

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4236659号
(P4236659)

(45) 発行日 平成21年3月11日(2009.3.11)

(24) 登録日 平成20年12月26日(2008.12.26)

(51) Int.Cl.		F I			
HO4N	1/413	(2006.01)	HO4N	1/413	D
HO4N	7/26	(2006.01)	HO4N	7/13	Z
HO4N	1/41	(2006.01)	HO4N	1/41	Z

請求項の数 14 (全 39 頁)

(21) 出願番号	特願2005-319812 (P2005-319812)	(73) 特許権者	000005049
(22) 出願日	平成17年11月2日(2005.11.2)		シャープ株式会社
(65) 公開番号	特開2007-129456 (P2007-129456A)		大阪府大阪市阿倍野区長池町2番2号
(43) 公開日	平成19年5月24日(2007.5.24)	(74) 代理人	100064746
審査請求日	平成18年10月30日(2006.10.30)		弁理士 深見 久郎
		(74) 代理人	100085132
			弁理士 森田 俊雄
		(74) 代理人	100083703
			弁理士 仲村 義平
		(74) 代理人	100096781
			弁理士 堀井 豊
		(74) 代理人	100098316
			弁理士 野田 久登
		(74) 代理人	100109162
			弁理士 酒井 将行

最終頁に続く

(54) 【発明の名称】 画像符号化装置、画像符号化方法、画像符号化プログラムおよび記録媒体

(57) 【特許請求の範囲】

【請求項1】

入力画像を所定のブロック単位で像域判定するブロック像域判定手段と、
前記ブロック像域判定手段の像域判定結果に基づいて、前記入力画像のデータを使用して、画像再生時に有効であるか無効であるかを前記所定のブロック単位で規定された2以上の層データを生成する層データ生成手段と、
前記層データ生成手段が生成した前記2以上の層データの各々に適する符号化条件で、前記2以上の層データの各々に適する前記所定のブロック単位のサイズ以下のブロック単位で、前記2以上の層データの各々を非可逆符号化処理する符号化手段とを備え、
前記2以上の層データのうち、少なくとも2つの層データは画像データであり、
前記層データ生成手段は、前記画像データである層データに基づく画像のうち、前記画像再生時に無効であると規定された1以上のブロック内の各々のブロック画像を構成する複数の画素値を、透過色であるか否かを示す透過色値に設定する、画像符号化装置。

【請求項2】

前記符号化手段は、各々が、前記2以上の層データのそれぞれに適する符号化条件で、前記2以上の層データのそれぞれに適するブロック単位で、前記2以上の層データをそれぞれ符号化処理する2以上のデータ符号化手段を含む、請求項1に記載の画像符号化装置。

【請求項3】

前記2以上の層データのうち、少なくとも1つの層データは画像データであり、

前記層データ生成手段は、前記画像データである層データに基づく画像のうち、前記画像再生時に無効であると規定された1以上のブロック内の各々のブロック画像を構成する複数の画素値を設定し、

前記層データ生成手段により設定された前記複数の画素値は、前記入力画像のデータを使用して設定される場合より、前記符号化手段が前記画像データの層データを前記符号化処理した符号化データ容量が小さくなる値である、請求項1または請求項2に記載の画像符号化装置。

【請求項4】

前記2以上の層データのうち、少なくとも1つの層データは画像データであり、

前記入力画像の解像度を変換した変換画像を生成する解像度変換手段をさらに備え、

前記層データ生成手段は、前記画像データである層データに基づく画像のうち、前記画像再生時に有効であると規定された1以上のブロックにそれぞれ対応する、前記変換画像内の1以上のブロック画像のデータを設定する、請求項1～請求項3のいずれかに記載の画像符号化装置。

【請求項5】

前記入力画像を画素単位で像域判定する画素像域判定手段をさらに備え、

前記ブロック像域判定手段は、前記画素像域判定手段の画素単位の像域判定結果に基づいて、ブロック単位の像域判定を行なう、請求項1～請求項3のいずれかに記載の画像符号化装置。

【請求項6】

前記ブロック像域判定手段は、判定対象のブロックが、文字または線を含むか否かを判定し、

前記層データ生成手段が生成した前記2以上の層データの少なくとも1つの層データは画像データであり、

前記画像データである層データに基づく画像は、文字または線が表示される画像である、請求項1～請求項3のいずれかに記載の画像符号化装置。

【請求項7】

前記ブロック像域判定手段は、判定対象のブロックが、文字、線およびエッジ部の少なくとも1つを含むか否かを判定し、

前記層データ生成手段が生成した前記2以上の層データの少なくとも1つの層データは画像データであり、

前記画像データである層データに基づく画像は、文字、線およびエッジ部の少なくとも1つが表示される画像である、請求項1～請求項3のいずれかに記載の画像符号化装置。

【請求項8】

前記層データ生成手段は、前記画像データである少なくとも2つの層データのうち、前記透過色値が設定されていない、少なくとも1つの層データに基づく画像のうち、前記画像再生時に無効であると規定された1以上のブロック内の各々のブロック画像を構成する複数の画素値を、前記透過色値と同一の値に設定する、請求項1～請求項7のいずれかに記載の画像符号化装置。

【請求項9】

複数のデータを関連付けた関連付けデータを生成するフォーマット手段をさらに備え、

前記フォーマット手段は、前記符号化処理された前記2以上の層データを関連付け、

前記2以上の層データのうちの少なくとも1つの層データは、前記透過色値に設定されたデータを含む、請求項1～請求項8のいずれかに記載の画像符号化装置。

【請求項10】

前記非可逆符号化処理は、JPEG方式に従う符号化処理である、請求項1～請求項9のいずれかに記載の画像符号化装置。

【請求項11】

前記所定のブロックは、横および縦の画素数が等しく、前記画素数は8に自然数を乗じた値である、請求項1～請求項10のいずれかに記載の画像符号化装置。

10

20

30

40

50

【請求項 1 2】

入力画像を所定のブロック単位で像域判定する工程と、
前記像域判定する工程の像域判定結果に基づいて、前記入力画像のデータを使用して、画像再生時に有効であるか無効であるかを前記所定のブロック単位で規定された 2 以上の層データを生成する工程と、

前記 2 以上の層データを生成する工程が生成した前記 2 以上の層データの各々に適する符号化条件で、前記 2 以上の層データの各々に適する前記所定のブロック単位のサイズ以下のブロック単位で、前記 2 以上の層データの各々を非可逆符号化処理する工程とを含み

前記 2 以上の層データのうち、少なくとも 2 つの層データは画像データであり、
前記層データを生成する工程は、前記画像データである層データに基づく画像のうち、前記画像再生時に無効であると規定された 1 以上のブロック内の各々のブロック画像を構成する複数の画素値を、透過色であるか否かを示す透過色値に設定する工程を含む、画像符号化方法。

10

【請求項 1 3】

コンピュータに画像処理を実行させるための画像符号化プログラムであって、
入力画像を所定のブロック単位で像域判定するステップと、
前記像域判定するステップの像域判定結果に基づいて、前記入力画像のデータを使用して、画像再生時に有効であるか無効であるかを前記所定のブロック単位で規定された 2 以上の層データを生成するステップと、

20

前記 2 以上の層データを生成するステップが生成した前記 2 以上の層データの各々に適する符号化条件で、前記 2 以上の層データの各々に適する前記所定のブロック単位のサイズ以下のブロック単位で、前記 2 以上の層データの各々を非可逆符号化処理するステップとをコンピュータに実行させ、

前記 2 以上の層データのうち、少なくとも 2 つの層データは画像データであり、
前記層データを生成するステップは、前記画像データである層データに基づく画像のうち、前記画像再生時に無効であると規定された 1 以上のブロック内の各々のブロック画像を構成する複数の画素値を、透過色であるか否かを示す透過色値に設定するステップを含む、画像符号化プログラム。

30

【請求項 1 4】

請求項 1 3 に記載の画像符号化プログラムを記録した、記録媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、画像符号化装置、画像符号化方法、画像符号化プログラムおよび記録媒体に関し、特に、複数の層毎に画像を処理する画像符号化装置、画像符号化方法、画像符号化プログラムおよび記録媒体に関する。

【背景技術】

【0002】

近年、紙の書類等、あらゆる物が電子化されるようになってきている。そのため、電子化されたデータを記録する装置や媒体に、より多くのデータを記憶させるには、データ量の効率的な削減が求められる。

40

【0003】

たとえば、文字および写真のような複数の異なる要素が配置された画像のデータを、記録装置または記録媒体に記録する際には、画像データの容量を効率よく削減する技術が重要となる。すなわち、画質劣化が小さく、かつ、データ容量の削減量が大きくなるように、符号化する技術が重要となる。以下においては、たとえば、文字、写真等の要素が表示される画像内の領域を像域ともいう。ここで、記録装置は、たとえば、パーソナルコンピュータ、複写機または複合機などに備えられるハードディスクである。また、記録媒体は、たとえば、CD-R である。

50

【 0 0 0 4 】

特開平 7 - 2 1 2 6 0 1 号公報 (特許文献 1) には、符号化処理で用いるマトリクス (ブロック) 単位で像域分離を行ない、像域分離後の結果にマトリクスを使用した符号化を適用する技術 (以下、従来技術 A ともいう) が開示されている。

【 0 0 0 5 】

また、特開平 7 - 2 1 2 6 0 1 号公報 (特許文献 1) には、写真部をマトリクス単位で効率的に分離するための技術 (以下、従来技術 A ともいう) が開示されている。従来技術 A では、写真部をマトリクス単位で符号化 (たとえば、J P E G (Joint Photographic Coding Experts Group) し、文字部 (写真以外の部分) を 2 値化して可逆符号化 (たとえば、ランレングス符号化) する。これにより、複数の異なる像域から成る画像データを効率よく符号化することが可能である。

10

【 0 0 0 6 】

特開 2 0 0 4 - 1 8 7 0 0 0 号公報 (特許文献 2) には、I T U - T 勧告 T . 4 4 に規定される M R C (Mixed Raster Content) に基づく技術 (以下、従来技術 B ともいう) が開示されている。

【 0 0 0 7 】

M R C では、文字、線および写真など複数の異なる像域から成る画像データを、前景プレーン、背景プレーン、マスクプレーンの合計 3 層のプレーンに分離する。ここで、マスクプレーンとは、前景プレーンおよび背景プレーンのいずれを選択するかを示すプレーンである。そして、M R C では、3 層のプレーンを、個別に符号化する方式を規定している。

20

【 0 0 0 8 】

一般に、M R C を使用する画像符号化装置では、文字および線を表す前景プレーンと、写真などを表す背景プレーンとを選択する情報であるマスクプレーンは、画素単位で構成されている。

【 0 0 0 9 】

また、M R C を使用する画像符号化装置では、文字および線の高解像度での形状情報の保持はマスクプレーンに委ねて高解像度での可逆符号化を行なう。そして、文字および線の色情報を保持する前景プレーンと、解像度低下による劣化が目立たない写真などの背景プレーンとは、マスクプレーンに比べて低解像度で非可逆符号化される場合が多い。

30

【 0 0 1 0 】

また、特開 2 0 0 4 - 1 8 7 0 0 0 号公報 (特許文献 2) には、前景プレーン中における背景像域のドント・ケア (don't care) 画素や、背景プレーン中における前景像域のドント・ケア画素の画素値の算出方法を工夫することにより、前景プレーンおよび背景プレーンの解像度を低解像度化する際に発生する、色濁りなどを低減する技術が開示されている。ここで、ドント・ケア画素とは、データを合成して画像を再生する際に参照されない画素のことである。

【 0 0 1 1 】

一方、画像データを電子的に記録する場合、データフォーマット (データ形式) の選択が重要となる。たとえば、一つの装置内で画像データを符号化して、保存し、復号処理も実行する場合は独自のデータフォーマットでもよい。しかしながら、符号化して、記録されたデータを装置間で伝送する場合などは、一般に広く用いられるデータフォーマットに従うことが好ましい。

