【図 1】



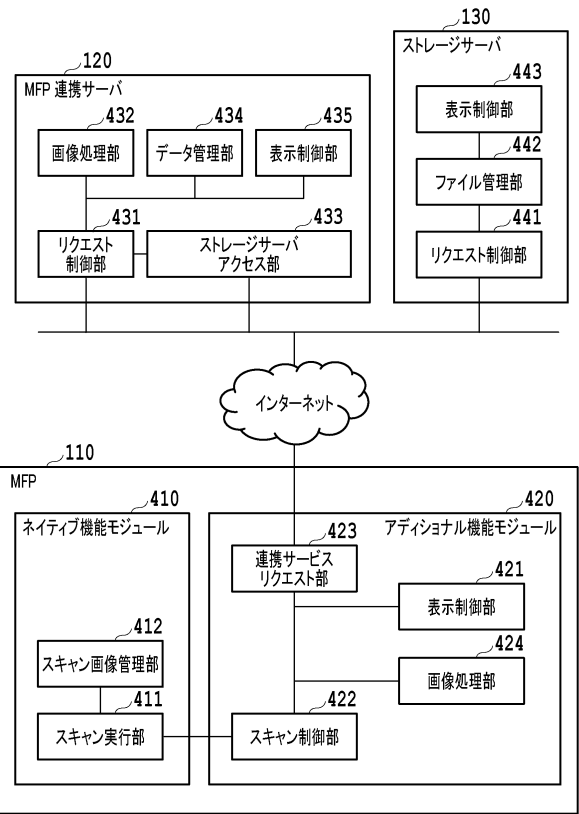
【図 2】



【図 3】



【図 4】



10

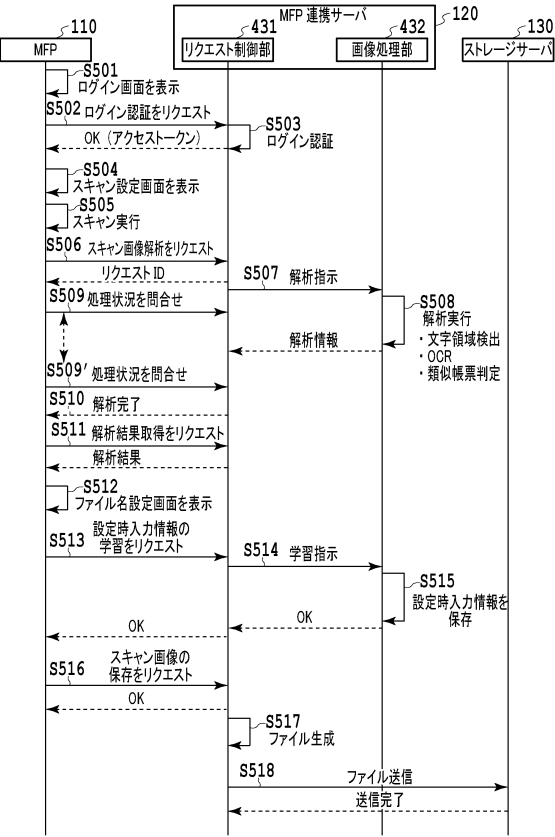
20

30

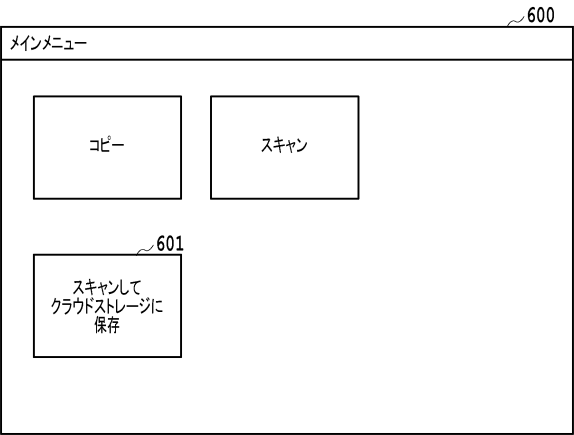
40

50

【 図 5 】



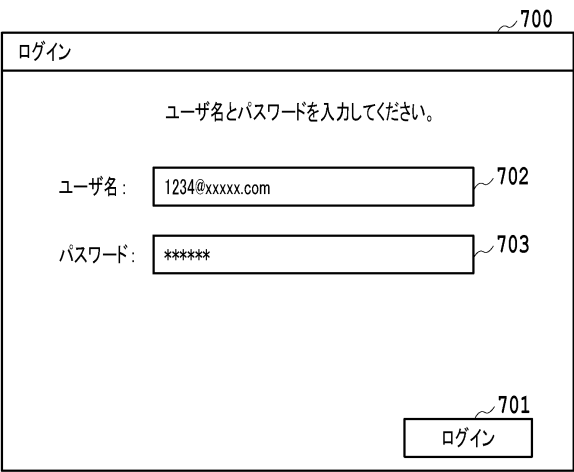
【 図 6 】



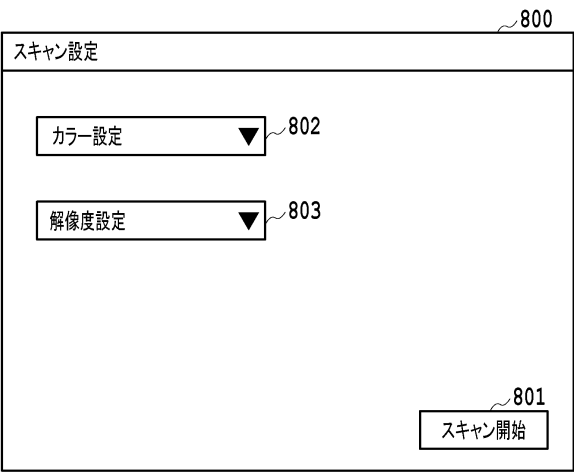
10

20

【 図 7 】



【 図 8 】

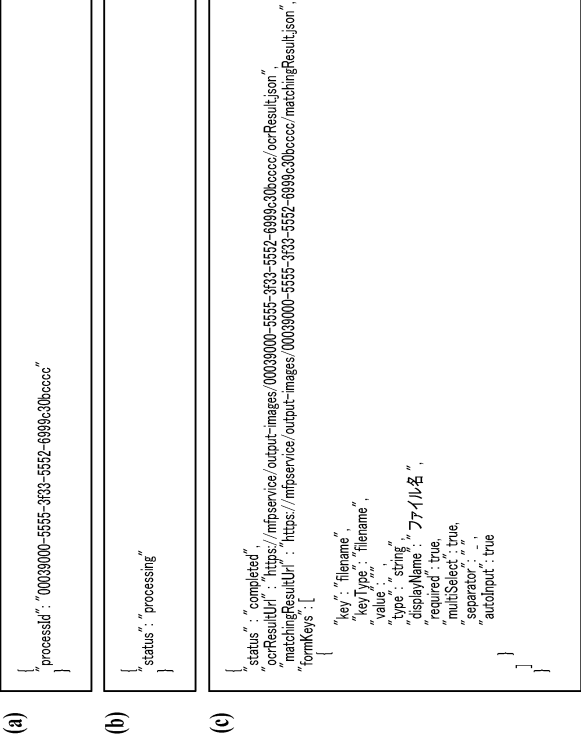


30

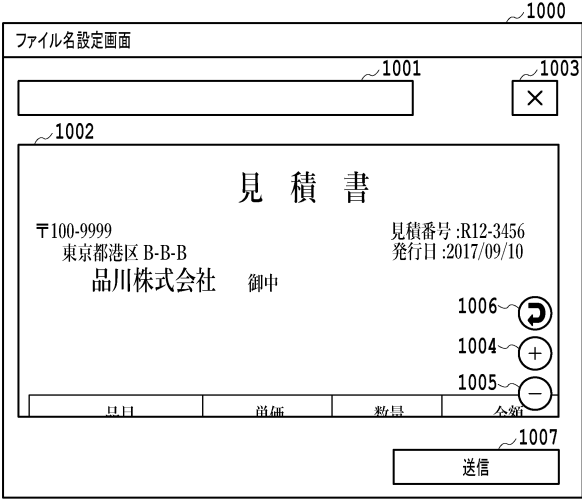
40

50

【図 9】



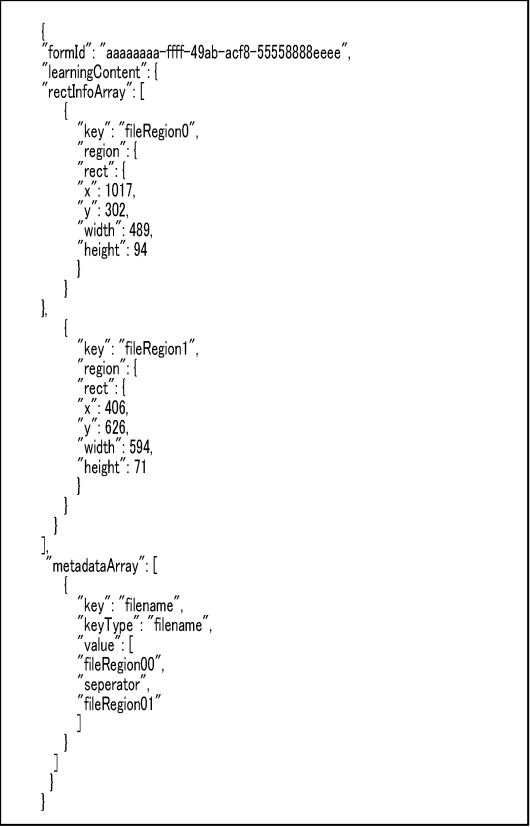
【図 10】



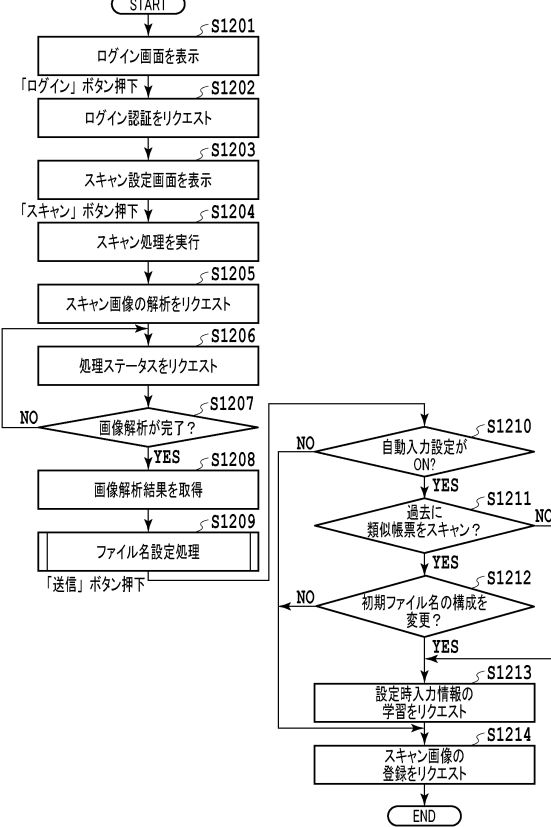
10

20

【図 11】



【図 12】



30

40

50

【 図 1 3 】

見 積 書

