

(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2012-134847

(P2012-134847A)

(43) 公開日 平成24年7月12日(2012.7.12)

(51) Int.Cl.		F I				テーマコード (参考)
HO4N	1/41	(2006.01)	HO4N	1/41	C	5C159
HO4N	7/26	(2006.01)	HO4N	7/13	Z	5C178

審査請求 未請求 請求項の数 6 O L (全 22 頁)

(21) 出願番号 特願2010-286311 (P2010-286311)
 (22) 出願日 平成22年12月22日 (2010.12.22)

(71) 出願人 000006747
 株式会社リコー
 東京都大田区中馬込1丁目3番6号
 (74) 代理人 100084250
 弁理士 丸山 隆夫
 (72) 発明者 青木 真路
 東京都大田区中馬込1丁目3番6号 株式会社リコー内
 Fターム(参考) 5C159 MA00 MA04 MA23 MC01 MC11
 MC38 ME02 PP01 PP15 PP16
 PP20 SS26 SS28
 5C178 AC10 AC12 AC14 BC01 BC02
 BC06 BC91 CC59 CC60 EC64

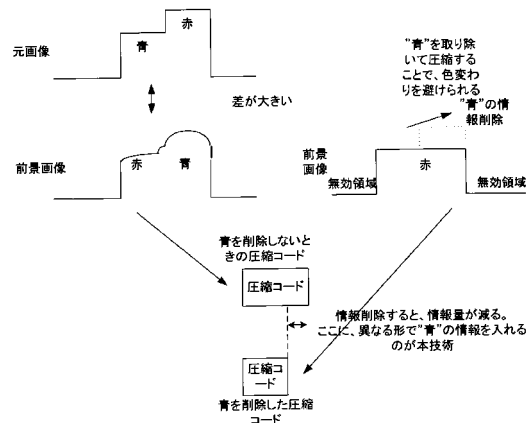
(54) 【発明の名称】 画像圧縮装置、画像圧縮方法、及びプログラム

(57) 【要約】

【課題】 文字の色変わりをプレビュー画面上で改善する。

【解決手段】 異なる複数の色彩から構成される画像を矩形型の画像領域に区分して圧縮する画像圧縮装置であって、第1の色彩情報を有する矩形型の画像領域を特定する第1の画像領域特定手段、第1の色彩情報を有する矩形型の画像領域に隣接する第2の色彩情報を有する矩形型の画像領域を特定する第2の画像領域特定手段を有し、第1の圧縮コード及び第2の圧縮コードを生成し、第1の圧縮コードと第2の圧縮コードのデータ量の和である総圧縮コードから、第2の圧縮コードのデータ量を減じ、第2の色彩情報に基づいて、第2の色彩情報を識別する情報であるベクトル情報を生成し、減ぜられた第2の圧縮コードのデータ量の部分に、ベクトル情報を格納し、圧縮画像情報を生成する。

【選択図】 図1 2



【特許請求の範囲】

【請求項 1】

異なる複数の色彩から構成される画像を矩形型の画像領域に区分して圧縮する圧縮手段を有する画像圧縮装置であって、

複数の文字画像の中から第 1 の色彩情報を有する矩形型の画像領域を特定する第 1 の画像領域特定手段と、

前記第 1 の画像領域特定手段により特定された前記第 1 の色彩情報を有する矩形型の画像領域に隣接する前記複数の文字画像の中から第 2 の色彩情報を有する矩形型の画像領域を特定する第 2 の画像領域特定手段と、を有し、

前記圧縮手段は、

第 1 の圧縮コードを生成する第 1 の圧縮コード生成手段と、

第 2 の圧縮コードを生成する第 2 の圧縮コード生成手段と、

前記第 1 の圧縮コードのデータ量と前記第 2 の圧縮コードのデータ量との和である総圧縮コードのデータ量から、前記第 2 の圧縮コードのデータ量に相当するデータ量を減ずるデータ量削減手段と、

前記第 2 の色彩情報に基づいて、前記第 2 の色彩情報を識別する情報であるベクトル情報を生成するベクトル情報生成手段と、

前記減ぜられた前記第 2 の圧縮コードのデータ量に相当するデータ量の部分に、前記ベクトル情報生成手段により生成された前記ベクトル情報を格納するベクトル情報格納手段と、

前記第 1 の圧縮コードのデータの中に前記ベクトル情報を格納した圧縮画像情報を生成する圧縮画像情報生成手段と、

を備えたことを特徴とする画像圧縮装置。

【請求項 2】

前記圧縮画像情報をプレビュー表示する表示手段を設け、前記第 2 の画像領域特定手段は、前記表示手段にプレビュー表示される画像に対する領域指定を行うことにより、前記第 2 の色彩情報を有する矩形型の画像領域が特定されることを特徴とする請求項 1 に記載の画像圧縮装置。

【請求項 3】

前記第 2 の色彩情報を有する矩形型の画像領域は、前記第 1 の色彩情報を有する矩形型の画像領域に隣接する上下左右の領域の中から特定されることを特徴とする請求項 1 又は 2 に記載の画像圧縮装置。

【請求項 4】

各々が異なる複数の色彩情報を有する複数の矩形型の画像領域と、これら複数の画像領域の前記複数の色彩情報を識別する情報である複数のベクトル情報とを画像領域情報群として予め用意しておき、前記領域指定がなされたときに、前記画像領域情報群の中から、前記第 2 の色彩情報を有する矩形型の画像領域が特定されることを特徴とする請求項 2 又は 3 に記載の画像圧縮装置。

【請求項 5】

異なる複数の色彩から構成される画像を矩形型の画像領域に区分して圧縮する圧縮手段を有する画像圧縮装置における画像圧縮方法であって、

第 1 の画像領域特定手段により、複数の文字画像の中から第 1 の色彩情報を有する矩形型の画像領域を特定する工程と、

第 2 の画像領域特定手段により、前記第 1 の画像領域特定手段により特定された前記第 1 の色彩情報を有する矩形型の画像領域に隣接する前記複数の文字画像の中から第 2 の色彩情報を有する矩形型の画像領域を特定する工程と、を有し、

前記圧縮手段は、

第 1 の圧縮コード生成手段により、第 1 の圧縮コードを生成する工程と、

第 2 の圧縮コード生成手段により、第 2 の圧縮コードを生成する工程と、

データ量削減手段により、前記第 1 の圧縮コードのデータ量と前記第 2 の圧縮コードの

10

20

30

40

50

データ量との和である総圧縮コードのデータ量から、前記第2の圧縮コードのデータ量に相当するデータ量を削減する工程と、

ベクトル情報生成手段により、前記第2の色彩情報に基づいて、前記第2の色彩情報を識別する情報であるベクトル情報を生成する工程と、

ベクトル情報格納手段により、前記減ぜられた前記第2の圧縮コードのデータ量に相当するデータ量の部分に、前記ベクトル情報生成手段により生成された前記ベクトル情報を格納する工程と、

圧縮画像情報生成手段により、前記第1の圧縮コードのデータの中に前記ベクトル情報を格納した圧縮画像情報を生成する工程と、

を備えたことを特徴とする画像圧縮方法。

10

【請求項6】

異なる複数の色彩から構成される画像を矩形型の画像領域に区分して圧縮する圧縮手段を有する画像圧縮装置のコンピュータに実行させるプログラムであって、

第1の画像領域特定手段により、複数の文字画像の中から第1の色彩情報を有する矩形型の画像領域を特定する処理と、

第2の画像領域特定手段により、前記第1の画像領域特定手段により特定された前記第1の色彩情報を有する矩形型の画像領域に隣接する前記複数の文字画像の中から第2の色彩情報を有する矩形型の画像領域を特定する処理と、を有し、

前記圧縮手段は、

第1の圧縮コード生成手段により、第1の圧縮コードを生成する処理と、

第2の圧縮コード生成手段により、第2の圧縮コードを生成する処理と、

データ量削減手段により、前記第1の圧縮コードのデータ量と前記第2の圧縮コードのデータ量との和である総圧縮コードのデータ量から、前記第2の圧縮コードのデータ量に相当するデータ量を削減する処理と、

ベクトル情報生成手段により、前記第2の色彩情報に基づいて、前記第2の色彩情報を識別する情報であるベクトル情報を生成する処理と、

ベクトル情報格納手段により、前記減ぜられた前記第2の圧縮コードのデータ量に相当するデータ量の部分に、前記ベクトル情報生成手段により生成された前記ベクトル情報を格納する処理と、

圧縮画像情報生成手段により、前記第1の圧縮コードのデータの中に前記ベクトル情報を格納した圧縮画像情報を生成する処理と、

を備えたことを特徴とするプログラム。

20

30

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、画像圧縮装置、画像圧縮方法、及び画像圧縮プログラムに関する。