40

【 0 0 1 2 】

たとえば、J P E G 方式に従うデータフォーマットは、パーソナルコンピュータやデジタルスチルカメラなどで広く利用されている。また、A d o b e (登録商標) 社によって開発された P D F (Portable Document Format) も、広く利用されている画像データフォーマットの一つである。

【特許文献 1】特開平 7 - 2 1 2 6 0 1 号公報

【特許文献 2】特開 2 0 0 4 - 1 8 7 0 0 0 号公報

50

【発明の開示】

【発明が解決しようとする課題】

【0013】

従来技術 A では、文字に対しては、2 値化して可逆符号化する。したがって、従来技術 A では、写真像域中に存在する色文字や色線については、可逆符号化の対象とすることが困難となり、写真像域向けの符号化条件が適用される。

【0014】

すなわち、従来技術 A では、色などの条件によっては、符号化することで文字や線の形状が見た目に劣化するという問題がある。さらに、従来技術 A では、広く利用されているデータフォーマットを使用しておらず、汎用性に欠けるという問題点がある。

【0015】

また、従来技術 B では、文字および線の形状情報の保持をマスクプレーンに委ねている。そのため、マスクプレーンを符号化する場合、高解像度で、かつ、可逆である必要があるため、マスクプレーンの保持に多くのデータ容量が必要となる。

【0016】

また、従来技術 B では、ドント・ケア画素に起因する色濁りを低減しているが、完全な回避は困難という問題点がある。

【0017】

本発明は、上述の問題点を解決するためになされたものであって、その目的は、符号化処理において画質の劣化を最小限に抑えつつ、データ容量を大幅に削減した汎用性の高いデータを作成可能な画像符号化装置を提供することである。

【0018】

本発明の他の目的は、符号化処理において画質の劣化を最小限に抑えつつ、データ容量を大幅に削減した汎用性の高いデータを作成可能な画像符号化方法を提供することである。

【0019】

本発明のさらに他の目的は、符号化処理において画質の劣化を最小限に抑えつつ、データ容量を大幅に削減した汎用性の高いデータを作成可能な画像符号化プログラムを提供することである。

【0020】

本発明のさらに他の目的は、符号化処理において画質の劣化を最小限に抑えつつ、データ容量を大幅に削減した汎用性の高いデータを作成可能な画像符号化プログラムを記録した記録媒体を提供することである。

【課題を解決するための手段】

【0021】

上述の課題を解決するために、この発明のある局面に従う画像符号化装置は、入力画像を所定のブロック単位で像域判定するブロック像域判定手段と、ブロック像域判定手段の像域判定結果に基づいて、入力画像のデータを使用して、画像再生時に有効であるか無効であるかを所定のブロック単位で規定された 2 以上の層データを生成する層データ生成手段と、層データ生成手段が生成した 2 以上の層データの各々に適する符号化条件で、2 以上の層データの各々に適するブロック単位で、2 以上の層データの各々を符号化処理する符号化手段とを備える。

【0022】

この発明に従えば、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【0023】

好ましくは、符号化手段は、各々が、2 以上の層データのそれぞれに適する符号化条件で、2 以上の層データのそれぞれに適するブロック単位で、2 以上の層データをそれぞれ符号化処理する 2 以上のデータ符号化手段を含む。

【0024】

この発明に従えば、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【0025】

好ましくは、2以上の層データのうち、少なくとも1つの層データは画像データであり、層データ生成手段は、画像データである層データに基づく画像のうち、画像再生時に無効であると規定された1以上のブロック内の各々のブロック画像を構成する複数の画素値を設定し、層データ生成手段により設定された複数の画素値は、入力画像のデータを使用して設定される場合より、符号化手段が画像データの層データを符号化処理した符号化データ容量が小さくなる値である。

【0026】

この発明に従えば、データ容量の削減量がより大きくなるように、かつ、設定した画素値に起因する色濁りの影響を再生画像に生じさせることなく、画像を符号化することができる。

【0027】

好ましくは、2以上の層データのうち、少なくとも1つの層データは画像データであり、入力画像の解像度を変換した変換画像を生成する解像度変換手段をさらに備え、層データ生成手段は、画像データである層データに基づく画像のうち、画像再生時に有効であると規定された1以上のブロックにそれぞれ対応する、変換画像内の1以上のブロック画像のデータを設定する。

【0028】

この発明に従えば、データ容量の削減量がより大きくなるように画像を符号化することができる。

【0029】

好ましくは、入力画像を画素単位で像域判定する画素像域判定手段をさらに備え、ブロック像域判定手段は、画素像域判定手段の画素単位の像域判定結果に基づいて、ブロック単位の像域判定を行なう。

【0030】

この発明に従えば、ブロック単位の像域判定をより正確に行なうことができる。

好ましくは、ブロック像域判定手段は、判定対象のブロックが、文字または線を含むか否かを判定し、層データ生成手段が生成した2以上の層データの少なくとも1つの層データは画像データであり、画像データである層データに基づく画像は、文字または線が表示される画像である。

【0031】

この発明に従えば、高周波成分が比較的大きい文字または線を含むブロックに、適切なパラメータを設定して符号化を行い、文字または線を比較的高精細に保持して符号化することができる。

【0032】

好ましくは、ブロック像域判定手段は、判定対象のブロックが、文字、線およびエッジ部の少なくとも1つを含むか否かを判定し、層データ生成手段が生成した2以上の層データの少なくとも1つの層データは画像データであり、画像データである層データに基づく画像は、文字、線およびエッジ部の少なくとも1つが表示される画像である。

【0033】

この発明に従えば、像域の判定をブロック単位で行っているため、エッジに注目して層データを生成することができる。また、高周波成分が比較的大きい文字または線またはエッジを含むブロックに、適切なパラメータを設定して符号化を行い、文字または線またはエッジを比較的高精細に保持して符号化することができる。

【0034】

好ましくは、2以上の層データのうち、少なくとも1つの層データは画像データであり、2以上の層データのうち、画像データ以外の層データのうち少なくとも1つの層データは、各々が、画像データである層データの画像再生時に所定のブロック単位で有効である

10

20

30

40

50

が無効であることを示す複数のマスクデータから構成されるマスク層データである。

【0035】

この発明に従えば、画素単位でマスクデータを生成する場合に比べて、複数のマスクデータから構成されるマスク層データの容量を削減することができる。

【0036】

好ましくは、マスク層データを可逆符号化する可逆符号化手段をさらに備える。

この発明に従えば、マスク層データの容量を更に削減することができる。

【0037】

好ましくは、2以上の層データのうち、少なくとも2つの層データは画像データであり、層データ生成手段は、画像データである層データに基づく画像のうち、画像再生時に無効であると規定された1以上のブロック内の各々のブロック画像を構成する複数の画素値を、透過色であるか否かを示す透過色値に設定する。

10

【0038】

この発明に従えば、マスク層データを独立して保持することなく、複数の層データを重ね合わせて画像を再生することができる。

【0039】

好ましくは、層データ生成手段は、画像データである少なくとも2つの層データのうち、透過色値が設定されていない、少なくとも1つの層データに基づく画像のうち、画像再生時に無効であると規定された1以上のブロック内の各々のブロック画像を構成する複数の画素値を、透過色値と同一の値に設定する。

20

【0040】

この発明に従えば、透過色部分について下層のデータを用いる場合に、より正確に画像を再生することを可能とする。

【0041】

好ましくは、複数のデータを関連付けた関連付けデータを生成するフォーマット手段をさらに備え、フォーマット手段は、符号化処理された2以上の層データと、可逆符号化されたマスク層データとを関連付ける。

【0042】

この発明に従えば、汎用性の高いデータフォーマットを用いて表現することができる。

好ましくは、複数のデータを関連付けた関連付けデータを生成するフォーマット手段をさらに備え、フォーマット手段は、符号化処理された2以上の層データを関連付け、2以上の層データのうちの少なくとも1つの層データは、透過色値に設定されたデータを含む。

30

【0043】

この発明に従えば、汎用性の高いデータフォーマットを用いて表現することができる。

好ましくは、符号化処理は、非可逆符号化処理である。

【0044】

この発明に従えば、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【0045】

好ましくは、非可逆符号化処理は、JPEG方式に従う符号化処理である。

この発明に従えば、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

40

【0046】

好ましくは、所定のブロックは、横および縦の画素数が等しく、画素数は8に自然数を乗じた値である。

【0047】

この発明に従えば、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【0048】

50

この発明の他の局面に従う画像符号化方法は、入力画像を所定のブロック単位で像域判定する工程と、像域判定する工程の像域判定結果に基づいて、入力画像のデータを使用して、画像再生時に有効であるか無効であるかを所定のブロック単位で規定された2以上の層データを生成する工程と、2以上の層データを生成する工程が生成した2以上の層データの各々に適する符号化条件で、2以上の層データの各々に適するブロック単位で、2以上の層データの各々を符号化処理する工程とを含む。

【0049】

この発明に従えば、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【0050】

好ましくは、符号化処理は、非可逆符号化処理である。

この発明に従えば、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【0051】

この発明のさらに他の局面に従うと、コンピュータに画像処理を実行させるための画像符号化プログラムは、入力画像を所定のブロック単位で像域判定するステップと、像域判定するステップの像域判定結果に基づいて、入力画像のデータを使用して、画像再生時に有効であるか無効であるかを所定のブロック単位で規定された2以上の層データを生成するステップと、2以上の層データを生成するステップが生成した2以上の層データの各々に適する符号化条件で、2以上の層データの各々に適するブロック単位で、2以上の層データの各々を符号化処理するステップとをコンピュータに実行させる。

【0052】

この発明に従えば、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【0053】

好ましくは、符号化処理は、非可逆符号化処理である。

この発明に従えば、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【0054】

この発明のさらに他の局面に従うと、記録媒体は、画像符号化プログラムを記録した媒体である。

【0055】

この発明に従えば、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【発明の効果】

【0056】

本発明に係る画像符号化装置は、入力画像を所定のブロック単位で像域判定し、像域判定結果に基づいて、入力画像のデータを使用して、画像再生時に有効であるか無効であるかを所定のブロック単位で規定された2以上の層データを生成する。そして、生成した2以上の層データの各々に適する符号化条件で、2以上の層データの各々に適するブロック単位で、2以上の層データの各々を符号化処理する。

【0057】

したがって、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【0058】

本発明に係る画像符号化方法は、入力画像を所定のブロック単位で像域判定し、像域判定結果に基づいて、入力画像のデータを使用して、画像再生時に有効であるか無効であるかを所定のブロック単位で規定された2以上の層データを生成する。そして、生成した2以上の層データの各々に適する符号化条件で、2以上の層データの各々に適するブロック単位で、2以上の層データの各々を符号化処理する。

10

20

30

40

50

【 0 0 5 9 】

したがって、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【 0 0 6 0 】

本発明に係る画像符号化プログラムは、入力画像を所定のブロック単位で像域判定し、像域判定結果に基づいて、入力画像のデータを使用して、画像再生時に有効であるか無効であるかを所定のブロック単位で規定された2以上の層データを生成する。そして、生成した2以上の層データの各々に適する符号化条件で、2以上の層データの各々に適するブロック単位で、2以上の層データの各々を符号化処理する。

【 0 0 6 1 】

したがって、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【 0 0 6 2 】

本発明に係る記録媒体は、画像符号化プログラムを記録する。画像符号化プログラムは、入力画像を所定のブロック単位で像域判定し、像域判定結果に基づいて、入力画像のデータを使用して、画像再生時に有効であるか無効であるかを所定のブロック単位で規定された2以上の層データを生成する。そして、生成した2以上の層データの各々に適する符号化条件で、2以上の層データの各々に適するブロック単位で、2以上の層データの各々を符号化処理する。