〒100-9999  
東京都港区 B-B-B  
品川株式会社 御中

見積番号 :R12-3456  
発行日 :2017/09/10

| 品目      | 単価      | 数量 | 金額      |
|---------|---------|----|---------|
| GF-1544 | 11,550  | 3  | 34,650  |
| EF24-70 | 134,100 | 1  | 134,100 |
|         |         |    |         |
|         |         |    |         |
|         |         | 合計 | 168,750 |

川崎株式会社  
〒200-1111  
神奈川県横浜市 2-2-200

【 図 1 4 】

1400

ファイル名条件設定

1401

☒ ファイル名を自動入力する

1402

ファイル名の区切り文字

アンダースコア

1403

初期ファイル名

1404

保存

10

20

【 図 1 5 】

```
{
  "imageWidth": 2490,
  "imageHeight": 3515,
  "regions": [
    {
      "rect": {
        "x": 1019,
        "y": 303,
        "width": 489,
        "height": 95
      },
      "text": "見 積 書"
    },
    {
      "rect": {
        "x": 261,
        "y": 446,
        "width": 243,
        "height": 43
      },
      "text": "〒100-9999"
    },
    {
      "rect": {
        "x": 1584,
        "y": 446,
        "width": 262,
        "height": 36
      },
      "text": "見積番号:"
    },
    {
      "rect": {
        "x": 1874,
        "y": 443,
        "width": 230,
        "height": 47
      },
      "text": "R12-3456"
    }
  ]
}
```