【背景技術】

【0002】

見た目上は1つの画像を、複数の画像を合成することで生成する技術がある。例えば、文字の形状を表す文字画像、絵柄を表す背景画像、文字の色を表す前景画像、黒文字の位置を示す黒文字画像により1つの合成画像を形成する技術がある。すなわち、合成された画像上の一画素の情報は、1つ以上の画像情報を持つこととなり、画素位置によって、どの画像が情報を持つかが異なる。文字画像、背景画像、前景画像、黒文字画像は1つのスキャナ画像から生成されるが、各画像に分解することで高圧縮なファイルを生成することができる。

40

【0003】

上記合成画像を形成する画像の中で、前景画像は、ブロック単位のJPEG (Joint Photographic Experts Group) 圧縮を行うことが行われている。解像度を落として異なる色が多く含まれるブロックを圧縮すると、それを復元した際に、元の色が失われることがしばしばある。このような文字の色変わりを修正するために、生成した画像を、一度画像処

50

理装置に備え付けられたモニタ（タッチパネル）でプレビュー表示し、ユーザがモニタ上で修正を指示することができる技術がある。

【発明の概要】

【発明が解決しようとする課題】

【0004】

特許文献1には、スキャナ画像から各構成画像を生成する場合に生じる色の混在や間違いを修正する手段として、ユーザが意図する通りに文字領域の色を設定する目的で、文字領域を抽出して文字領域に対応する画像を生成し、生成した画像を1つの構成要素とする合成画像をプレビュー表示させ、文字領域の色を指定色に変換するための手段を持ち、文字領域を、指定色に変換する技術が開示されている。

10

【0005】

しかしながら、今までの技術は、文字領域の形状を表す複数の画像と、文字領域それぞれに対して、文字の色を1色で表現するファイル構成に関するものであり、文字の色を画像（前景画像）で表現する場合にそのまま適応できないという問題があった。したがって、文字の判読性を改善することができないという問題は依然として解消されていない。

【0006】

そこで本発明は、上記問題点に鑑みてなされたもので、文字の色を画像で表現するファイル作成方法において、文字の色変わりをプレビュー画面上で改善することができる画像圧縮装置、画像圧縮方法、及びプログラムを提供することを目的とする。

20

【課題を解決するための手段】

【0007】

上記課題を解決するため、請求項1に記載の本発明における画像圧縮装置は、異なる複数の色彩から構成される画像を矩形型の画像領域に区分して圧縮する圧縮手段を有する画像圧縮装置であって、複数の文字画像の中から第1の色彩情報を有する矩形型の画像領域を特定する第1の画像領域特定手段と、前記第1の画像領域特定手段により特定された前記第1の色彩情報を有する矩形型の画像領域に隣接する前記複数の文字画像の中から第2の色彩情報を有する矩形型の画像領域を特定する第2の画像領域特定手段と、を有し、前記圧縮手段は、第1の圧縮コードを生成する第1の圧縮コード生成手段と、第2の圧縮コードを生成する第2の圧縮コード生成手段と、前記第1の圧縮コードのデータ量と前記第2の圧縮コードのデータ量との和である総圧縮コードのデータ量から、前記第2の圧縮コードのデータ量に相当するデータ量を減ずるデータ量削減手段と、前記第2の色彩情報に基づいて、前記第2の色彩情報を識別する情報であるベクトル情報を生成するベクトル情報生成手段と、前記減ぜられた前記第2の圧縮コードのデータ量に相当するデータ量の部分に、前記ベクトル情報生成手段により生成された前記ベクトル情報を格納するベクトル情報格納手段と、前記第1の圧縮コードのデータの中に前記ベクトル情報を格納した圧縮画像情報を生成する圧縮画像情報生成手段と、を備えたことを特徴とする。

30

【0008】

また、本発明における画像圧縮装置は、請求項1に記載の画像圧縮装置において、前記圧縮画像情報をプレビュー表示する表示手段を設け、前記第2の画像領域特定手段は、前記表示手段にプレビュー表示される画像に対する領域指定を行うことにより、前記第2の色彩情報を有する矩形型の画像領域が特定されることを特徴とする。

40

【0009】

さらに、本発明における画像圧縮装置は、請求項1又は2に記載の画像圧縮装置において、前記第2の色彩情報を有する矩形型の画像領域は、前記第1の色彩情報を有する矩形型の画像領域に隣接する上下左右の領域の中から特定されることを特徴とする。

【0010】

また、本発明における画像圧縮装置は、請求項2又は3に記載の画像圧縮装置において、各々が異なる複数の色彩情報を有する複数の矩形型の画像領域と、これら複数の画像領域の前記複数の色彩情報を識別する情報である複数のベクトル情報とを画像領域情報群として予め用意しておき、前記領域指定がなされたときに、前記画像領域情報群の中から、

50

前記第2の色彩情報を有する矩形型の画像領域が特定されることを特徴とする。

【0011】

そして、請求項5に記載の本発明における画像圧縮方法は、異なる複数の色彩から構成される画像を矩形型の画像領域に区分して圧縮する圧縮手段を有する画像圧縮装置における画像圧縮方法であって、第1の画像領域特定手段により、複数の文字画像の中から第1の色彩情報を有する矩形型の画像領域を特定する工程と、第2の画像領域特定手段により、前記第1の画像領域特定手段により特定された前記第1の色彩情報を有する矩形型の画像領域に隣接する前記複数の文字画像の中から第2の色彩情報を有する矩形型の画像領域を特定する工程と、を有し、前記圧縮手段は、第1の圧縮コード生成手段により、第1の圧縮コードを生成する工程と、第2の圧縮コード生成手段により、第2の圧縮コードを生成する工程と、データ量削減手段により、前記第1の圧縮コードのデータ量と前記第2の圧縮コードのデータ量との和である総圧縮コードのデータ量から、前記第2の圧縮コードのデータ量に相当するデータ量を削減する工程と、ベクトル情報生成手段により、前記第2の色彩情報に基づいて、前記第2の色彩情報を識別する情報であるベクトル情報を生成する工程と、ベクトル情報格納手段により、前記減ぜられた前記第2の圧縮コードのデータ量に相当するデータ量の部分に、前記ベクトル情報生成手段により生成された前記ベクトル情報を格納する工程と、圧縮画像情報生成手段により、前記第1の圧縮コードのデータの中に前記ベクトル情報を格納した圧縮画像情報を生成する工程と、を備えたことを特徴とする。

10

【0012】

また、請求項6に記載の本発明におけるプログラムは、異なる複数の色彩から構成される画像を矩形型の画像領域に区分して圧縮する圧縮手段を有する画像圧縮装置のコンピュータに実行させるプログラムであって、第1の画像領域特定手段により、複数の文字画像の中から第1の色彩情報を有する矩形型の画像領域を特定する処理と、第2の画像領域特定手段により、前記第1の画像領域特定手段により特定された前記第1の色彩情報を有する矩形型の画像領域に隣接する前記複数の文字画像の中から第2の色彩情報を有する矩形型の画像領域を特定する処理と、を有し、前記圧縮手段は、第1の圧縮コード生成手段により、第1の圧縮コードを生成する処理と、第2の圧縮コード生成手段により、第2の圧縮コードを生成する処理と、データ量削減手段により、前記第1の圧縮コードのデータ量と前記第2の圧縮コードのデータ量との和である総圧縮コードのデータ量から、前記第2の圧縮コードのデータ量に相当するデータ量を削減する処理と、ベクトル情報生成手段により、前記第2の色彩情報に基づいて、前記第2の色彩情報を識別する情報であるベクトル情報を生成する処理と、ベクトル情報格納手段により、前記減ぜられた前記第2の圧縮コードのデータ量に相当するデータ量の部分に、前記ベクトル情報生成手段により生成された前記ベクトル情報を格納する処理と、圧縮画像情報生成手段により、前記第1の圧縮コードのデータの中に前記ベクトル情報を格納した圧縮画像情報を生成する処理と、を備えたことを特徴とする。

20

30

【発明の効果】

【0013】

本発明によれば、プレビュー画面において、文字の色変わりを指定された領域に対応する前景画像上のブロックを、そのブロック内に含まれる色を減少させて圧縮を行うことにより、そのブロックの圧縮コード量を低減させて、当該減少させた色を、当該減少させた色と近い当該ブロックに隣接する色のベクトル情報として、当該低減させた圧縮コードの部分に格納すると同時に、減少させた色に対応する画素を識別する情報を格納するようにしたので、文字の色を画像（前景画像）で表現する技術において、プレビュー画面上で、文字の色変わりを改善することができる画像圧縮装置、画像圧縮方法、及びプログラムを得ることができる。