【 0 0 6 3 】

したがって、画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【 発明を実施するための最良の形態 】

【 0 0 6 4 】

以下、図面を参照しつつ、本発明の実施の形態について説明する。以下の説明では、同一の部品には同一の符号を付してある。それらの名称および機能も同じである。したがって、それらについての詳細な説明は繰り返さない。

【 0 0 6 5 】

< 第1の実施の形態 >

図1は、第1の実施の形態における画像符号化装置1000の構成を示すブロック図である。図1を参照して、画像符号化装置1000は、画像処理部100と、データバス50と、制御部120と、一時記憶部130と、記憶部140と、記録媒体アクセス部150と、記録媒体70とを備える。

【 0 0 6 6 】

データバス50には、画像処理部100、制御部120、一時記憶部130、記憶部140および記録媒体アクセス部150が接続されている。

【 0 0 6 7 】

記憶部140には、画像処理部100および制御部120に後述する処理を行なわせるための画像符号化プログラム72、入力画像データ、その他各種プログラムおよびデータ等が記憶されている。記憶部140は、画像処理部100および制御部120によってデータアクセスされる。

【 0 0 6 8 】

記憶部140は、大容量のデータを記憶可能なハードディスクである。なお、記憶部140は、ハードディスクに限定されることなく、電源を供給されなくてもデータを不揮発的に保持可能な媒体（たとえば、フラッシュメモリ）であればよい。

【 0 0 6 9 】

画像処理部100は、詳細は後述するが、記憶部140に記憶された画像符号化プログラム72に従って、後述する画像処理を行なう。

【 0 0 7 0 】

制御部120は、記憶部140に記憶された画像符号化プログラム72に従って、画像

10

20

30

40

50

符号化装置 1 0 0 0 内の各部に対する各種処理や、演算処理等を行なう機能を有する。また、制御部 1 2 0 は、画像処理部 1 0 0 で行なわれている処理を監視する機能を有する。

【 0 0 7 1 】

制御部 1 2 0 は、マイクロプロセッサ (Microprocessor)、プログラミングすることができる L S I (Large Scale Integration) である F P G A (Field Programmable Gate Array)、特定の用途のために設計、製造される集積回路である A S I C (Application Specific Integrated Circuit)、その他の演算機能を有する回路のいずれであってもよい。

【 0 0 7 2 】

一時記憶部 1 3 0 は、データを一時的に記憶する機能を有する。一時記憶部 1 3 0 は、画像処理部 1 0 0 および制御部 1 2 0 によってデータアクセスされ、ワークメモリとして動作する。一時記憶部 1 3 0 は、R A M (Random Access Memory)、S R A M (Static Random Access Memory)、D R A M (Dynamic Random Access Memory)、S D R A M (Synchronous DRAM)、D D R - S D R A M (Double Data Rate SDRAM)、R D R A M (Rambus Dynamic Random Access Memory)、D i r e c t - R D R A M (Direct Rambus Dynamic Random Access Memory)、その他、データを揮発的に記憶保持可能な構成を有する回路のいずれであってもよい。

【 0 0 7 3 】

記録媒体 7 0 には、前述した画像符号化プログラム 7 2 が記録されている。

記録媒体アクセス部 1 5 0 は、画像符号化プログラム 7 2 が記録された記録媒体 7 0 から、画像符号化プログラム 7 2 を読出す機能を有する。記録媒体 7 0 に記憶されている画像符号化プログラム 7 2 は、制御部 1 2 0 のインストール処理により、記録媒体アクセス部 1 5 0 から読み出され、記憶部 1 4 0 に記憶される。

【 0 0 7 4 】

このインストール処理用プログラムは、予め、記憶部 1 4 0 に格納されており、インストール処理は、制御部 1 2 0 が、インストール処理用プログラムに基づいて行なう。

【 0 0 7 5 】

なお、記憶部 1 4 0 には、画像符号化プログラム 7 2 が記憶されていなくてもよい。この場合、制御部 1 2 0 は、記録媒体アクセス部 1 5 0 を介して、記録媒体 7 0 に記憶された画像符号化プログラム 7 2 を読み出して、画像符号化プログラム 7 2 に基づいた所定の処理を行なう。

【 0 0 7 6 】

記録媒体 7 0 は、画像符号化装置 1 0 0 0 から着脱可能な媒体である。すなわち、記録媒体 7 0 に記録されている画像符号化プログラム 7 2 は、媒体等に記録されてプログラム製品として流通される。また、記録媒体 7 0 もプログラム製品として流通される。

【 0 0 7 7 】

記録媒体 7 0 は、D V D - R O M (Digital Versatile Disk Read Only Memory)、D V D - R (Digital Versatile Disk Recordable)、D V D - R A M (Digital Versatile Disk Random Access Memory)、D V D + R W (Digital Versatile Disk Re-Writable)、D V D - R W、C D - R O M (Compact Disk Read Only Memory)、M O (Magneto Optical Disk)、M D (Mini Disc) (登録商標)、フロッピー (登録商標) ディスク、着脱可能なハードディスク、C F (Compact Flash) カード、S M (Smart Media (登録商標))、M M C (Multi Media Card)、S D (Secure Digital) メモリーカード、メモリースティック (登録商標)、x D ピクチャーカードおよび U S B メモリ、カセットテープ、磁気テープ、I C カード (メモリーカードを含む)、光カード、マスク R O M、E P R O M、E E P R O M、フラッシュ R O M、その他不揮発性メモリのいずれであってもよい。

【 0 0 7 8 】

画像処理部 1 0 0 は、ブロック像域判定部 1 0 1 と、第 1 画像縮小部 1 0 2 と、第 2 画像縮小部 1 0 3 と、マスク生成部 1 0 4 と、前景画像生成部 1 0 5 と、背景画像生成部 1 0 6 とを含む。

10

20

30

40

50

【 0 0 7 9 】

ブロック像域判定部 1 0 1 には、入力画像のデータ（以下、入力画像データともいう）が入力される。入力画像は、たとえば、図示しないイメージスキャナ等を用いて読み込まれた画像である。入力画像データは、R G B などの色空間に従うデジタル画像データである。入力画像データは、記憶部 1 4 0 に記憶される。

【 0 0 8 0 】

ブロック像域判定部 1 0 1 は、詳細は後述するが、入力画像の像域種別を所定のブロック単位で判定する。ここで、所定のブロック単位とは、入力画像、前景画像および背景画像の各々の解像度と、前景画像符号化部 1 0 8、背景画像符号化部 1 0 9 の各々の符号化処理単位に基づいて、システム設計者が予め設定すればよい。また、所定のブロック単位は、処理条件に応じて、予め設定したブロックサイズの中から、自動的に選択されるようにしてもよい。ブロックサイズの具体例については、後述する。

10

【 0 0 8 1 】

ブロック像域判定部 1 0 1 は、マスク生成部 1 0 4 と接続される。ブロック像域判定部 1 0 1 は、像域種別の判定結果をマスク生成部 1 0 4 へ出力する。なお、ブロック像域判定部 1 0 1 は、たとえば、専用の L S I によって構成される。

【 0 0 8 2 】

第 1 画像縮小部 1 0 2 には、入力画像データが入力される。第 1 画像縮小部 1 0 2 は、入力画像の解像度を下げることにより、入力画像を縮小する。第 1 画像縮小部 1 0 2 は、前景画像生成部 1 0 5 と接続される。第 1 画像縮小部 1 0 2 は、縮小した入力画像（以下、第 1 縮小画像ともいう）のデータを、前景画像生成部 1 0 5 へ出力する。第 1 縮小画像は、文字、線等の像域に対し後述する処理が行なわれる画像である。なお、第 1 画像縮小部 1 0 2 は、たとえば、専用の L S I によって構成される。

20

【 0 0 8 3 】

第 2 画像縮小部 1 0 3 には、入力画像データが入力される。第 2 画像縮小部 1 0 3 は、入力画像の解像度を下げることにより、入力画像を縮小する。第 2 画像縮小部 1 0 3 は、背景画像生成部 1 0 6 と接続される。第 2 画像縮小部 1 0 3 は、縮小した入力画像（以下、第 2 縮小画像ともいう）のデータを、背景画像生成部 1 0 6 へ出力する。第 2 縮小画像は、写真等の像域に対し後述する処理が行なわれる画像である。なお、第 2 画像縮小部 1 0 3 は、たとえば、専用の L S I によって構成される。

30

【 0 0 8 4 】

ここで、第 1 縮小画像および第 2 縮小画像の解像度（サイズ）は、目標とする画質や符号化画像データ容量に応じて決定すればよい。

【 0 0 8 5 】

たとえば、第 1 縮小画像の縦および横方向の解像度（サイズ）は、入力画像の縦および横方向の解像度（サイズ）をそれぞれ、2 の M 乗の値および 2 の M ' 乗の値で除した値を有する解像度（サイズ）とすればよい。なお、M、M ' は、0 以上の整数とし、 $M = M'$ の関係が成立することが好ましい。また、第 2 縮小画像の縦および横方向の解像度（サイズ）は、入力画像の縦および横方向の解像度（サイズ）をそれぞれ、2 の N 乗の値および 2 の N ' 乗の値で除した解像度（サイズ）とすればよい。なお、N、N ' は、0 以上の整数とし、 $N = N'$ の関係が成立することが好ましい。

40

【 0 0 8 6 】

また、写真等の像域に対し後述する処理が行なわれる第 2 縮小画像の解像度は、文字、線等の像域に対し後述する処理が行なわれる第 1 縮小画像の解像度よりも、解像度が低いこと（ $M < N$ かつ $M' < N'$ ）が好ましい。なぜなら、一般に、文字や線の解像度より、写真や背景色の解像度が低くても、写真や背景色の画質の劣化は目立たないからである。解像度の組み合わせに関する詳細な説明および具体例については、後述する。

【 0 0 8 7 】

第 1 画像縮小部 1 0 2 および第 2 画像縮小部 1 0 3 における、画像の縮小処理には、ニアレストネイバー（Nearest Neighbor）法、バイリニア（Bilinear）法、バイキュービッ

50

ク (Bicubic) 法など、画像の解像度変換に一般的に用いられる既存の補間手法が使用される。

【 0 0 8 8 】

マスク生成部 1 0 4 は、ブロック像域判定部 1 0 1 と接続される。マスク生成部 1 0 4 は、ブロック像域判定部 1 0 1 で判定された像域種別の判定結果に基づいて、マスク (マスクデータ) を生成する。以下においては、複数のマスクデータから構成されるデータをマスク層データともいう。

【 0 0 8 9 】

マスクデータは、画像再生時に、処理対象のブロック画像が、前景画像として有効または無効であるかを示す情報である。ここで、ブロック画像とは、ブロック単位の画像のことをいう。また、「有効」とは、画像再生時に下層 (本実施の形態では背景層) を上書きすることを示す。また、「無効」とは、画像再生時に下層 (本実施の形態では背景層) のデータをそのまま用いることを示す。

10

【 0 0 9 0 】

本実施の形態では、画像またはデータが、3層構造で処理される。1つ目の層は、マスク層である。マスク層では、マスク層データ、後述するマスク符号化データが処理される。2つ目の層は、前景層である。前景層では、前景画像が処理される。3つ目の層は、背景層である。背景層では、背景画像が処理される。

【 0 0 9 1 】

画像処理部 1 0 0 は、さらに、マスク符号化部 1 0 7 と、前景画像符号化部 1 0 8 と、背景画像符号化部 1 0 9 と、フォーマット部 1 1 0 とを含む。

20

【 0 0 9 2 】

マスク生成部 1 0 4 は、マスク符号化部 1 0 7 と接続される。マスク生成部 1 0 4 は、生成したマスクをマスク符号化部 1 0 7 へ出力する。

【 0 0 9 3 】

また、マスク生成部 1 0 4 は、前景画像生成部 1 0 5 と接続される。マスク生成部 1 0 4 は、生成したマスクデータを前景画像生成部 1 0 5 へ出力する。また、マスク生成部 1 0 4 は、背景画像生成部 1 0 6 と接続される。マスク生成部 1 0 4 は、生成したマスクデータを背景画像生成部 1 0 6 へ出力する。なお、マスク生成部 1 0 4 は、たとえば、専用の L S I によって構成される。