※文字領域の数分 "regions" に要素が追加される

【 図 1 6 】

```
{
  "matched": true,
  "formId": "aaaaaaa-ffff-49ab-acf8-55558888eeee",
  "matchingScore": 0.74582005269911589,
  "rectInfoArray": [
    {
      "key": "fileRegion0",
      "region": {
        "rect": {
          "x": 1017,
          "y": 302,
          "width": 489,
          "height": 94
        },
        "text": "見 積 書"
      }
    },
    {
      "key": "fileRegion1",
      "region": {
        "rect": {
          "x": 386,
          "y": 626,
          "width": 624,
          "height": 70
        },
        "text": "下丸子株式会社"
      }
    }
  ],
  "metadataArray": [
    {
      "key": "filename",
      "keyType": "filename",
      "value": [
        "fileRegion0",
        "separator",
        "fileRegion1"
      ]
    }
  ]
}
```

30

40

【 図 1 7 】

# 見 積 書

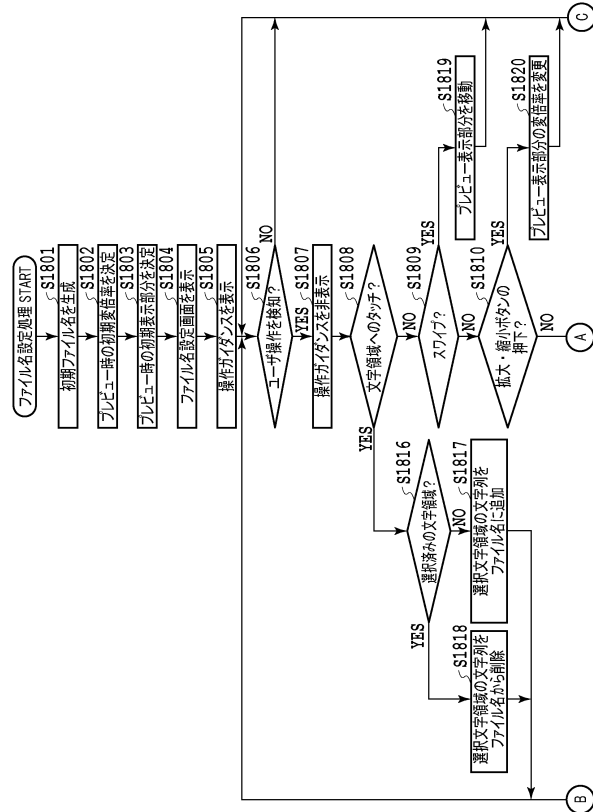
〒100-9999  
東京都大田区 A-A-A  
下丸子株式会社 御中

見積番号 :R12-3500  
発行日 :2017/09/29

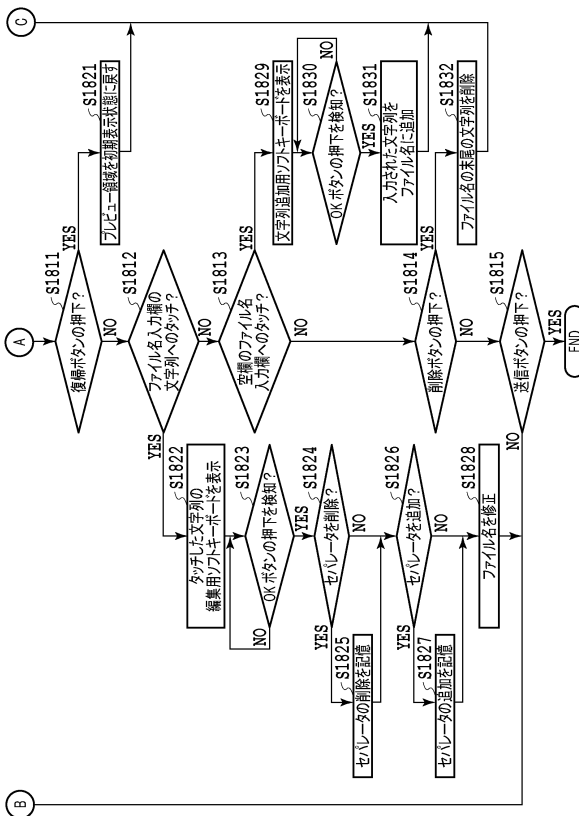
| 品目      | 単価     | 数量 | 金額     |
|---------|--------|----|--------|
| GF-1555 | 10,000 | 2  | 20,000 |
|         |        |    |        |
|         |        |    |        |
|         |        |    |        |
|         |        | 合計 | 20,000 |