40

【図面の簡単な説明】

【0014】

【図1】本発明の実施形態における画像処理装置の概略構成を示すブロック図である。

50

【図 2】本発明の実施形態における高圧縮 P D F のフォーマット処理について説明する図である。

【図 3】本発明の実施形態における色変わりについて説明する図である。

【図 4】本発明の実施形態における色わりに付いて説明する図である。

【図 5】本発明の実施形態におけるデータの流れを示すブロック図である。

【図 6】本発明の実施形態における原画像の分離処理の概念図である。

【図 7】本発明の実施形態におけるコントローラの動作の流れを示すフロー図である。

【図 8】本発明の実施形態における指定範囲内の文字画素を含む前景画像のブロックに対応する処理対象画像情報を取得するイメージを示す図である。

【図 9】本発明の実施形態における修正文字画素候補の中から、修正するブロックを決定するフローを示すフロー図である。

10

【図 10】本発明の実施形態における処理対象のブロックが、さらに先のブロックで置き換えが可能であることを示す模式図である。

【図 11】本発明の実施形態における H D D に蓄積されている画素の状態を示す模式図である。

【図 12】本発明の実施形態における圧縮コードを削除した部分に修正画素情報を入れることを説明する図である。

【図 13】本発明の実施形態における J P E G 圧縮の基本方式の処理手順を示す図である。

【図 14】本発明の実施形態における J P E G 画像のジグザグスキャンのテーブルを示す図である。

20

【図 15】本発明の実施形態における J P E G 圧縮における A C 成分のハフマン符号表の一例を示す図である。

【図 16】本発明の実施形態におけるエントロピー符号化において、エントロピーが減ると、置き換え後のビットサイズが減ることを説明する図である。

【図 17】本発明の実施形態におけるエントロピー符号化において、減少したビット数の分だけ無効領域が増加することを説明する図である。

【図 18】本発明の他の実施形態における置き換える色を持つブロックへのベクトルを示す図である。

30

【発明を実施するための形態】

【0015】

次に、本発明を実施するための形態について図面を参照して詳細に説明する。なお、各図中、同一又は相当する部分には同一の符号を付しており、その重複説明は適宜に簡略化乃至省略する。

【0016】

本発明は、複数の画像を合成して表示する画像フォーマットに関し、文字の色を画像（前景画像）で表現する技術において、プレビュー画面上に表示された文字の色変わりに関して、指定された領域に対応する前景画像上のブロックを、そのブロック内に含まれる色のデータ量を減少させて圧縮を行うことにより、そのブロックの圧縮コード量を低減させると共に、データ量を減少させた色を、当該減少させた色と近い当該ブロックに隣接する色のベクトル情報として、上記低減させた圧縮コードの部分に格納すると同時に、減少させた色に対応する画素を識別する情報を格納することが特徴となっている。

40

【0017】

まず、図 1 を用いて、本発明における画像圧縮装置が搭載される画像処理装置について説明する。図 1 は、本発明の実施形態におけるデジタル式の画像処理装置の概略構成を示したブロック図である。カラー画像処理装置 1 が複写機として動作する場合、スキャナ 2 は原稿 3 から画像データを読み取り、当該画像データ（アナログ信号）をデジタルデータに変換して出力する。

【0018】

スキャナ補正部 4 は、後述するように、スキャナ 2 から出力された画像データ（デジタ

50

ルデータ)に対して、画像領域を文字、線画や写真等に分類したり、原稿画像のRGB (Red Green Blue) データに対してフィルタ処理等の画像処理を施す。

【0019】

圧縮処理部5は、スキャナ補正後のRGB各8Bit画像データとエッジ文字領域信号(1Bit)、色領域信号(1Bit)を圧縮処理して、汎用バス6にデータを送出する。圧縮後の画像データは、汎用バス6を通過して、コントローラ7に送られる。コントローラ7は図示しない半導体メモリを持ち、送られたデータを蓄積するようになっている。蓄積データには、書誌情報として画像サイズや読み取った原稿の種類も記録する。

【0020】

なお、ここでは画像データに対して圧縮処理を行うとしたが、汎用バス6の帯域が十分に広く、蓄積するHDD (Hard Disk Drive) の容量が大きければ、非圧縮の状態でも画像データを扱っても良い。

【0021】

次に、コントローラ7は、HDDの画像データを、汎用バス6を介して伸張処理部8に送出する。伸張処理部8は、圧縮処理されていた画像データを元のRGB各8Bitデータとエッジ文字領域信号(1Bit)、色領域信号(1Bit)に伸張し、プリンタ補正部9に送出する。

【0022】

プリンタ補正部9では、RGB画像データを色補正処理部にてYMCBk (Yellow Magenta Cyan Black) データに変換して、エッジ文字領域信号であり色領域信号でない部分は、黒文字としてBkの単色データに置き換える。さらに、補正処理、中間調処理等が行われ、プロッタ10の明暗特性の補正処理や階調数変換処理を行う。

【0023】

ここでの階調数変換処理では、誤差拡散やディザ処理を用いて各色8Bitから2Bitへと画像データの変換を行う。プロッタ10は、レーザビーム書き込みプロセスを用いた転写紙印字ユニットで、2Bitの画像データを感光体に潜像として描画し、トナーによる作像/転写処理後、転写紙にコピー画像11を形成する。

【0024】

ネットワーク12を介して外部PC端末13に画像データを配信する配信スキャナとして動作する場合は、コントローラ7がHDDに蓄積された画像データに対して、色変換処理、階調処理、フォーマット処理等を行う。

【0025】

階調処理では配信スキャナ動作時のモードに従った階調変換処理を行う。フォーマット処理では、JPEGやTIFF (Tagged Image File Format) やPDF (Portable Document Format) 形式への汎用画像フォーマット変換等を行う。設定によっては、高圧縮PDF画像の画像データを生成して、それらの画像に対してPDFヘッダを付加することとなる。その後、画像データはNIC (Network Interface Controller) 15を介して外部PC端末13に配信される。

【0026】

配信される画像データは、操作部16に備え付けられたプレビュー画像表示部によってプレビュー確認が可能となっている。プレビュー表示設定がなされている場合には、フォーマット処理で成形された画像データを基に、プレビュー表示用の画像をコントローラ7の内部で生成する。

【0027】

プレビュー画面は一般的に小さいので、プレビュー表示用に縮小処理を行う。プレビュー表示用に生成された画像データは操作部16に転送される。操作部16では転送された画像データを表示する。縮小処理は必須ではない。縮小することで画像データの損失が発生するため、画質重視の場合には縮小処理は行わない。

【0028】

プレビュー画像を確認することにより、このままの画像で良いと判断した場合には、所

10

20

30

40

50

定のボタンを押下することで、外部PC端末13に配信されることとなる。プレビュー画像を確認した結果、原稿が間違っていた等の理由で読み取りから再度やり直したい場合には、所定のボタンを押すことで、再度読み取りから開始されることとなる。

【0029】

プレビュー画像上の一部分を修正したいと判断した場合には、修正箇所を画面上で指定することにより、修正が可能である。操作部16に備え付けられたプレビュー画像表示部は、タッチパネル形式となっており、タッチパネルにかけられた圧力の位置情報を画像処理装置が認識することが可能になっている。そのため、ユーザが修正を指定した箇所を画像処理装置が認識するようになっている。この修正に関わる処理については、本発明の核となる部分であるので、後述する。

10

【0030】

また、画像処理装置が、ネットワーク12を介して外部PC端末13からプリントアウトするプリンタとして動作する場合、NIC15を介して受信するデータから、画像及びプリント指示を行うコマンドを解析し、画像データとして、印刷可能な状態にビットマップ展開し、展開されたデータを圧縮してデータを蓄積する。蓄積されたデータは随時大容量の記憶装置であるHDD14に書き込まれる。画像データを蓄積する時に、後述する書誌情報もHDD14に書き込む。

【0031】

次にコントローラ7は、HDD14の画像データを、汎用バス6を介して伸張処理部8に送出する。伸張処理部8は圧縮処理されていた画像データを元の8Bitデータに伸張し、プリンタ補正部9に送出する。プリンタ補正部9では、RGB入力ならば、色補正処理部にてYMcBkデータに変換をする。

20

【0032】

次にYMcBkそれぞれ独立に補正処理、中間調処理等が行われ、プロッタ10の明暗特性の補正処理や階調数変換処理を行う。ここでの階調数変換処理では、誤差拡散やディザ処理を用いて8Bitから2Bitへと画像データの変換を行う。プロッタ10はレーザビーム書き込みプロセスを用いた転写紙印字ユニットで、2Bitの画像データを感光体に潜像として描画し、トナーによる作像/転写処理後、転写紙にコピー画像を形成する。

【0033】

デジタル画像処理装置においては、一般に原稿をスキャナにより読み取り、画像データをデジタルデータに変換するとともに、原稿の画像領域(像域)を、異なる特徴を有する領域に分類(像域分離)する。注目画素が、分類された画像領域(像域)の何れの領域に属するものが判定された結果にしたがい、画像データに対して種々の画像処理を施す。これにより、出力画像の画像品質を大きく向上することができる。操作部16からは、ユーザの動作に関わるモードの設定やスタートの指示を行う。設定を行い、スタートの指示をすることで、上記説明した動作が開始される。

30

【0034】

次に、本発明の実施形態における高圧縮PDF画像圧縮装置について説明する。図2は、本発明の実施形態における高圧縮PDFのフォーマット処理について説明する図である。高圧縮PDFは、文字画像、前景画像、背景画像、黒文字画像から構成されている。