30

【 0 0 9 4 】

前景画像生成部 1 0 5 は、マスク生成部 1 0 4 と、第 1 画像縮小部 1 0 2 とに接続される。前景画像生成部 1 0 5 は、マスク生成部 1 0 4 で生成された複数のマスクデータからなるマスク層データに基づいて、第 1 画像縮小部 1 0 2 で縮小された第 1 縮小画像のデータを使用して、前景画像 (前景層データ) を生成する。また、前景画像生成部 1 0 5 は、前景画像符号化部 1 0 8 と接続される。前景画像生成部 1 0 5 は、生成した前景画像のデータを前景画像符号化部 1 0 8 へ出力する。なお、前景画像生成部 1 0 5 は、たとえば、専用の L S I によって構成される。

【 0 0 9 5 】

背景画像生成部 1 0 6 は、マスク生成部 1 0 4 と、第 2 画像縮小部 1 0 3 とに接続される。背景画像生成部 1 0 6 は、マスク生成部 1 0 4 で生成された複数のマスクデータからなるマスク層データに基づいて、第 2 画像縮小部 1 0 3 で縮小された第 2 縮小画像のデータを使用して、背景画像 (背景層データ) を生成する。また、背景画像生成部 1 0 6 は、背景画像符号化部 1 0 9 と接続される。背景画像生成部 1 0 6 は、生成した背景画像のデータを背景画像符号化部 1 0 9 へ出力する。なお、背景画像生成部 1 0 6 は、たとえば、専用の L S I によって構成される。

40

【 0 0 9 6 】

マスク符号化部 1 0 7 は、マスク生成部 1 0 4 と接続される。マスク符号化部 1 0 7 は、マスク生成部 1 0 4 で生成された複数のマスクデータからなるマスク層データを可逆符号化する。マスクデータは、前述したように、対応するブロック画像が、前景画像として

50

有効であるか否かを示すデータである。

【 0 0 9 7 】

可逆符号化では、各画素（または各ブロック）単位における選択のための情報を2値で表現する場合（2値画像データを使用する場合）、2値画像の可逆符号化に適した符号化方式が使用される。当該符号化方式は、MR（Modified Read）、MMR（Modified Modified Read）、JBIG（Joint Bi-level Image Coding Expert Group）等のいずれの符号化方式であってもよい。

【 0 0 9 8 】

マスク符号化部107は、フォーマット部110と接続される。マスク符号化部107は、符号化したマスクデータをフォーマット部110へ出力する。なお、マスク符号化部107は、たとえば、専用のLSIによって構成される。

10

【 0 0 9 9 】

前景画像符号化部108は、前景画像生成部105と接続される。前景画像符号化部108は、前景画像生成部105で生成された前景画像を非可逆符号化する。非可逆符号化を行なう符号化方式は、多値画像の符号化に好適なJPEG方式が使用される。なお、非可逆符号化を行なう符号化方式は、JPEG方式に限定されることなく、他の符号化方式であってもよい。

【 0 1 0 0 】

また、前景画像符号化部108は、フォーマット部110と接続される。前景画像符号化部108は、符号化した前景画像のデータをフォーマット部110へ出力する。なお、前景画像符号化部108は、たとえば、専用のLSIによって構成される。

20

【 0 1 0 1 】

背景画像符号化部109は、背景画像生成部106と接続される。背景画像符号化部109は、背景画像生成部106で生成された背景画像を非可逆符号化する。非可逆符号化を行なう符号化方式は、多値画像の符号化に好適なJPEG方式が使用される。なお、非可逆符号化を行なう符号化方式は、JPEG方式に限定されることなく、他の符号化方式であってもよい。

【 0 1 0 2 】

また、背景画像符号化部109は、フォーマット部110と接続される。背景画像符号化部109は、符号化した背景画像のデータをフォーマット部110へ出力する。なお、背景画像符号化部109は、たとえば、専用のLSIによって構成される。

30

【 0 1 0 3 】

フォーマット部110は、マスク符号化部107、前景画像符号化部108および背景画像符号化部109と接続される。フォーマット部110は、マスク符号化部107、前景画像符号化部108および背景画像符号化部109で、それぞれ生成された、マスク符号化データ、前景符号化データ、背景符号化データを統合的に取り扱うことができるように関係付けたデータ（以下、関係付けデータともいう）を作成する。

【 0 1 0 4 】

フォーマット部110は、マスク符号化データ、後述する前景画像符号化データ、後述する背景画像符号化データおよび関係付けデータを一つの符号化画像データとして、たとえば、一つの汎用的な電子データ（以下、汎用統合データともいう）を生成する。なお、汎用統合データは、図1の圧縮画像データに相当する。フォーマット部110が生成する汎用統合データのフォーマットは、たとえば、Adobe（登録商標）社のPDFに従うデータフォーマットである。

40

【 0 1 0 5 】

フォーマット部110は、汎用統合データを、データバス50を介して、一時記憶部130または記憶部140に記憶させる。また、フォーマット部110は、汎用統合データを、LAN（Local Area Network）などの通信経路を介して外部機器へ出力してもよい。

【 0 1 0 6 】

なお、前述の説明では、第1画像縮小部102および第2画像縮小部103は、それぞ

50

れ異なる構成要素として説明した。しかしながら、第1画像縮小部102および第2画像縮小部103は、一つの画像縮小部として構成されてもよい。この場合、画像縮小部は、解像度変換時のパラメータを変更して、適宜、第1画像縮小部102または第2画像縮小部103の機能による処理を行なえばよい。

【0107】

また、前述の説明では、前景画像符号化部108および背景画像符号化部109は、それぞれ異なる構成要素として説明した。しかしながら、前景画像符号化部108および背景画像符号化部109のどちらもJPEG方式を用いる場合、前景画像符号化部108および背景画像符号化部109は、一つのJPEG符号化LSIとして構成されてもよい。

【0108】

また、前述のブロック像域判定部101、第1画像縮小部102、第2画像縮小部103、マスク生成部104、前景画像生成部105、背景画像生成部106、マスク符号化部107、前景画像符号化部108、背景画像符号化部109およびフォーマット部110の各々が行なう処理は、専用のLSIではなく、パーソナルコンピュータなどの一般的なコンピュータのマイクロプロセッサまたは制御部120により実行されてもよい。この場合、たとえば、後述する画像符号化処理をコンピュータまたは制御部120に実行させるためのプログラムとして記述してもよい。当該プログラムは、画像符号化プログラム72であり、記録媒体70に記録されて流通する。

【0109】

コンピュータにより、後述する画像符号化処理が実行される場合、画像符号化プログラム72は、記録媒体70に記録され、コンピュータが備える記録媒体アクセス部により、コンピュータが備える記憶部に読み込まれ、CPU(Central Processing Unit)で実行される。

【0110】

制御部120により、後述する画像符号化処理が実行される場合、画像符号化プログラム72は、記録媒体70に記録され、制御部120により実行される。

【0111】

さらに、画像符号化プログラム72は、インターネットなどのネットワークを介して他の装置からダウンロードされてもよい。

【0112】

また、画像処理部100に含まれる、前述した複数の専用のLSIとコンピュータのマイクロプロセッサとを組み合わせてもよい。また、前述した複数の専用のLSIの一部または全てを一つの専用のLSIとして構成してもよい。

【0113】

次に、第1の実施の形態における画像符号化装置1000の具体的な処理内容について説明する。

【0114】

図2は、第1の実施の形態において、画像符号化装置1000で実行される画像符号化処理のフローチャートを示す。図2を参照して、ステップS101では、画像データ読み出し処理が行なわれる。画像データ読み出し処理では、画像処理部100が、記憶部140に記憶されている入力画像データを、所定のブロック単位で読み出す。

【0115】

図3は、第1の実施の形態において、画像符号化装置1000で処理される画像を示す図である。なお、図3に示される各画像の解像度を忠実に図にて表現することは困難である。そのため、図3では、後述する図6のデータテーブルT100の組合せ番号“1”に対応する解像度と、ブロックサイズの関係を想定して、相対的な関係を例示する。

【0116】

図3を参照して、入力画像100Gは、入力画像データに基づく画像である。入力画像100Gは、文字と、写真とが配置された画像である。なお、写真が表示されている領域内には、文字も配置されている。

10

20

30

40

50

【 0 1 1 7 】

ステップ S 1 0 1 の画像データ読み出し処理において、読み出される画像は、たとえば、所定のブロックに対応する領域 1 0 0 R 内の画像である。領域 1 0 0 R のサイズは、たとえば、横 3 2 画素、縦 3 2 画素のサイズである。ステップ S 1 0 1 の処理が繰り返される毎に、領域 1 0 0 R の位置は、前回の位置と重ならない位置に移動する。なお、初めてステップ S 1 0 1 の処理が行なわれるときは、領域 1 0 0 R の位置は、入力画像 1 0 0 G の左上である。

【 0 1 1 8 】

具体的には、ステップ S 1 0 1 の処理が繰り返される毎に、領域 1 0 0 R の位置は、右に移動する。そして、領域 1 0 0 R の位置が入力画像 1 0 0 G の右端になると、次のステップ S 1 0 1 の処理では、領域 1 0 0 R の位置は、領域 1 0 0 R の縦のサイズだけ下の位置であって、入力画像 1 0 0 G の左端に移動する。上記処理が、繰り返されることにより、最終的に、領域 1 0 0 R の位置は、入力画像 1 0 0 G の右下の位置（領域 1 0 0 R F の位置）まで移動する。

10

【 0 1 1 9 】

再び、図 2 を参照して、読み出された所定のブロック単位の入力画像データは、ブロック像域判定部 1 0 1、第 1 画像縮小部 1 0 2 および第 2 画像縮小部 1 0 3 へ入力される。以下においては、ブロック単位の画像データを、ブロック画像データともいう。また、前述したように、ブロック単位の画像のことをブロック画像という。また、画像データが入力画像データである場合のブロック画像データを入力ブロック画像データともいう。画像が入力画像である場合のブロック画像を入力ブロック画像ともいう。その後、ステップ S 1 1 0 に進む。

20

【 0 1 2 0 】

ステップ S 1 1 0 では、像域判定処理が行なわれる。像域判定処理では、ブロック像域判定部 1 0 1 が、ステップ S 1 0 1 により入力された入力ブロック画像データに対し、後述する処理を行なう。ブロック像域判定部 1 0 1 は、領域 1 0 0 R 内の各画素単位ではなく、所定のブロック画像（たとえば、領域 1 0 0 R 内の画像）単位で後述する処理を行なう。この点が、本発明の一つの特徴である。像域判定処理は、処理対象の像域（ブロック画像）が、文字または線が存在する像域（ブロック画像）あるか否かを判定する処理である。

30

【 0 1 2 1 】

図 4 は、像域判定処理のフローチャートである。図 4 を参照して、ステップ S 1 1 2 では、ブロック像域判定部 1 0 1 が、入力された入力ブロック画像（領域 1 0 0 R 内の画像）内に、文字または線が存在するか否かを判定する。当該判定は、予め生成した画素単位マスクプレーンに基づいて行なわれる。

【 0 1 2 2 】

画素単位マスクプレーンは、入力画像の各画素が、文字または線画部分に該当する画素であるか否かを示す複数のデータ（以下、文字判定データともいう）から構成される。したがって、画素単位マスクプレーンは、入力画像（たとえば、入力画像 1 0 0 G）の全画素数と同じ数の文字判定データから構成される。また、複数の文字判定データは、入力画像の複数の画素（全画素）にそれぞれ対応する。

40

【 0 1 2 3 】

本実施の形態では、文字判定データが、入力画像の文字または線画部分に該当する画素に対応する場合、文字判定データを“ 1 ”とする。また、文字判定データが、入力画像の文字または線画部分に該当しない画素に対応する場合、文字判定データを“ 0 ”とする。