川崎株式会社  
〒200-1111  
神奈川県横浜市 2-2-200

【 図 1 8 】



【 図 19 】



【 図 2 0 】

初期状態で  
プレビュー領域に  
表示する部分

## 見 積 書

〒100-9999

東京都港区 B-B-B

品川株式会社

御中

見積番号 :R12-3456

発行日 :2017/09/10

| 品目      | 単価      | 数量 | 金額      |
|---------|---------|----|---------|
| GF-1544 | 11,550  | 3  | 34,650  |
| EF24-70 | 134,100 | 1  | 134,100 |
|         |         |    |         |
|         |         | 合計 | 168,750 |

川崎株式会社

〒200-1111

神奈川県横浜市 2-2-200

【 図 2 1 】

1000

ファイル名設定画面

1001

川崎株式会社

1003

×

1002

合計

168,750

川崎株式会社

〒200-1111

神奈川県横浜市 2-2-200

1006

1004

1005

1007

送信

【 図 2 2 】

2200

ソフトウェア

2201

川崎株式会社

2202

キャンセル

OK

10

【 図 2 3 】

1000

ファイル名設定画面

1001

2301

1003

×

1002

プレビューの文字をタッチするとファイル名を入力できます。

〒100-9999

東京都港区 B-B-B

品川株式会社 御中

見積番号 :R12-3456

発行日 :2017/09/10

1006

1004

1005

1007

送信

【 図 2 4 】

1000

ファイル名設定画面

1001

見積書 下丸子株式会社

1003

×

1002

見積書

〒100-9999

東京都大田区 A-A-A

下丸子株式会社 御中

見積番号 :R12-3500

発行日 :2017/09/29

1006

1004

1005

1007

送信

20

30

40

50

初稿状態で  
プレビュー欄に  
表示する部分へ

|                                    |         |       |         |                                   |  |
|------------------------------------|---------|-------|---------|-----------------------------------|--|
| 〒100-9999<br>東京都港区B-B<br>品川株式会社 御中 |         | 見 積 書 |         | 見積番号: R12-3456<br>発行日: 2017/09/10 |  |
| 品目                                 | 単価      | 数量    | 金額      |                                   |  |
| GF-1544                            | 11,550  | 3     | 34,650  |                                   |  |
| EF24-70                            | 134,100 | 1     | 134,100 |                                   |  |
|                                    |         |       |         |                                   |  |
|                                    |         |       |         |                                   |  |
|                                    |         | 合計    | 168,750 |                                   |  |

川崎株式会社  
〒200-1111  
神奈川県横浜市 2-2-200

---

フロントページの続き

- (56)参考文献 特開 2 0 1 9 - 0 4 0 2 5 0 ( J P , A )  
特開 2 0 1 5 - 2 1 5 8 7 8 ( J P , A )  
特開 2 0 1 7 - 1 5 1 7 9 3 ( J P , A )  
特開 2 0 0 5 - 1 5 7 4 5 8 ( J P , A )  
特開平 0 6 - 2 5 1 1 8 5 ( J P , A )
- (58)調査した分野 (Int.Cl. , D B 名)
- |         |               |
|---------|---------------|
| G 0 6 F | 3 / 0 4 8 8 6 |
| G 0 6 F | 3 / 0 4 8 4 2 |
| G 0 6 F | 1 6 / 5 8 3   |
| H 0 4 N | 1 / 0 0       |
| G 0 6 F | 1 6 / 1 4     |