40

【0035】

文字画像は、文字/非文字の2値で構成され、文字の形状、位置を示す。前景画像は、文字画像で文字となっている画素の色情報を持つ。背景画像は、文字画像で非文字となった箇所についての情報を持つ。黒文字画像は、処理対象画像の黒い文字の形状、位置を示す。

【0036】

背景画像及び前景画像において無効領域と記した箇所は、情報を持つ必要のない領域を示す。例えば、文字画像で文字とされている画素の色は前景画像が持っているので、背景画像において対応する領域は無効領域となる。高圧縮PDFを生成する際のフォーマット

50

処理では、上記4つの画像を生成した上で、それぞれを個別圧縮する。

【0037】

次に、前景画像の色変わりについて図3を用いて説明する。図3は、本発明の実施形態における色変わりについて説明する図である。図2では明示していないが、前景画像は、解像度を落としてからブロック圧縮を行う。解像度を落とすときに、ブロック内に異なる色の文字が含まれていた場合、情報が失われることとなる。その結果、ユーザには文字の色変わりとして知覚される場合がある。

【0038】

具体的には、図3において、解像度を落とす前の画像として、赤い文字列だけの画像部分Aと、赤い文字列と青い文字列との境界面における画像部分Bとからなる画像について解像度を落としてからブロック圧縮を行うと、赤い文字列だけの画像部分Aを縮小処理しても、赤い文字列だけの画像が再現されるが、赤い文字列と青い文字列との境界面における画像部分Bを縮小処理すると、赤と青が混ざってしまい、ユーザにとっては色変わりとして認識される場合がある。

10

【0039】

同様に、図4を用いて前景画像の色変わりに付いて説明する。図4は、本発明の実施形態における色変わりに付いて説明する図である。図4において、処理対象画像が、赤と青からなる文字画像であった場合、その文字画像の文字となっている画素の色情報を持つ前景画像は、斜線で表した部分となり、この画像の解像度を落としてからブロック圧縮を行った後に展開すると、赤色の情報と青色の情報との境界面では、赤と青が混合し、文字画像と合成して表示すると、ユーザにとっては色変わりとして知覚されてしまう。

20

【0040】

次に、図5を用いて、本発明の実施の形態におけるコントローラ7(図1)とHDD14(図1)間のデータの流れについて説明する。図5は、本発明の実施形態におけるデータの流れを示すブロック図である。図5において、出力画像生成部51と、出力画像組立部52は、上述したフォーマット処理の中にあるものである。

【0041】

出力画像生成部51では、処理対象画像から背景画像、文字画像、前景画像、黒文字画像を生成する。如何なるアルゴリズムを使用するかは不問であるが、ここでは、図6に示すアルゴリズムに基づく分離処理を用いるものとする。図6は、本発明の実施形態における原画像の分離処理の概念図である。

30

【0042】

図6において、まず、ステップ(以下、「S」という。)10において、取得された原画像の各画素を、例えば明度(画像がRGBで表現されている場合は、G成分で代用してもよい)に係る閾値で分け、該閾値より明るい画素を背景(第三属性)に分類し、該閾値より暗い画素を前景(第一属性)に分類する。

【0043】

S11では、前景(第一属性)に分類された画素のうち、黒に非常に近い画素値の画素(例えば、R、G、B各色が全てある一定の閾値以下の画素)を選び、該画素を黒文字(第二属性)に分類し直す。

40

【0044】

各属性に基づいて、つまり、第一属性に基づいて生成された第一画像が、例えば図6のAであり、第二属性に基づいて生成された第二画像が、例えば図6のCであり、第三属性に基づいて生成された第三画像が、例えば図6のBである。

【0045】

S12では、生成された第一画像の所属画素(第一画像のうち、画素値が設定されている画素)の位置を参照し、該所属画素と同一位置の画素の画素値をON(例えば、1)に設定し、該画素以外の画素の画素値をOFF(例えば、0)に設定した選択データを生成する。生成された選択データが、例えば図6のDである。

【0046】

50

S 1 3では、第一画像の非可視領域の画素値が設定され、第三画像の非可視領域の画素値が設定される。非可視領域の画素値を設定された第一画像が、例えば図6のEであり、例えば、図6のFである。

【0047】

図6に示されるように、原画像に含まれる特定の色又は明度に係る領域（例えば、黒文字）を、独立した画像（レイヤー）として原画像より分離することにより、該特定の色又は明度に係る単独の色又は明度（黒又は明度ゼロ）で再現（出力）することができる。

【0048】

図5に戻り、本発明においては、図6に示した選択データDが文字画像であるものとする。図6のアルゴリズムを用いて生成された4つの画像C、D、E、Fは、一度HDD14（図1）に格納される。

10

【0049】

次に、プレビュー画像生成部53で、プレビュー用の画像を生成する。HDD14（図1）に蓄積された4つの画像を展開して、必要に応じてそれぞれの画像に対して縮小処理や拡大処理を行い、合成することでプレビュー用の画像を生成する。縮小処理や拡大処理の方法は問わないが、生成元の4つの画像上の各画素と、生成されたプレビュー用画像の各画素との対応関係は、保持しておくようにする。

【0050】

出力画像修正部54では、ユーザによって指定された修正領域から、修正画素を決定し、修正画素の表示上の色を修正するよう出力画像を編集する処理を行う。修正画素に対応する文字画像の画像データと、修正画素に対応する前景画像上の画素を含むブロックの画像データ及びブロック近隣の前景画像データ、及び蓄積されている処理対象画像データから、修正を加えた前景画像、修正画素を識別するための黒文字画像を生成する。出力画像修正部54の処理については、本発明の核となる部分であるので、後述する。

20

【0051】

出力画像組立部52は、出力画像修正部54で出力した前景画像と、蓄積済みの文字画像、黒文字画像、背景画像を1つのPDFファイルに纏める。なお、ユーザが修正を指示しない場合には、出力画像修正部54は通らずに、出力画像生成部53で出力された画像が出力画像組立部52に流れ、PDFファイルが生成されることとなる。

【0052】

次に、図7を用いて、プレビュー画像を表示した直後に、コントローラ7（図1）で動作する処理について説明する。図7は、本発明の実施形態におけるコントローラの動作の流れを示すフロー図である。上述したように、プレビュー画像の所定操作で、ユーザは、プレビュー画像の処理の再開、処理のやり直し、一部分の修正について選択することが可能である。ここでは、一部分の修正を選択した場合のフローを記している。

30

【0053】

まず、S71において、ユーザの不満箇所を特定する。すなわち、プレビュー画面上で、どこを修正したいかの指定を受け付ける。ユーザは、タッチパネルをペンでなぞる等の所定の方法で、修正箇所の範囲指定をする。画像処理装置は、その指定範囲から、画像上のどこが修正範囲かを把握する。

40

【0054】

次に、S72において、当該指定範囲内に文字画素が含まれているか否かを判断する。すなわち、修正範囲内に文字画素が含まれているか否かを調べる。文字画素とは、文字画像で文字とされている画素を意味する。本発明は、文字画素の色を修正することを目的としているので、修正範囲内に文字画素が含まれていない場合には、修正はできないので、その旨の表示を行う（S72：N）。

【0055】

プレビュー用の画像と文字画像の対応関係は、上述したようにプレビュー用画像生成時に保存してあるので、HDD14（図1）から指定修正範囲に対応する文字画像を読み込むことで指定修正範囲内に文字画素が含まれているか否かを調べることができる。

50

【0056】

文字画素が含まれていた場合は(S72:Y)、対応する前景画像のブロックもHDD14(図1)から読み込む。前景画像のブロックは、JPEGであれば8×8が単位である。プレビュー用の画像と前景画像の対応関係は、上述したように、プレビュー用画像生成時に保存してあるため、これらの情報から情報の取得が可能である。

【0057】

次に、S73において、処理対象となる画像情報を取得する。すなわち、指定した範囲内の文字画素を含む前景画像のブロックに対応する処理対象画像の情報を取得する。この処理対象となる画像情報を取得するイメージを図8に示す。図8は、本発明に実施形態における指定範囲内の文字画素を含む前景画像のブロックに対応する処理対象画像情報を取得するイメージを示す図である。

10

【0058】

図8において、プレビュー画面上において、修正を希望する画像範囲を指定すると、当該指定範囲を含む前景画像のブロックは、対応する処理対象画像の領域である4つの矩形型のブロック1から4に対応し、これらのブロックには、「あいう」という文字画素が含まれている。

【0059】

図7に戻り、S74において、修正対象となる文字画素が決定される。指定範囲内の文字画素に対応する画素について、処理対象画像と前景画像との差分を計算する。この差分が大きい画素を、指定範囲内の修正画素候補とする。

20

【0060】

修正範囲内の文字であっても、必ずしもその中のすべての画素を修正したいという意図があるわけではない。範囲の指定はユーザが入力するため、修正の必要のない文字も含まれている可能性もある。また、文字列に含まれる文字の輪郭の色を指定したい場合には、その文字列を囲むように領域指定するのがユーザにとって自然であるためである。こうした場合、修正の必要のない文字の中部も修正範囲に含まれてしまう。そのため、指定された領域を基に、修正文字画素を決定する必要がある。実際、文字の色変わりが気になるのは、異なる色の文字と同一ブロックに存在する文字が多いため、文字の輪郭部分が多い。