【 0 1 2 4 】

画素単位マスクプレーンの作成は、たとえば、入力画像の各画素の輝度データを用いて、特開平 2 - 1 2 3 4 7 9 号公報または特開平 2 - 1 5 5 0 8 7 号公報に開示されている技術により行なわれる。具体的には、制御部 1 2 0 が、記憶部 1 4 0 に記憶されている入力画像データから、予め用意した尾根画素検出パターンを適用して尾根画素を検出する。

50

そして、制御部 120 が、予め定めた領域単位で計数することより、計数値に基づいて文字または線画部分に該当する画素であるか否かを判定し、複数の文字判定データの各々を“1”または“0”に設定する。制御部 120 は、以上の処理により生成した画素単位マスクプレーンを、記憶部 140 に記憶させる。なお、画像符号化装置 1000 の外部で、画素単位マスクプレーンを予め生成し、記憶部 140 に記憶させてもよい。

【0125】

次に、入力されたブロック画像（領域 100R 内の画像）内に、文字または線が存在するか否かを判定する具体的な処理を説明する。

【0126】

ブロック像域判定部 101 は、記憶部 140 に記憶された画素単位マスクプレーンを参照し、入力されたブロック画像（領域 100R 内の画像）内の複数の画素にそれぞれ対応する複数の文字判定データのうち、“1”に設定された文字判定データが1つ以上あるか否かを判定する。“1”に設定された文字判定データが1つ以上あれば、ブロック画像内には、文字または線が存在することになる。なお、判定の対象となる“1”に設定された文字判定データの数は、1以上に限定されることなく、N（2以上の自然数）以上であってもよい。以下においては、文字または線が存在するブロック画像を、線分ブロック画像ともいう。また、文字または線が存在しないブロック画像を、非線分ブロック画像ともいう。

10

【0127】

なお、ブロック像域判定部 101 は、複数の文字判定データが、全て“1”に設定されている場合、判定対象のブロック画像を、線分ブロック画像と判定してもよい。ブロック像域判定部 101 は、複数の文字判定データのうち、“0”に設定された文字判定データが1つ以上ある場合、判定対象のブロック画像を、非線分ブロック画像と判定してもよい。

20

【0128】

ステップ S112 において、YES ならば、ステップ S114A に進む。一方、ステップ S112 において、NO ならば、ステップ S114B に進む。

【0129】

なお、ステップ S112 では、予め生成した画素単位マスクプレーンを使用しなくても、判定対象のブロック画像が、線分ブロック画像および非線分ブロック画像のいずれであるか判定することが可能である。

30

【0130】

次に、画素単位マスクプレーンを使用せず、判定対象のブロック画像内の複数の画素値を用いて、判定対象のブロック画像が、線分ブロック画像および非線分ブロック画像のいずれであるか判定する方法を説明する。

【0131】

この方法は、特開平 5 - 014701 号公報に開示されている技術を用いる。具体的には、まず、ブロック像域判定部 101 が、判定対象のブロック画像内の複数の画素値から最大濃度レベルと最小濃度レベルの差を求める。次に、ブロック像域判定部 101 は、濃度レベルの差のデータに基づいて閾値（たとえば、中間レベル）を定める。次に、ブロック像域判定部 101 は、ブロック内の画素を所定の方向に走査した際に濃度レベルが極大および極小となる点を抽出し、極大および極小の画素レベルの差からパルスの高さを求める。次に、ブロック像域判定部 101 は、求めたパルスの高さが、上記の閾値よりも大きい場合について計数する。次に、ブロック像域判定部 101 は、計数値が所定の値よりも小さいか否かで、判定対象のブロック画像が、線分ブロック画像または非線分ブロック画像であると判定することができる。

40

【0132】

また、他の方法としては、特開平 5 - 114045 号公報に開示されている技術を用いる。具体的には、まず、ブロック像域判定部 101 が、「判定対象のブロック画像内で所定の閾値以上の濃度（たとえば、閾値以下の輝度値）を持つ画素の総数」を「判定対象の

50

ブロック画像内の画素の総数」で除算した値（以下、黒画素密度ともいう）を算出する。次に、ブロック像域判定部 101 は、黒画素密度の大小に基づいて、判定対象のブロック画像が、線分ブロック画像または非線分ブロック画像であるか判定することができる。すなわち、線分ブロック画像では黒画素密度が比較的小さい傾向にあることを利用する。

【0133】

また、他の方法としては、特開平 5 - 114045 号公報に開示されている技術を用いる。具体的には、まず、ブロック像域判定部 101 が、「判定対象のブロック画像内で所定の閾値以上の画素値（輝度値）を持つ画素の総数」を「判定対象のブロック画像内の画素の総数」で除算した値（以下、黒画素密度ともいう）を算出する。次に、ブロック像域判定部 101 は、黒画素密度の大小に基づいて、判定対象のブロック画像が、線分ブロッ

10

【0134】

ク画像または非線分ブロック画像であるか判定することができる。すなわち、線分ブロック画像では黒画素密度が比較的小さい傾向にあることを利用する。

【0135】

また、他の方法としては、文字、線に加え、エッジ部分も判定に利用する。具体的には、まず、ブロック像域判定部 101 が、判定対象のブロック画像について横および縦の各方向に Sobel オペレータなどの微分（または差分）オペレータを適用して、対応する画素単位で適用結果の絶対値を加算して傾きの大きさを算出する。

20

【0136】

次に、ブロック像域判定部 101 は、算出した傾きの大きさが閾値より大きい画素を文字・線またはエッジ部分であると判定する。判定対象のブロック画像が、文字・線またはエッジ部分を含む場合、ブロック像域判定部 101 は、判定対象のブロック画像が、文字・線またはエッジ部分を含む線分ブロック画像であると判定することができる。

【0137】

なお、判定対象のブロック画像が、文字・線またはエッジ部分を含むか否かの判定は、判定対象のブロック画像の画素値および当該ブロック画像の周辺の画素値を用いて行なわれてもよい。

【0138】

ステップ S114A では、ブロック像域判定部 101 が、線分ブロックデータを“1”に設定する。線分ブロックデータは、処理対象のブロック画像が、線分ブロック画像であるか否かを示すデータである。線分ブロックデータが“1”に設定されている場合は、処理対象のブロック画像が、線分ブロック画像であることを示す。一方、線分ブロックデータが“0”に設定されている場合は、処理対象のブロック画像が、非線分ブロック画像であることを示す。

30

【0139】

ブロック像域判定部 101 は、“1”に設定した線分ブロックデータを、マスク生成部 104 へ送信する。また、ブロック像域判定部 101 は、処理対象のブロック画像を特定する情報と対応づけて、“1”に設定した線分ブロックデータを、一時記憶部 130 に記憶させる。その後、この像域判定処理は終了し、図 2 の画像符号化処理に戻り、ステップ S110 の次のステップ S120 に進む。

40

【0140】

ステップ S114B では、ブロック像域判定部 101 が、線分ブロックデータを“0”に設定する。ブロック像域判定部 101 は、“0”に設定した線分ブロックデータを、マスク生成部 104 へ送信する。また、ブロック像域判定部 101 は、処理対象のブロック画像を特定する情報と対応づけて、“0”に設定した線分ブロックデータを、一時記憶部 130 に記憶させる。その後、この像域判定処理は終了し、図 2 の画像符号化処理に戻り、ステップ S110 の次のステップ S120 に進む。

【0140】

再び、図 3 を参照して、画像 110G は、像域判定処理により設定された線分ブロックデータの状態を視覚的に分かりやすく示した画像である。入力画像 100G に対し、ブロック画像単位で前述の像域判定処理を行なった場合、処理対象のブロック画像が、線分ブ

50

ロック画像と判定されたブロック画像内の全画素は、白（たとえば、画素値“1”）で示される。また、処理対象のブロック画像が、非線分ブロック画像と判定されたブロック画像内の全画素は、黒（たとえば、画素値“0”）で示される。

【0141】

再び、図2を参照して、ステップS120では、マスク生成処理が行なわれる。マスク生成処理では、像域判定処理により設定された線分ブロックデータに基づいて、マスクデータが生成される。マスクデータは、処理対象のブロック画像が、前景画像として有効であるか否かを示すデータである。マスクデータは、“0”または“1”のいずれかを示す1ビットのデータである。

【0142】

マスクデータが“1”に設定された場合、処理対象のブロック画像は、前景画像として有効となる。一方、マスクデータが“0”に設定された場合、処理対象のブロック画像は、前景画像として無効となる。マスクデータは、ステップS110の像域判定処理で処理される1つのブロック画像に対し、1つの値を示す。なお、マスクデータは、ステップS110の像域判定処理で処理される1つのブロック画像内の全ての画素に対し、1つの値を示してもよい。

【0143】

図5は、マスク生成処理のフローチャートである。図5を参照して、ステップS122では、処理対象のブロック画像が、線分ブロック画像であるか否かが判定される。具体的には、マスク生成部104が、受信した線分ブロックデータが“1”に設定されているか否かを判定する。ステップS122において、YESならば、ステップS124Aに進む。一方、ステップS122において、NOならば、ステップS124Bに進む。

【0144】

ステップS124Aでは、マスク生成部104が、マスクデータを“1”に設定する。すなわち、処理対象のブロック画像は、前景画像として有効となる。マスク生成部104は、“1”に設定したマスクデータを、マスク符号化部107、前景画像生成部105および背景画像生成部106へ送信する。また、マスク生成部104は、処理対象のブロック画像を特定する情報と対応づけて、“1”に設定したマスクデータを、一時記憶部130に記憶させる。その後、このマスク生成処理は終了し、図2の画像符号化処理に戻り、ステップS120の次のステップS132に進む。

【0145】

ステップS124Bでは、マスク生成部104が、マスクデータを“0”に設定する。すなわち、処理対象のブロック画像は、前景画像として無効となる。マスク生成部104は、“0”に設定したマスクデータを、マスク符号化部107、前景画像生成部105および背景画像生成部106へ送信する。また、マスク生成部104は、処理対象のブロック画像を特定する情報と対応づけて、“0”に設定したマスクデータを、一時記憶部130に記憶させる。その後、このマスク生成処理は終了し、図2の画像符号化処理に戻り、ステップS120の次のステップS132に進む。

【0146】

再び、図3を参照して、画像120Gは、マスク生成処理により設定されたマスクデータの状態を視覚的に分かりやすく示した画像である。マスク生成処理により、前景画像として有効と判定されたブロック画像は、白（たとえば、画素値“1”）の1画素で示される。また、マスク生成処理により、前景画像として無効と判定されたブロック画像は、黒（たとえば、画素値“0”）の1画素で示される。ここで、画像120Gの解像度は、画像110Gの解像度600dpi（dots per inch）を、ブロック像域判定処理単位の画素数32画素で除した、18.75dpiとしている。なぜなら、一つのブロック画像に対し、一つの値が設定されているからである。

【0147】

再び、図2を参照して、ステップS132では、画像縮小処理Aが行なわれる。画像縮小処理Aでは、第1画像縮小部102が、ステップS101により入力された入力プロッ

10

20

30

40

50

ク画像データに基づいて、入力ブロック画像の解像度を下げる。すなわち、入力ブロック画像を縮小する。入力ブロック画像の縮小は、以下に説明するデータテーブルT100に基づいて行なわれる。

【0148】

図6は、データテーブルT100を示す図である。データテーブルT100は、画像の解像度とブロックサイズとの関係を示すテーブルである。図6を参照して、組合せ番号とは、画像の解像度と対応するブロックサイズの組み合わせを特定するための番号である。なお、画像の解像度と対応するブロックサイズの組み合わせは、データテーブルT100に示されるものに限定されることはない。

【0149】

たとえば、組合せ番号“1”では、入力画像、前景画像および背景画像の解像度を、それぞれ、600dpi、300dpiおよび150dpiとする。この場合、像域判定処理は、横32画素、縦32画素のサイズのブロック画像を1つの単位として行なわれる。また、前景画像は、横16画素、縦16画素のサイズのブロック画像毎に処理される。また、背景画像は、横8画素、縦8画素のサイズのブロック画像毎に処理される。組合せ番号“2”～“4”も、前述した組合せ番号“1”と同様であるので詳細な説明は繰り返さない。

【0150】

ここで、ブロックサイズの画素数を決定する際の基準に関して説明する。画像の解像度とブロックサイズとの関係を示すデータテーブルT100において、ブロックサイズの横画素数および縦画素数は、8の倍数である。8の倍数が用いられる理由は、前景画像符号化部108および背景画像符号化部109が、JPEG方式により画像を符号化するためである。

【0151】

JPEG方式は、公知の通り、8画素×8画素のサイズのブロック画像毎に処理を行なう。JPEG方式において、ブロック画像毎の処理では、まずDCT（離散コサイン変換）が行なわれ、1個のDC（直流）成分値と、63個のAC（交流）成分値が算出される。そして、算出された64個の値を、それぞれの成分に対応する量子化幅を規定した量子化テーブルを用いて、除算することにより量子化する。

【0152】

さらに、DC成分値については、一つ前に処理されたブロック画像のDC成分値との差分値を利用してハフマン符号化が行なわれる。AC成分値については、ブロック画像内の63個の値をジグザグスキャンしてハフマン符号化が行なわれる。なお、ハフマン符号化は可逆な符号化であり、一般に、JPEG方式における非可逆性は、量子化誤差およびDCT計算時の誤差に起因するものである。

【0153】

したがって、JPEG方式を用いて画像を符号化した際に生じる画質の劣化は、基本的に処理単位であるブロック画像（8画素×8画素）内で生じるものである。すなわち、ブロック画像内の画質の劣化は、ブロック画像内で閉じて生じるものである。そのため、JPEGの処理対象となるブロック画像は、当該ブロック画像外の画素の値や、周囲のブロック画像により、画質劣化が生じることはない。

【0154】

そこで、本発明では、画像の解像度とブロック画像のサイズとの関係を示すデータテーブルT100に示すように、像域判定、前景画像、背景画像の各処理におけるブロック画像のサイズの横画素数および縦画素数を8の倍数で規定している。これにより、前景画像、背景画像を生成する際、画像再生時に不要なブロックに、ブロック画像単位で任意の画素値を設定することが可能となる。

【0155】

すなわち、不要なブロック画像内のドント・ケア画素の画素値に起因する色にじみを防ぐことができる。その結果、不要なブロック画像に設定する画素値に関わらず、再生画像

10

20

30

40

50

を一定の品質に保つことができるという効果を奏する。

【0156】

なお、像域判定、前景画像、背景画像の各処理におけるブロック画像のサイズについて、最小解像度の画像（本実施の形態では背景画像）の処理単位となるブロック画像の最小のサイズは、8画素×8画素とすることが好ましい。以下においては、画像の処理単位となるブロック画像の最小のサイズを最小ブロック画像サイズともいう。

【0157】

また、最小解像度以上の画像（本実施の形態では入力画像と前景画像）の処理単位となるブロック画像の横画素数および縦画素数の各々は、以下の式（1）により算出される値とすることが好ましい。

画素数 = 8 × (処理対象の画像の解像度) / (最小解像度) . . . (1)

上記のように、最小ブロック画像サイズを8画素×8画素に設定することにより、像域分離精度を最大限に高めることができる。ここで、像域分離精度とは、文字または線が存在する像域（ブロック画像）と、文字または線が存在しない像域（ブロック画像）とを分離する精度である。また、最小ブロック画像サイズを処理単位としない画像には、式（1）で算出される値を用いてブロック画像のサイズを設定する。これにより、入力画像、前景画像および背景画像の間で、一つのブロック画像に対応する画像情報が、相対的に画像中の同じブロック画像部分を指し示すことを可能とする。

【0158】

具体的には、データテーブルT100の組合せ番号“1”では、式（1）に各値を代入することにより以下ようになる。まず、像域判定処理における処理単位となるブロック画像の横画素数は、8 × 600 / 150 = 32となる。また、前景画像の処理単位となるブロック画像の横画素数は、8 × 300 / 150 = 16となる。なお、背景画像の処理単位となるブロック画像の横画素数は、式（1）に関係なく8とする。

【0159】

また、データテーブルT100の組合せ番号“2”では、式（1）に各値を代入することにより以下ようになる。まず、像域判定処理における処理単位となるブロック画像の横画素数は、8 × 600 / 300 = 16となる。また、前景画像の処理単位となるブロック画像の横画素数は、8 × 600 / 300 = 16となる。なお、背景画像の処理単位となるブロック画像の横画素数は、式（1）に関係なく8とする。

【0160】

また、データテーブルT100の組合せ番号“3”では、式（1）に各値を代入することにより以下ようになる。まず、像域判定処理における処理単位となるブロック画像の横画素数は、8 × 600 / 150 = 32となる。また、前景画像の処理単位となるブロック画像の横画素数は、8 × 600 / 150 = 32となる。なお、背景画像の処理単位となるブロック画像の横画素数は、式（1）に関係なく8とする。

【0161】

また、データテーブルT100の組合せ番号“4”では、式（1）に各値を代入することにより以下ようになる。まず、像域判定処理における処理単位となるブロック画像の横画素数は、8 × 300 / 150 = 16となる。また、前景画像の処理単位となるブロック画像の横画素数は、8 × 300 / 150 = 16となる。なお、背景画像の処理単位となるブロック画像の横画素数は、式（1）に関係なく8とする。

【0162】

図7は、各画像の解像度と、各画像の処理単位となるブロック画像の相対的な大小関係を例示した図である。図7（A）は、データテーブルT100の組合せ番号“1”の各データに対応した、入力画像、前景画像および背景画像の各々の、解像度および処理単位となるブロック画像のサイズの関係を示す。

【0163】

図7（B）は、データテーブルT100の組合せ番号“2”の各データに対応した、入力画像、前景画像および背景画像の各々の、解像度および処理単位となるブロック画像の

10

20

30

40

50

サイズの関係を示す。

【 0 1 6 4 】

図 7 (C) は、データテーブル T 1 0 0 の組合せ番号 “ 3 ” の各データに対応した、入力画像、前景画像および背景画像の各々の、解像度および処理単位となるブロック画像のサイズの関係を示す。

【 0 1 6 5 】

図 7 (D) は、データテーブル T 1 0 0 の組合せ番号 “ 4 ” の各データに対応した、入力画像、前景画像および背景画像の各々の、解像度および処理単位となるブロック画像のサイズの関係を示す。

【 0 1 6 6 】

たとえば、データテーブル T 1 0 0 の組合せ番号 “ 1 ” では、入力画像、前景画像および背景画像の解像度は、それぞれ、6 0 0 d p i 、 3 0 0 d p i および 1 5 0 d p i である。

【 0 1 6 7 】

図 7 (A) を参照して、ブロック画像 1 0 G は、入力画像の処理単位となるブロック画像である。ブロック画像 1 0 G A は、前景画像の処理単位となるブロック画像である。ブロック画像 1 0 G B は、背景画像の処理単位となるブロック画像である。

【 0 1 6 8 】

ブロック画像 1 0 G 、 ブロック画像 1 0 G A およびブロック画像 1 0 G B のサイズは、それぞれ、3 2 画素 × 3 2 画素、1 6 画素 × 1 6 画素および 8 画素 × 8 画素とする。これにより、入力画像、前景画像および背景画像の間で、一つのブロック画像に対応する画像情報が、相対的に画像中の同じブロック画像部分を指し示すことになる。なお、データテーブル T 1 0 0 の組合せ番号が “ 2 ” ~ “ 4 ” についても、図 7 (B) 、 図 7 (C) 、 図 7 (D) に示すように、前述したデータテーブル T 1 0 0 の組合せ番号 “ 1 ” の場合と同様であるので詳細な説明は繰り返さない。

【 0 1 6 9 】

再び、図 2 を参照して、ステップ S 1 3 2 では、第 1 画像縮小部 1 0 2 が、たとえば、データテーブル T 1 0 0 の組合せ番号 “ 1 ” の各データおよびステップ S 1 0 1 により入力された入力ブロック画像データに基づいて、入力ブロック画像を縮小する。たとえば、ステップ S 1 0 1 により入力された入力ブロック画像のサイズが 3 2 画素 × 3 2 画素である場合、1 6 画素 × 1 6 画素のブロック画像に縮小する。縮小処理には、前述したように、ニアレストネイバー法、バイリニア法、バイキュービック法など、画像の解像度変換に一般的に用いられる既存の補間手法が使用される。

【 0 1 7 0 】

再び、図 3 を参照して、第 1 縮小画像 1 1 1 G は、第 1 画像縮小部 1 0 2 により、全ての入力ブロック画像が、画像縮小処理 A により、それぞれ縮小された複数の縮小ブロック画像から構成される画像である。第 1 縮小画像 1 1 1 G の解像度は、入力画像の解像度 (6 0 0 d p i) に対して、半分の解像度 (3 0 0 d p i) となっている。

【 0 1 7 1 】

ステップ S 1 3 2 (画像縮小処理 A) では、たとえば、入力画像 1 0 0 G 内の領域 1 0 0 R 内の画像が、第 1 縮小画像 1 1 1 G 内の領域 1 1 1 R 内の画像に縮小される。以下においては、画像縮小処理 A により縮小された画像を第 1 縮小ブロック画像ともいう。

【 0 1 7 2 】

再び、図 2 を参照して、ステップ S 1 3 2 (画像縮小処理 A) では、第 1 画像縮小部 1 0 2 が、生成した第 1 縮小ブロック画像のデータを、前景画像生成部 1 0 5 へ送信する。その後、ステップ S 1 4 0 に進む。

【 0 1 7 3 】

ステップ S 1 4 0 では、前景画像生成処理が行なわれる。前景画像生成処理では、ステップ S 1 2 0 により設定されたマスクデータに基づいて、ステップ S 1 3 2 で生成された第 1 縮小ブロック画像を使用して、前景画像を生成する。

10

20

30

40

50

【 0 1 7 4 】

図 8 は、前景画像生成処理のフローチャートである。図 8 を参照して、ステップ S 1 4 1 では、前景画像生成部 1 0 5 が、前述のマスク生成処理により、マスク生成部 1 0 4 が送信したマスクデータを受信する。その後、ステップ S 1 4 2 に進む。

【 0 1 7 5 】

ステップ S 1 4 2 では、第 1 画像縮小部 1 0 2 から送信された第 1 縮小ブロック画像が、前景画像として有効であるか否かが判定される。具体的には、受信したマスクデータが “ 1 ” であるか否かが、前景画像生成部 1 0 5 により判定される。マスクデータが “ 1 ” である場合、第 1 縮小ブロック画像は前景画像として有効となる。一方、マスクデータが “ 0 ” である場合、第 1 縮小ブロック画像は前景画像として無効となる。

10

【 0 1 7 6 】

ステップ S 1 4 2 において、YES ならば、ステップ S 1 4 4 A に進む。一方、ステップ S 1 4 2 において、NO ならば、ステップ S 1 4 4 B に進む。

【 0 1 7 7 】

ステップ S 1 4 4 A では、前景画像生成部 1 0 5 が、受信した第 1 縮小ブロック画像のデータを、前景画像符号化部 1 0 8 へ送信する。その後、この前景画像生成処理は終了し、図 2 の画像符号化処理に戻り、ステップ S 1 4 0 の次のステップ S 1 5 0 に進む。