【0061】

また、この場合、修正画素ではなく、修正画素候補にしておくのは次の理由による。すなわち、

30

本発明の性質上、ブロック内の修正文字画素の数、ブロック内の画素位置に、いくつかの制約がある。そのため、処理対象画像と前景画像との差分が大きいすべての画素を修正できるわけではない。そのため、以後述べるフローにおいて最終的に修正する文字画素を決定する。

【0062】

ここで、本発明における修正に関する制約について説明する。本発明においては、修正画素の色は、近隣のブロックの或る1色となる。後述するが、修正画素の色を、同一ブロック内の修正画素以外の色に置き換えてブロックを圧縮し、圧縮したブロックの高周波成分の領域に、元の修正画素の色を符号化する。ブロック内には、修正画素ではない画素も含まれているため、ブロック内で、どれが修正画素かを知る必要がある。文字画像で文字となっている画素に関しては、黒文字画像で非文字にしているため、修正画素の黒文字画像の画素を黒文字にすることで、識別が可能である。すなわち、出力画像を組み立てる時に、文字画像が文字かつ黒文字画像が非黒文字の場合には、修正画素ではなく、文字画像が文字かつ黒文字画像が黒文字の場合には修正画素であるとするにより識別が可能になる。

40

【0063】

黒文字画像は2値のデータであるので、基本的には修正画素であるかないかの2値のみ情報を持つことができる。圧縮したブロックの高周波成分の中には、修正画素を置き換える色を持つブロックへのベクトル情報を持つ必要がある。同一ブロック内に、異なる色に

50

置き換える修正画素を持つためには、複数のブロックへのベクトル情報と、修正画素のブロック上の位置までも持つ必要が生じ、ファイルサイズの観点からは現実的ではない。そのため、基本的には1つのブロックに対して、1つの色に置き換える修正画素を決定することとなる。

【0064】

また、置き換える色を含むブロックも、ブロック自体は、縦横数十画素に対応する小さい矩形型のものであるため、そのブロック内に大きく異なる色の文字が含まれていることは想定していない。

【0065】

図7に戻り、S75においては、類似色ブロックを検索する。すなわち、修正文字画素候補から、修正文字画素を決定するステップである。修正文字画素候補を以下に述べる方法でクラスタに分類し、修正すると決定したクラスタの修正文字画素の色を置き換えるブロックとして決定する。この修正文字画素候補の中から、修正するブロックを決定するフローについて図9を用いて説明する。

10

【0066】

図9は、本発明の実施形態における修正文字画素候補の中から、修正するブロックを決定するフローを示すフロー図である。上述したように、1つのブロックに含まれる画素を置き換える色は、基本的には1色である。そのため、修正画素候補をクラスタリングして、1つ以上のクラスタに分け、最も修正すべきクラスタ内の修正画素候補を修正画素とする。

20

【0067】

図9において、S91では、修正画素候補をクラスタリングする。クラスタリング方法に関しては問わないのでここでは詳述しない。似たような色のグループを同一クラスタにすればよい。

【0068】

S92では、クラスタの代表値を算出する。クラスタの修正画素候補の代表値として、修正画素に対応する前景画像データの平均値を算出する。

【0069】

S93では、近隣ブロックの代表値を算出する。各クラスタに対して、近隣するブロックに近い色があるか否かを探す。近隣するブロックの文字画素に対応する前景画像データの平均値を算出する。

30

【0070】

S94では、近隣ブロックを選定する。S92, S93で算出した値を比較して、最も差が小さい近隣ブロックを選ぶ。ただし、所定の閾値よりも差が大きい場合には、近隣に似た色を持つブロックがないとして、出力しないこととする。ここでは、1ブロック以内のブロックを近隣ブロックとして扱うが、必ずしもこれに限られるものではない。しかしながら、遠く離れたブロックまで見に行くようにすると、後に格納することとなるベクトルの情報量が増えるため、好ましくない。

【0071】

S95では、クラスタがあるか否かを判断する。ただし、S94の出力が1つもない場合には、修正可能な色が見当たらなかったことを意味するので(S95:N)、S98に進む。

40

【0072】

S96では、クラスタの選定を行う。S94から出力があったクラスタのうち、修正画素候補の最も多いクラスタを選定する。このクラスタに含まれる修正文字画素候補を、修正文字画素とする。

【0073】

S97では、ベクトル算出を行う。修正画素が含まれるブロックからS96で選定したクラスタへのベクトルを算出する。

【0074】

50

S 9 8では、連続判断を行う。すなわち、S 9 4で説明したように、近隣ブロックは1ブロックとしているため、2ブロック以上離れた箇所に置き換えたい色がある場合には、ベクトルとして指定ができない。ここでは、上又は左の近隣ブロックに、修正画素を持つブロックがある場合には、そのブロックが持つベクトルのさらに先のブロックで置き換えが可能であるか否かを判断する。図10にその模式図を示す。図10は、本発明の実施形態における処理対象のブロックが、さらに先のブロックで置き換えが可能であることを示す模式図である。さらに先のブロックで置き換えが可能であれば、上又は左の近隣ブロックも同様であることを示すために、上又は左の近隣ブロックも同じベクトルを持つと考えられる。

【0075】

なお、ここで、上又は左としているのは、一般的に左上から右下に向けてラスタ処理をするため、左及び上のブロックはすでに修正画素が決まっているためである。右や下からラスタ処理をしている場合には上又は左に限られないことはいうまでもない。

【0076】

図9に戻り、S 9 9では、ベクトル検出が可能であるか否かを判断する。すなわち、S 9 8で置き換えが可能かどうかを調査した結果、上又は左（右又は下）のいずれかの方向で置き換えが可能であれば（S 9 9：Y）、S 9 6以降で圧縮コードを計算していくことになる。ベクトル検出ができない場合（S 9 9：N）には、S 9 0において、このブロックに対する処理は終了となる。すなわち、S 9 0では、指定された領域には、修正可能な文字がないこと、すなわち、不可能表示を操作部に表示する。

【0077】

図7に戻り、S 7 6では、圧縮コードの修正を行う。つまり、修正画素が含まれるブロックの前景画像の圧縮コードを生成する。図11は、本発明の実施形態におけるHDDに蓄積されている画素の状態を示す模式図である。前景画像は、ブロック単位の圧縮を行うものであるため、ブロックごとに圧縮コードが生成されている。HDD（図1）14に蓄積されている前景画像では図11の上図に示すように、無効画素の部分には、ファイルサイズを小さくするために、それ以外の画素の平均値が入っている。

【0078】

図7のS 7 6では、無効画素と、修正画素の両方について、修正画素以外の文字画素（図11では、ア）の平均値に置き換えている。こうすることで、ブロック内が均一に近づき、図12に示すように、ブロックに対応する圧縮コード量が減る。減った部分に、修正画素を置き換える色情報を入れる。図12は、本発明の実施形態における圧縮コードを削除した部分に修正画素情報を入れることを説明する図である。この修正画素情報の入れ方については後述する。修正画素のブロック内の画素は、修正画素以外の文字画素の平均とする。

【0079】

同時に、ブロック内の文字画素の修正画素と、それ以外の画素を区別するために、修正画素に対応する黒文字画像の画素を黒文字にする。これは、通常、文字画素で文字となっている画素は、黒文字画像で黒文字となっていないためである。もし文字画素で文字となっている画素は、黒文字画像で黒文字となっているのが通常であれば、逆に修正画素に対応する黒文字画像の画素を非黒文字にする必要がある。

【0080】

次に、前景画像の修正画素を置き換える色情報の入れ方について説明する。前景画像はJ P E G圧縮されているものとする。J P E Gの場合、図13に示すように8×8のブロックがハフマン符号化される。図13は、本発明の実施形態におけるJ P E G圧縮の基本方式の処理手順を示す図である。

【0081】

J P E G圧縮は、図13に示すように、離散コサイン変換（D C T：Discrete Cosine Transform）の後に量子化をしたものをハフマン符号化して出力される。D C成分とA C成分は、別々にハフマン符号化することとなる。D C成分は、前のブロックとの差分値を

10

20

30

40

50

符号化して、AC成分はジグザグスキャン(図14)で読み取った成分のコードをハフマン符号化する。

【0082】

ハフマン符号化ではハフマン符号表に応じてビット符号列が定まる。ビット符号の調整データを作成するためには、ハフマン符号表から適したビット符号列を選択する必要がある。

【0083】

図14は、本発明の実施形態におけるJPEG画像のジグザグスキャンのテーブルを示す図である。AC成分は、図14に示すジグザグスキャンのテーブルで読んだ順番で、係数のビット数と、符号化するブロック以前の0のランレングスに基づいて、図15示したようなハフマン符号表に基づいて符号化される。図15は、本発明の実施形態におけるJPEG圧縮におけるAC成分のハフマン符号表の一例を示す図である。