【 0 1 7 8 】

ステップ S 1 4 4 B では、第 1 縮小ブロック画像と同じサイズの画像（たとえば、1 6 × 1 6 のサイズの画像）の全ての画素を黒（画素値 “ 0 ” ）とした画像（以下、黒ブロック画像ともいう）のデータを、前景画像符号化部 1 0 8 へ送信する。その後、この前景画像生成処理は終了し、図 2 の画像符号化処理に戻り、ステップ S 1 4 0 の次のステップ S 1 5 0 に進む。

20

【 0 1 7 9 】

再び、図 3 を参照して、前景画像 1 3 1 G は、前景画像生成部 1 0 5 が、前景画像符号化部 1 0 8 へ送信する全てのブロック画像のデータから構成される画像である。前景画像 1 3 1 G の解像度は、第 1 縮小画像 1 1 1 G の解像度と同じで、3 0 0 d p i である。前景画像生成処理により、前景画像として有効と判定された第 1 縮小ブロック画像に対応する、前景画像 1 3 1 G 内のブロック画像は、当該判定された第 1 縮小ブロック画像となる。一方、前景画像生成処理により、前景画像として無効と判定された第 1 縮小ブロック画像に対応する、前景画像 1 3 1 G 内のブロック画像は、ブロック画像内の全ての画素が黒（画素値 “ 0 ” ）となる。

30

【 0 1 8 0 】

ステップ S 1 4 4 A で送信される画像は、たとえば、前景画像 1 3 1 G 内の領域 1 3 1 R 内の画像である。また、ステップ S 1 4 4 B で送信される画像は、たとえば、前景画像 1 3 1 G 内の黒のブロック画像である。

【 0 1 8 1 】

再び、図 2 を参照して、ステップ S 1 5 0 では、前景画像符号化処理が行なわれる。前景画像符号化処理では、前景画像符号化部 1 0 8 が、前景画像生成部 1 0 5 から受信した画像（たとえば、1 6 × 1 6 のサイズの画像）を、非可逆符号化する。非可逆符号化を行なう符号化方式は、ブロック単位で処理を行なう J P E G 方式が使用される。なお、非可逆符号化を行なう符号化方式は、J P E G 方式に限定されることなく、ブロック単位で処理を行なう他の符号化方式であってもよい。前景画像符号化部 1 0 8 は、非可逆符号化したデータ（以下、前景画像ブロック符号化データともいう）を、フォーマット部 1 1 0 へ送信する。その後、ステップ S 1 5 2 に進む。

40

【 0 1 8 2 】

ステップ S 1 5 2 では、画像縮小処理 B が行なわれる。画像縮小処理 B では、第 2 画像縮小部 1 0 3 が、たとえば、図 6 のデータテーブル T 1 0 0 の組合せ番号 “ 1 ” の各データおよびステップ S 1 0 1 により入力された入力ブロック画像データに基づいて、入力ブロック画像を縮小する。たとえば、ステップ S 1 0 1 により入力された入力ブロック画像

50

のサイズが 32 画素 × 32 画素である場合、8 画素 × 8 画素のブロック画像に縮小する。縮小処理には、前述したように、ニアレストネイバー法、バイリニア法、バイキュービック法など、画像の解像度変換に一般的に用いられる既存の補間手法が使用される。

【0183】

再び、図 3 を参照して、第 2 縮小画像 112 G は、第 2 画像縮小部 103 により、全ての入力ブロック画像が、それぞれ縮小された複数の縮小ブロック画像から構成される画像である。第 2 画像縮小部 103 の解像度は、入力画像の解像度 (600 dpi) に対して、4 分の 1 の解像度 (150 dpi) となっている。

【0184】

ステップ S152 (画像縮小処理 B) では、たとえば、入力画像 100 G 内の領域 100 R 内の画像が、第 2 縮小画像 112 G 内の領域 112 R 内の画像 (たとえば、8 画素 × 8 画素のサイズの画像) に縮小される。以下においては、画像縮小処理 B により縮小された画像を第 2 縮小ブロック画像 (たとえば、8 画素 × 8 画素のサイズの画像) ともいう。

【0185】

再び、図 2 を参照して、ステップ S152 (画像縮小処理 B) では、第 2 画像縮小部 103 が、生成した第 2 縮小ブロック画像のデータを、背景画像生成部 106 へ送信する。その後、ステップ S160 に進む。

【0186】

ステップ S160 では、背景画像生成処理が行なわれる。背景画像生成処理では、ステップ S120 により設定されたマスクデータに基づいて、ステップ S152 で生成された第 2 縮小ブロック画像を使用して、背景画像を生成する。

【0187】

図 9 は、背景画像生成処理のフローチャートである。図 9 を参照して、ステップ S161 では、背景画像生成部 106 が、前述のマスク生成処理により、マスク生成部 104 が送信したマスクデータを受信する。その後、ステップ S162 に進む。

【0188】

ステップ S162 では、第 2 画像縮小部 103 から送信された第 2 縮小ブロック画像が、背景画像として有効であるか否かが判定される。具体的には、受信したマスクデータが “0” であるか否かが、背景画像生成部 106 により判定される。マスクデータが “0” である場合、第 2 縮小ブロック画像は背景画像として有効となる。一方、マスクデータが “1” である場合、第 2 縮小ブロック画像は背景画像として無効となる。

【0189】

ステップ S162 において、YES ならば、ステップ S164 A に進む。一方、ステップ S162 において、NO ならば、ステップ S164 B に進む。

【0190】

ステップ S164 A では、背景画像生成部 106 が、受信した第 2 縮小ブロック画像のデータを、背景画像符号化部 109 へ送信する。その後、この背景画像生成処理は終了し、図 2 の画像符号化処理に戻り、ステップ S160 の次のステップ S170 に進む。

【0191】

ステップ S164 B では、第 2 縮小ブロック画像と同じサイズの画像 (たとえば、8 × 8 のサイズの画像) の全ての画素を黒 (画素値 “0”) とした画像 (以下、黒ブロック画像ともいう) のデータを、背景画像符号化部 109 へ送信する。その後、この背景画像生成処理は終了し、図 2 の画像符号化処理に戻り、ステップ S160 の次のステップ S170 に進む。

【0192】

再び、図 3 を参照して、背景画像 132 G は、背景画像生成部 106 が、背景画像符号化部 109 へ送信する全てのブロック画像のデータから構成される画像である。背景画像 132 G の解像度は、第 2 縮小画像 112 G の解像度と同じで、150 dpi である。背景画像生成処理により、前景画像として無効と判定された第 2 縮小ブロック画像に対応する、背景画像 132 G 内のブロック画像は、当該判定された第 2 縮小ブロック画像となる

10

20

30

40

50

。一方、前景画像生成処理により、前景画像として有効と判定された第2縮小ブロック画像に対応する、背景画像132G内のブロック画像は、ブロック画像内の全ての画素が黒（画素値“0”）となる。ステップS164Aで送信される画像は、たとえば、背景画像132G内の領域132R内の画像である。また、ステップS164Bで送信される画像は、たとえば、背景画像132G内の黒のブロック画像である。

【0193】

再び、図2を参照して、ステップS170では、背景画像符号化処理が行なわれる。背景画像符号化処理では、背景画像符号化部109が、背景画像生成部106から受信した画像（たとえば、8×8のサイズの画像）を、非可逆符号化する。非可逆符号化を行なう符号化方式は、ブロック単位で処理を行なうJPEG方式が使用される。なお、非可逆符号化を行なう符号化方式は、JPEG方式に限定されることなく、ブロック単位で処理を行なう他の符号化方式であってもよい。背景画像符号化部109は、非可逆符号化したデータ（以下、背景画像ブロック符号化データともいう）を、フォーマット部110へ送信する。その後、ステップS172に進む。

10

【0194】

ステップS172では、入力画像全体に対し処理が終了したか否かが判定される。具体的には、制御部120が、ステップS101の処理で読み出された入力ブロック画像が、入力画像内の右下のブロック画像（領域100RF内の画像）であるか否かを判定する。

【0195】

ステップS172において、YESならば、ステップS174に進む。一方、ステップS172において、NOならば、再度、ステップS101の処理が繰り返される。

20

【0196】

ステップS174では、マスク符号化処理が行なわれる。マスク符号化処理では、マスク符号化部107が、ステップS120が繰り返し行なわれることで生成された、複数のマスクデータ（2値データ）から構成されるマスク層データを、一時記憶部130から読み出す。複数のマスクデータは、入力画像内の複数のブロック画像にそれぞれ対応するデータである。そして、マスク符号化部107が、複数のマスクデータからなるマスク層データを可逆符号化する。

【0197】

可逆符号化では、2値画像の可逆符号化に適した符号化方式であるMMRが使用される。なお、可逆符号化は、MMRに限定されることなく、2値画像の可逆符号化に適した符号化方式（たとえば、前述したMR、JBIG等）であればどの符号化方式であってもよい。マスク符号化部107は、符号化した符号化データ（以下、マスク符号化データともいう）を、フォーマット部110へ送信する。その後、ステップS180に進む。

30

【0198】

ステップS180では、フォーマット処理が行なわれる。フォーマット処理では、フォーマット部110が、受信したマスク符号化データと、複数の前景画像ブロック符号化データと、複数の背景画像ブロック符号化データとの関連付けを行なう。当該関連付けは、汎用性の高いデータフォーマットを使用して行なわれる。

【0199】

ここで、複数の前景画像ブロック符号化データは、ステップS150の処理が繰り返し行なわれることで、前景画像符号化部108から受信した複数のデータである。以下においては、複数の前景画像ブロック符号化データから構成されるデータを前景画像符号化データともいう。

40

【0200】

また、複数の背景画像ブロック符号化データは、ステップS170の処理が繰り返し行なわれることで、背景画像符号化部109から受信した複数のデータである。以下においては、複数の背景画像ブロック符号化データから構成されるデータを背景画像符号化データともいう。

【0201】

50

汎用性の高いデータフォーマットの一例としては、Adobe (登録商標) 社の PDF である。なお、汎用性の高いデータフォーマットは、Adobe (登録商標) 社の PDF に限定されることなく、他のデータフォーマットであってもよい。

【0202】

たとえば、マスク符号化データがITU-T勧告T.6に規定されるMMR形式であり、また、前景画像符号化データおよび背景画像符号化データがJPEG形式である場合、フォーマット部110は、以下の関連付け処理を行なう。

【0203】

関連付け処理では、フォーマット部110が、Adobe (登録商標) 社の PDF 形式に基づいて、マスク符号化データ、前景画像符号化データ、背景画像符号化データを関連付けた関連付けデータを作成する。そして、フォーマット部110は、マスク符号化データ、前景画像符号化データ、背景画像符号化データおよび関連付けデータを一つの符号化画像データとして、前述した汎用統合データを生成する。

10

【0204】

汎用統合データを復号化する場合は、たとえば、前景画像符号化データをJPEGにより復号化した画像(以下、復号化前景画像ともいう)と、背景画像符号化データをJPEGにより復号化した画像(以下、復号化背景画像ともいう)と、マスク符号化データを復号化することによって得られる複数のマスクデータとを使用して行なわれる。

【0205】

具体的には、まず、復号化前景画像と、復号化背景画像とのサイズが異なる場合、復号化背景画像を、復号化前景画像と同じサイズに拡大する処理が行なわれる。そして、復号化背景画像を最下層の画像とし、“1”に設定されたマスクデータに対応する復号化前景画像内のブロック画像を、復号化背景画像内の対応する位置に上書きする。上記処理を繰返し行なうことにより、汎用統合データを復号化して、符号化する前の入力画像を得ることができる。

20

【0206】

なお、前述の説明では、ステップS101～ステップS170までの処理をブロック画像単位で、一連の処理とすることを前提に説明した。しかしながら、本発明の一つの特徴であるブロック画像単位での像域判定が実施され、マスク層データが可逆符号化、前景画像および背景画像が非可逆符号化されていれば、前述の処理の順に限定されるものではない。