10

【0084】

例えば、最も高周波な成分(図14の右下)の値の係数値が5であり、0のランレングスが0であった場合、5は2進表現で“101”であるから、図15のrw=3, run=0の箇所に記載されている“100”が選択される。符号は表に基づいて選択された“100”に5を表現する“101”が付加されて、最終的にこの係数に対する符号は、“100101”となる。

【0085】

ところで、修正画素に対応する画素を、その他の文字画素の平均値に置き換えたことで、修正画素が含まれるブロックの圧縮コードでは、高周波成分の値が低くなっているはずである。一番高い高周波成分の値をいくつか増減させても、元の画像よりは再現性が高くなる。本発明では、この高周波成分の領域に、情報を埋め込むものである。

20

【0086】

高周波成分の領域に埋め込むコードから、上下左右の4値を識別するため、表1に示すようなルールで、一番高い高周波成分の値を変える。例えば、ブロックの修正文字画素を、それに接する上のブロックの代表値に置き換える場合、前景画像ブロックの一番高い高周波成分の係数が5であるときには、5にもっとも近い4の倍数は4であるため、係数は4にして圧縮を行う。4の場合は、4を表す2進表現は、“100”であるため、この係数に対する圧縮コードは“100100”になる。

30

【表1】

置き換える色のブロック/置き換える色が同じブロック	一番高い高周波成分の係数
上	圧縮しようとしている係数にもっとも近い4の倍数
右	圧縮しようとしている係数にもっとも近い4の倍数+1
左	圧縮しようとしている係数にもっとも近い4の倍数+2
下	圧縮しようとしている係数にもっとも近い4の倍数+3

40

【0087】

なお、このようにして再生成した圧縮コードは、元の圧縮コードと同じビット数を持つとは限らない。すなわち、編集箇所のビット符号を、書き換えたい色に対応するビット符号に置き換えると、ビットの符号量は変化する。ところが、書き換えたい色に対応するビット符号は、エントロピー符号化を用いて圧縮されたファイルであれば、例外を除いて書

50

き換え前のコードよりもサイズが小さい。

【0088】

なぜならば、書き換えたい文字の色は、変えたい色に対する固定の値であり、ブロック内の画素値がすべてその値になるからである。書き換え対象のデータの値は、基本的には原稿から読み取った値である場合が多い。読み取る原稿では同じ色の画素であっても、読み取り特性により多少値にばらつきがある。エントロピー符号化では、エントロピーが大きい方が符号量は多いため、置き換え後のビット符号のほうが小さい(図16)。

【0089】

置き換えた結果、減少したビット数は、その領域に続く無効領域のビット符号を増やすことで吸収する。無効領域はどんな色に対応するデータでも画質に影響しないため、問題はない。その結果、置き換えたい領域とそれに続く無効領域のデータのみに影響が留まる。それ以降のデータのビットシフトは不要となる(図17)。なお、仮にビット符号が編集後に増えた場合は、無効領域のビット符号を減らすことで対応可能な場合は対応することもできる。

10

【0090】

このようにハフマン符号表では、同じビット数のコードが存在しうる。予め各コードに意味を割り当てて、編集内容に応じたコードを出力することで、ファイルサイズは変えることなく情報を埋め込むことができる。同じビット数のコードがない場合は、それよりもビット数が多く、複数のコードが存在するビット数のコードを出力する。本来出力すべきビット数よりもビット数が増えてしまうが、ここも編集領域とみなして、その後の無効領域で吸収すれば良い。このようにしてビットシフトの移動ロスを防ぐことが可能である。

20

【0091】

一般的には、前景画像に各ブロックにおいては、圧縮コードは減るため、前景画像のファイルサイズは増えることはない。黒文字画像に関しては、増えることになるが、2値のデータであるため、わずかな増加で済む。

【0092】

上述した実施形態では、修正画素は単一の色で置き換えることしかできなかった。それでも問題ないケースが多いが、本発明による他の実施形態では、同一ブロック内に、異なる色で置き換えたい修正画素を持つための方法について説明する。

30

【0093】

上述した実施形態との相違点は、図9におけるS96と図7におけるS76である。すなわち、図9のS96において、修正画素は、その修正画素が含まれるブロックの、一番近いブロックに置き換えるべき色を持つ可能性が高い。そのため、予めいくつかの考えられるパターンを用意しておき、ブロック内のすべての修正画素候補がパターンに一致するか否かを確認する。

【0094】

図18は、本発明の他の実施形態における置き換える色を持つブロックへのベクトルを示す図である。パターン1として、修正画素が最も近い上下左右のブロックに置き換える。パターン2として、修正画素が最も近い左右のブロックに置き換える。パターン3として、修正画素が最も近い上下のブロックに置き換える。なお、パターン1において、最も近いブロックが複数ある場合は、予めパターンとしてどちらにするかを決めておくものとする。そして、パターンに一致した場合は、どのパターンに一致したかを出力する。

40

【0095】

図7におけるS76について、本発明の実施形態との相違点のみを示す。圧縮コードには、パターンに関する情報も出力する必要があるため、表1に加えて、パターンの情報を追加する。

【表 2】

置き換える色のブロック/置き換える色が同じブロック	一番高い高周波成分の係数	
上	圧縮しようとしている係数にもっとも近い7の倍数	
右	圧縮しようとしている係数にもっとも近い7の倍数+1	
左	圧縮しようとしている係数にもっとも近い7の倍数+2	10
下	圧縮しようとしている係数にもっとも近い7の倍数+3	
パターン1(上下左右)	圧縮しようとしている係数にもっとも近い7の倍数+4	
パターン2 (左右)	圧縮しようとしている係数にもっとも近い7の倍数+5	
パターン3 (上下)	圧縮しようとしている係数にもっとも近い7の倍数+6	20

【0096】

以下に、本発明を用いて生成されたファイルのデコードについて記載する。
実施の形態について、

【表 3】

文字画像	黒文字画像	表示画素
文字	黒文字	画素が含まれる前景画像のブロックの一番高い高周波成分のAC成分の値にもとづいてデコードする。(表4) 参照
文字	非黒文字	前景画像の画素を表示
非文字	黒文字	黒文字画素として表示
非文字	非黒文字	背景画像の画素を表示

【表 4】

一番高い高周波成分の AC 成分の係数	置き換える色のブロック/置き換える色が同じブロック	
圧縮しようとしている係数にもっとも近い4の倍数	上のブロックの代表値を計算し、その値をこの画素の値として表示する	40
圧縮しようとしている係数にもっとも近い4の倍数+1	右のブロックの代表値を計算し、その値をこの画素の値として表示する	
圧縮しようとしている係数にもっとも近い4の倍数+2	左のブロックの代表値を計算し、その値をこの画素の値として表示する	
圧縮しようとしている係数にもっとも近い4の倍数+3	下のブロックの代表値を計算し、その値をこの画素の値として表示する	

なお、ブロックの代表値は、図9におけるS93で用いた計算方法で算出する。

【0097】

他の実施形態について、実施形態に対するデコードとの差分は、表4であるため、対応する表を以下に示す。

【表5】

一番高い高周波成分の AC 成分の係数	置き換える色のブロック/置き換える色が同じブロック	
圧縮しようとしている係数にもっとも近い4の倍数	上のブロックの代表値を計算し、その値をこの画素の値として表示する	10
圧縮しようとしている係数にもっとも近い4の倍数+1	右のブロックの代表値を計算し、その値をこの画素の値として表示する	
圧縮しようとしている係数にもっとも近い4の倍数+2	左のブロックの代表値を計算し、その値をこの画素の値として表示する	
圧縮しようとしている係数にもっとも近い4の倍数+3	下のブロックの代表値を計算し、その値をこの画素の値として表示する	
圧縮しようとしている係数にもっとも近い4の倍数+4	この画素が上下左右のブロックのどこに一番近いかを判断し、一番近いブロックの代表値にする。	20
圧縮しようとしている係数にもっとも近い4の倍数+5	この画素が左右のブロックのどこに一番近いかを判断し、一番近いブロックの代表値にする。	
圧縮しようとしている係数にもっとも近い4の倍数+6	この画素が上下のブロックのどこに一番近いかを判断し、一番近いブロックの代表値にする。	

【0098】

以上説明してきたように、従来技術と異なり、文字の色を画像（前景画像）で表現する技術において、プレビュー画面において、文字の色変わりを指定された領域に対応する前景画像上のブロックを、そのブロック内に含まれる色を減少させて圧縮を行うことにより、そのブロックの圧縮コード量を低減させて、当該減少させた色を、当該減少させた色と近い当該ブロックに隣接する色のベクトル情報として、当該低減させた圧縮コードの部分に格納すると同時に、減少させた色に対応する画素を識別する情報を格納するようにしたので、文字の色を画像（前景画像）で表現する技術において、プレビュー画面上で、文字の色変わりを改善することができる。

【0099】

以上、本発明の好適な実施の形態により本発明を説明した。ここでは特定の具体例を示して本発明を説明したが、特許請求の範囲に定義された本発明の広範囲な趣旨及び範囲から逸脱することなく、これら具体例に様々な修正及び変更が可能である。

【符号の説明】

【0100】

- 1 画像処理装置
- 10 プロッタ
- 11 コピー画像
- 12 ネットワーク
- 13 外部PC端末
- 14 HDD
- 15 NIC
- 16 操作部
- 2 スキャナ

10

20

30

40

50

- 3 原稿
- 4 スキャナ補正部
- 5 圧縮処理部
- 6 汎用バス
- 7 コントローラ
- 8 伸張処理部
- 9 プリンタ補正部

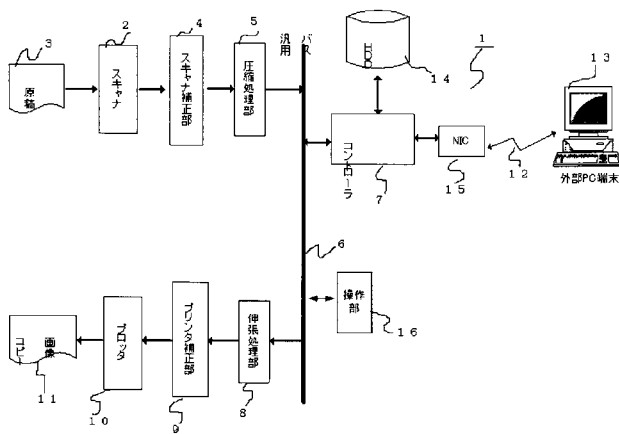
【先行技術文献】

【特許文献】

【0101】

【特許文献1】特開2008-236204号公報

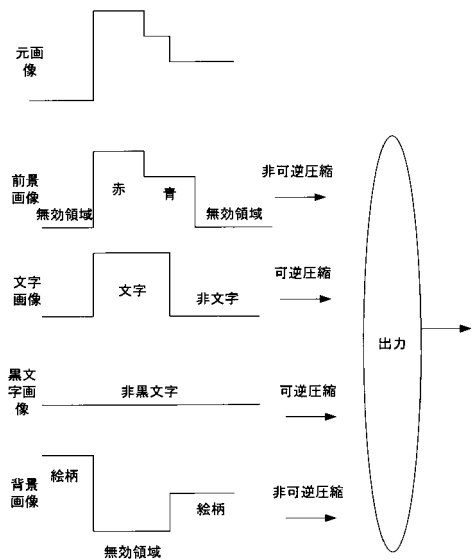
【図1】



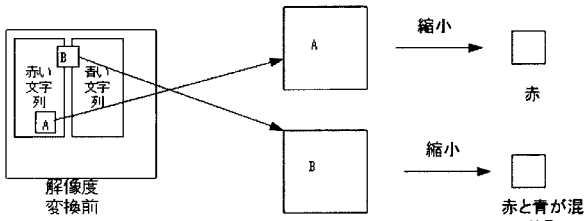
【図2】

1次元で記載

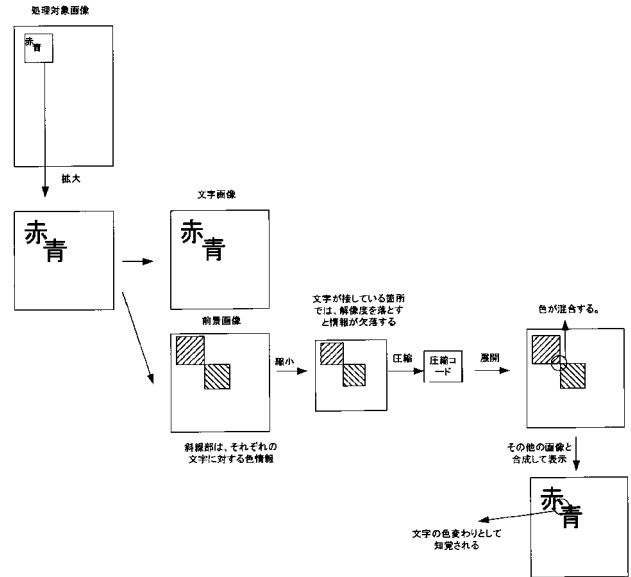
- ・文字画像が“文字”の箇所が有効画素。それ以外は無効画素
- ・文字/非文字を表現する。基本的には、黒文字画素は非文字になる。
- ・黒文字画素/非黒文字画素を表現する。
- ・黒文字画素がないときは、すべて非黒文字画素
- ・文字画素で“非文字”の箇所が有効画素。それ以外は無効画素



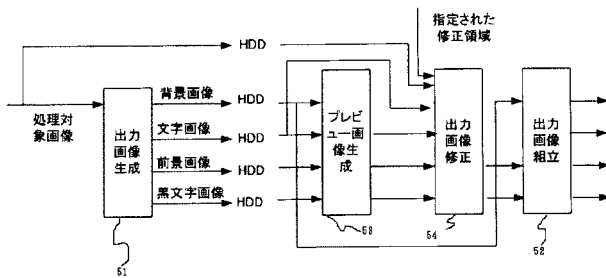
【 図 3 】



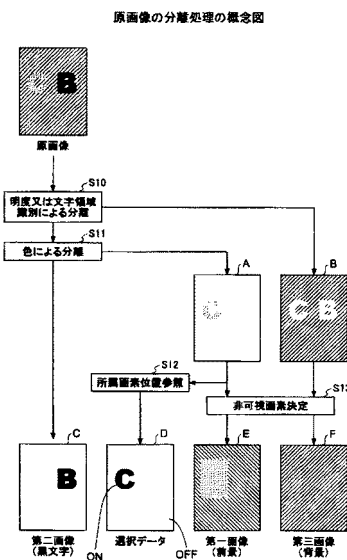
【 図 4 】



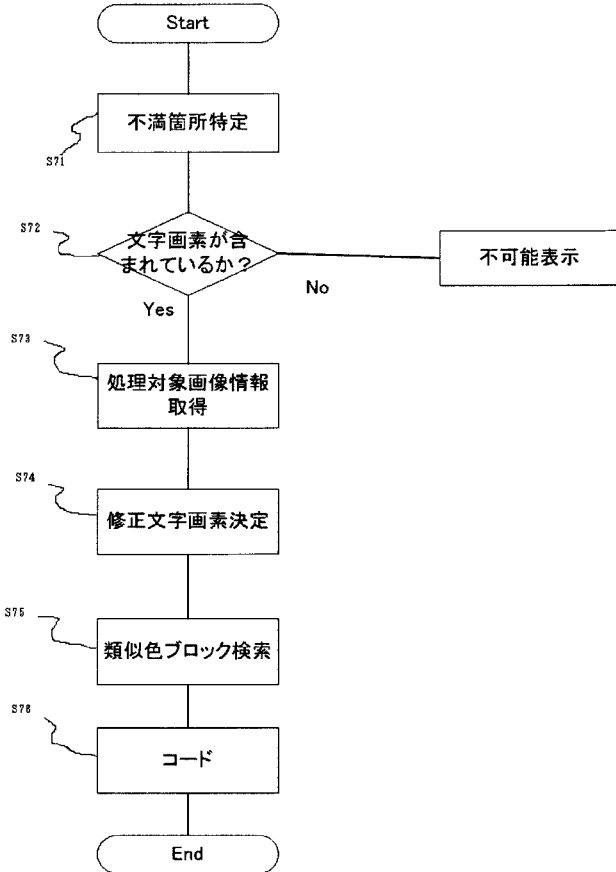
【 図 5 】



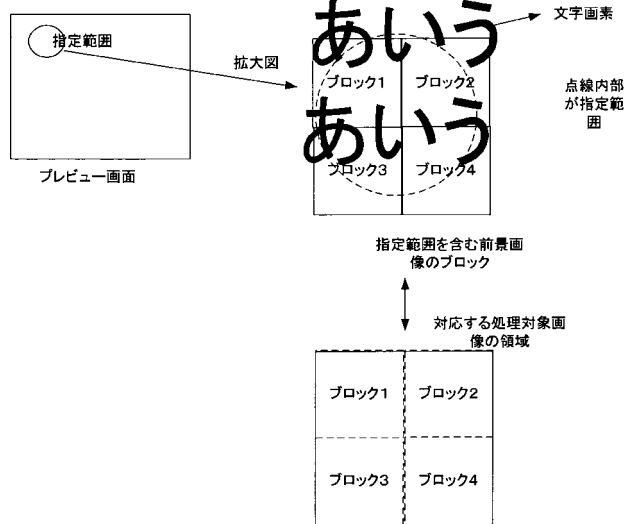
【 図 6 】



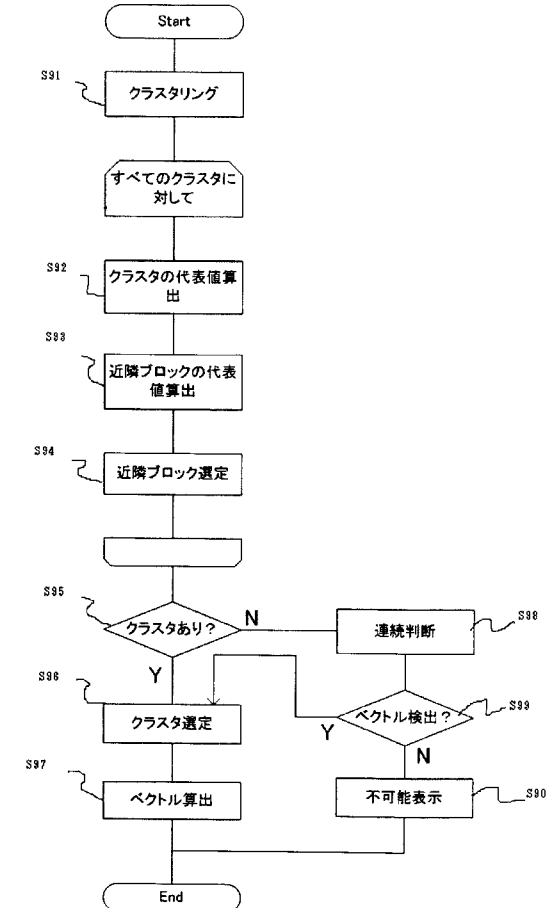
【図7】



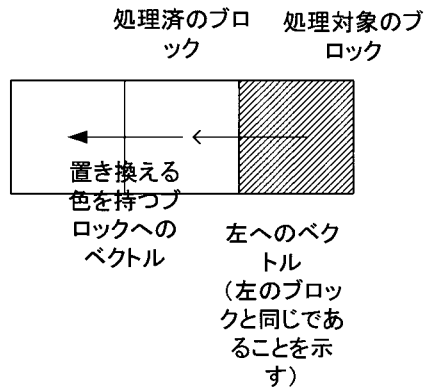
【図8】



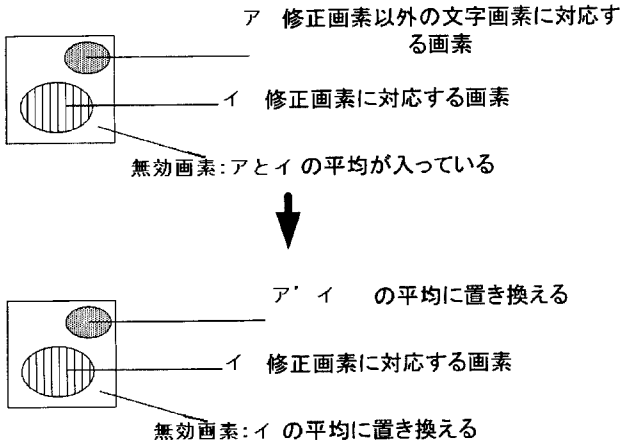
【図9】



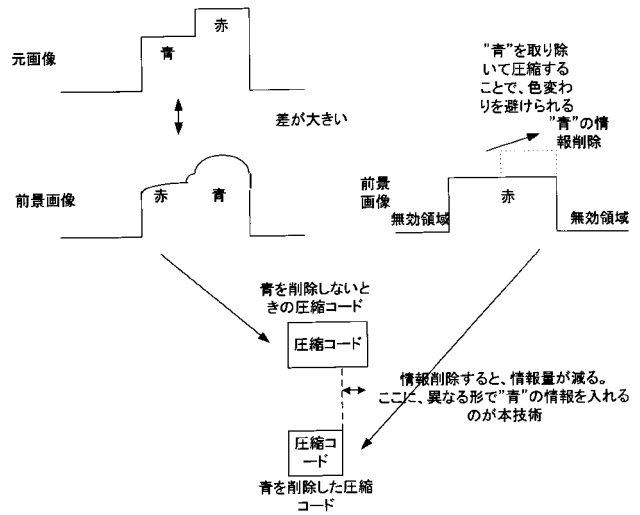
【図10】



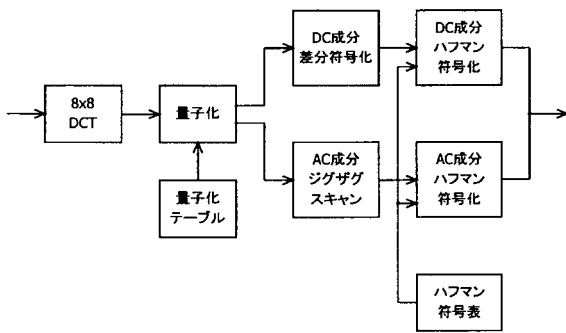
【図 1 1】



【図 1 2】



【図 1 3】

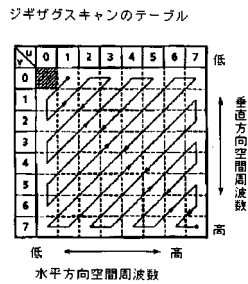


【図 1 5】

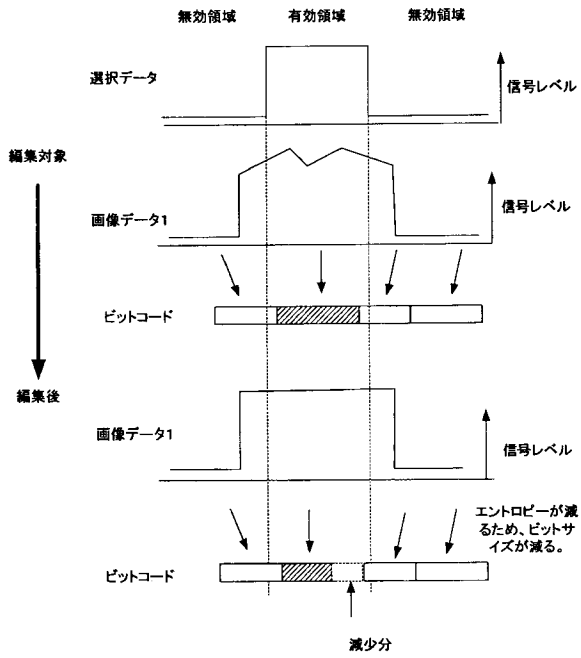
run	run = 1	run = 2	run = 3	run = 4	run = 5
0	00	01	100	1011	1111
1	1100	11011	11110001	111110110	1111110110
2	11100	1111001	111110111	11111101001	11111110001001
3	111010	111110111	11111110101	1111111000111	111111110010000
4	1110100	1111110000	11111110001010	111111110001011	1111111100110001000
5	1111010	1111110111	11111110011110	111111110011111	11111111101000000
6	11110101	11111110111	11111110100110	111111110100111	11111111101010000
7	11111010	11111110111	11111111010110	111111110110111	11111111101100000
8	111110100	1111111100000	111111110110110	111111110110111	11111111101110000
9	111111001	1111111011110	111111110111111	111111111000000	1111111111000001
10	111111001	111111101100000	111111111010001	111111111010100	1111111111001010
11	1111111001	111111101100000	111111111010001	111111111010100	1111111111010011
12	1111111000	1111111011011001	111111111011010	111111111011011	11111111110110100
13	11111111000	1111111011100010	1111111110110011	1111111110110010	11111111110110101
14	111111110100	11111111101100	111111111011010	111111111011101	1111111111101111
15	1111111001	111111111110101	111111111110110	111111111110111	111111111111000

run	run = 6	run = 7	run = 8	run = 9	run = 10
6	11111000	11111000	1111110110	11111110000010	111111110000011
7	111111110000100	111111110000101	1111111110000110	1111111110000111	1111111110001000
8	111111110000100	111111110000101	1111111110000110	1111111110000111	1111111110001000
9	1111111110001001	1111111110001010	1111111110000111	1111111110000111	1111111110001001
10	1111111110001001	1111111110000110	1111111110000111	1111111110000111	1111111110001001
11	1111111110001001	1111111110000110	1111111110000111	1111111110000111	1111111110001001
12	1111111110001001	1111111110000110	1111111110000111	1111111110000111	1111111110001001
13	1111111110001001	1111111110000110	1111111110000111	1111111110000111	1111111110001001
14	1111111110001001	1111111110000110	1111111110000111	1111111110000111	1111111110001001
15	1111111110001001	1111111110000110	1111111110000111	1111111110000111	1111111110001001

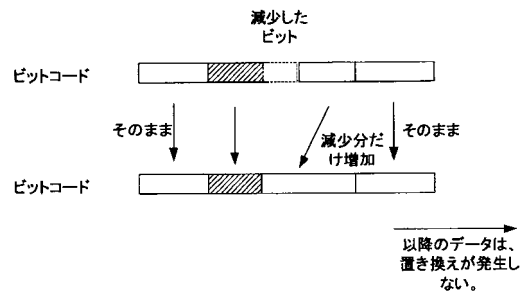
【図 1 4】



【 図 1 6 】



【 図 1 7 】



【 図 1 8 】