30

【0207】

たとえば、入力画像全体について、まず、マスク生成処理、前景画像生成処理、背景画像生成処理を行ない、各処理により得られたデータを、一時記憶部130に記憶させる。その後、マスク層データ、前景画像、背景画像を符号化してもよい。すなわち、ステップS150、S170の処理を、ステップS172の後で、かつ、ステップS180の前に行なうようにしてもよい。

【0208】

また、ステップS180のフォーマット処理をブロック画像単位で処理するようにしてもよい。すなわち、ステップS180の処理を、ステップS170の後で、かつ、ステップS172の前に行なうようにしてもよい。

40

【0209】

また、処理を並列化して同時に複数の処理がなされるようにしてもよい。たとえば、ステップS132～S150の処理と、ステップS152～S170の処理とを、ステップS120の後に並列に行なってもよい。

【0210】

また、ステップS174を、ステップS120の後で、かつ、ステップS172の前に配置し、マスクデータを所定数(たとえば2ライン、すなわち、「入力画像に対応する横方向ブロック数」×2)蓄積して、1ライン分更新が進むごとにステップS174の処理を実行するようにしてもよい。

50

【 0 2 1 1 】

また、ステップ S 1 3 2 の画像縮小処理 A と、ステップ S 1 5 2 の画像縮小処理 B とを、ステップ S 1 0 1 で読み出すブロック画像単位ではなく、異なる処理単位で行なってもよい。

【 0 2 1 2 】

以上説明したように、本実施の形態では、処理対象となる入力画像を複数の層で別々に処理する。複数の層は、マスク層、前景層、背景層からなる。前景層では、入力画像または入力画像を縮小した画像に対し、文字、線またはエッジ部分を含む部分のブロック画像を非可逆符号化し、前景画像を生成する。

【 0 2 1 3 】

背景層では、入力画像または入力画像を縮小した画像に対し、文字、線およびエッジ部分のいずれも含まない部分のブロック画像を非可逆符号化し、背景画像を生成する。文字、線またはエッジ部分が含まれる前景画像は、背景画像以上の解像度を有する。

【 0 2 1 4 】

したがって、文字、線またはエッジ部分の画質の劣化を最小限に抑えつつ、データ容量を大幅に削減することができるという効果を奏する。

【 0 2 1 5 】

また、本実施の形態では、複数の層の画像データおよびデータを汎用性の高いフォーマットに従って関連付けし、1つのデータにする。したがって、汎用性の高いデータを作成することができるという効果を奏する。

【 0 2 1 6 】

また、本実施の形態では、所定のブロック画像単位で、処理対象のブロック画像が、文字または線が存在するか否かを判定する。また、所定のブロック画像単位で、非可逆符号化処理（たとえば、J P E G による符号化処理）を行なう。当該非可逆符号化処理は、ブロック単位で処理を行なう。

【 0 2 1 7 】

したがって、不要なブロック画像内のドント・ケア画素の画素値に起因する色にじみを防ぐことができる。その結果、不要なブロック画像に設定する画素値に関わらず、再生画像を一定の品質に保つことができるという効果を奏する。

【 0 2 1 8 】

< 第 2 の実施の形態 >

第 2 の実施の形態では、マスク符号化データを使用せずに汎用統合データを生成する点が、第 1 の実施の形態と異なる。マスク符号化データの代わりに、前景画像に透過色を設定することにより、マスク符号化データを使用する必要がなくなる。そのため、第 1 の実施の形態よりも、汎用統合データのデータ容量を小さくすることができる。

【 0 2 1 9 】

図 1 0 は、第 2 の実施の形態における画像符号化装置 1 0 0 0 A の構成を示すブロック図である。図 1 0 を参照して、画像符号化装置 1 0 0 0 A は、図 1 の画像符号化装置 1 0 0 0 と比較して、画像処理部 1 0 0 の代わりに画像処理部 1 0 0 A を備える点が異なる。それ以外の構成は、画像符号化装置 1 0 0 0 と同様なので詳細な説明は繰り返さない。

【 0 2 2 0 】

画像処理部 1 0 0 A は、図 1 の画像処理部 1 0 0 と比較して、マスク符号化部 1 0 7 を含まない点と、透過色設定部 1 1 2 をさらに含む点とが異なる。なお、画像処理部 1 0 0 A は、画像処理部 1 0 0 と同様、記憶部 1 4 0 に記憶された画像符号化プログラム 7 2 に従って、後述の処理を行なう。

【 0 2 2 1 】

透過色設定部 1 1 2 は、前景画像生成部 1 0 5 と、背景画像生成部 1 0 6 とに接続される。前景画像生成部 1 0 5 は、透過色を設定すべきブロック画像の情報を、透過色設定部 1 1 2 へ送信する。透過色設定部 1 1 2 は、透過色を設定すべきブロック画像の情報を、前景画像生成部 1 0 5 から受信する。透過色設定部 1 1 2 は、透過色を設定したブロック

10

20

30

40

50

画像の情報を前景画像生成部 1 0 5 および背景画像生成部 1 0 6 へ送信する。

【 0 2 2 2 】

フォーマット部 1 1 0 は、前景画像符号化部 1 0 8 から、前景画像ブロック符号化データ A を受信する。前景画像ブロック符号化データ A は、前述の前景画像ブロック符号化データと比較して、透過色を示すデータを含む点が異なる。透過色の表現については、たとえば、透過色に設定する画素の画素値を “ 0 ” で表現し、画素値 “ 0 ” が透過色であることを示すデータを別途設けることによって実現できる。なお、ある画素値が透過色であることを示すことは、たとえば、A d o b e (登録商標) 社の P D F によって表現可能である。上記以外の画像処理部 1 0 0 A の構成は、画像処理部 1 0 0 と同様なので詳細な説明は繰り返さない。

10

【 0 2 2 3 】

透過色設定部 1 1 2 は、透過色を設定すべきブロック画像中の各画素に対して、透過色の色情報（たとえば、画素値 “ 0 ” ）を設定する。なお、透過色設定部 1 1 2 は、たとえば、専用の L S I によって構成される。

【 0 2 2 4 】

また、前述のブロック像域判定部 1 0 1、第 1 画像縮小部 1 0 2、第 2 画像縮小部 1 0 3、マスク生成部 1 0 4、前景画像生成部 1 0 5、背景画像生成部 1 0 6、前景画像符号化部 1 0 8、背景画像符号化部 1 0 9、フォーマット部 1 1 0 および透過色設定部 1 1 2 の各々が行なう処理は、専用の L S I ではなく、パーソナルコンピュータなどの一般的なコンピュータのマイクロプロセッサまたは制御部 1 2 0 により実行されてもよい。

20

【 0 2 2 5 】

この場合、たとえば、後述する画像符号化処理 A をコンピュータまたは制御部 1 2 0 に実行させるためのプログラムとして記述してもよい。当該プログラムは、画像符号化プログラム 7 2 であり、記録媒体 7 0 に記録されて流通する。

【 0 2 2 6 】

コンピュータにより、後述する画像符号化処理 A が実行される場合、画像符号化プログラム 7 2 は、記録媒体 7 0 に記録され、コンピュータが備える記録媒体アクセス部により、コンピュータが備える記憶部に読み込まれ、C P U で実行される。

【 0 2 2 7 】

制御部 1 2 0 により、後述する画像符号化処理 A が実行される場合、画像符号化プログラム 7 2 は、記録媒体 7 0 に記録され、制御部 1 2 0 により実行される。

30

【 0 2 2 8 】

さらに、画像符号化プログラム 7 2 は、インターネットなどのネットワークを介して他の装置からダウンロードされてもよい。

【 0 2 2 9 】

また、画像処理部 1 0 0 A に含まれる、前述した複数の専用の L S I とコンピュータのマイクロプロセッサとを組み合わせてもよい。また、前述した複数の専用の L S I の一部または全てを一つの専用の L S I として構成してもよい。

【 0 2 3 0 】

次に、第 2 の実施の形態における画像符号化装置 1 0 0 0 A の具体的な処理内容について説明する。

40

【 0 2 3 1 】

図 1 1 は、第 2 の実施の形態において、画像符号化装置 1 0 0 0 A で実行される画像符号化処理 A のフローチャートを示す。図 1 1 を参照して、画像符号化処理 A は、図 2 の画像符号化処理と比較して、ステップ S 1 4 0 の代わりに、ステップ S 1 4 0 A の処理が行なわれる点と、ステップ S 1 8 0 の代わりに、ステップ S 1 8 0 A の処理が行なわれる点と、ステップ S 1 7 4 の処理が行なわれない点と異なる。以下、異なる部分を主に説明する。

【 0 2 3 2 】

まず、ステップ S 1 0 1 では、前述したのと同様、画像処理部 1 0 0 A が、記憶部 1 4

50

0に記憶されている入力画像データを、所定のブロック単位で読み出す。なお、入力画像データの読出す順序は、前述したステップS101の処理と同様なので詳細な説明は繰り返さない。

【0233】

図12は、第2の実施の形態において、画像符号化装置1000Aで処理される画像を示す図である。なお、図12に示される各画像の解像度を忠実に図にて表現することは困難である。そのため、図12では、前述した図6のデータテーブルT100の組合せ番号“1”に対応する解像度と、ブロックサイズの関係性を想定して、相対的な関係を例示する。

【0234】

図12を参照して、入力画像100Gは、入力画像データに基づく画像である。入力画像100Gについては、前述したので詳細な説明は繰り返さない。

【0235】

再び、図11を参照して、読み出された所定のブロック単位の入力画像データは、ブロック像域判定部101、第1画像縮小部102および第2画像縮小部103へ入力される。その後、ステップS110に進む。

【0236】

ステップS110では、前述したのと同様、像域判定処理が行なわれる。

再び、図12を参照して、画像110Gは、像域判定処理により設定された線分ブロックデータの状態を視覚的に分かりやすく示した画像である。画像110Gについては、前述したので詳細な説明は繰り返さない。

【0237】

再び、図11を参照して、ステップS110の処理の後、ステップS120の処理が行なわれる。

【0238】

ステップS120では、前述したのと同様、マスク生成処理が行なわれる。なお、本実施の形態におけるマスク生成処理では、第1の実施の形態のマスク生成処理のうち、マスク生成部104が、マスクデータを、マスク符号化部107へ送信する処理は行なわれない。

【0239】

再び、図12を参照して、画像120Gは、マスク生成処理により設定されたマスクデータの状態を視覚的に分かりやすく示した画像である。画像120Gについては、前述したので詳細な説明は繰り返さない。

【0240】

再び、図11を参照して、ステップS120の処理の後、ステップS132の処理が行なわれる。

【0241】

ステップS132では、前述したのと同様、画像縮小処理Aが行なわれる。

再び、図12を参照して、第1縮小画像111Gは、第1画像縮小部102により、全ての入力ブロック画像が、画像縮小処理Aにより、それぞれ縮小された複数の縮小ブロック画像から構成される画像である。第1縮小画像111Gについては、前述したので詳細な説明は繰り返さない。

【0242】

再び、図11を参照して、ステップS132（画像縮小処理A）では、第1画像縮小部102が、生成した第1縮小ブロック画像のデータを、前景画像生成部105へ送信する。その後、ステップS140Aに進む。

【0243】

ステップS140Aでは、前景画像生成処理Aが行なわれる。前景画像生成処理Aは、ステップS120により設定されたマスクデータに基づいて、ステップS132で生成された第1縮小ブロック画像を使用して、前景画像を生成する。

10

20

30

40

50

【 0 2 4 4 】

図 1 3 は、前景画像生成処理 A のフローチャートである。図 1 3 を参照して、ステップ S 2 4 1 では、前景画像生成部 1 0 5 が、前述のマスク生成処理により、マスク生成部 1 0 4 が送信したマスクデータを受信する。その後、ステップ S 2 4 2 に進む。

【 0 2 4 5 】

ステップ S 2 4 2 では、第 1 画像縮小部 1 0 2 から送信された第 1 縮小ブロック画像が、前景画像として有効であるか否かが判定される。具体的には、受信したマスクデータが “ 1 ” であるか否かが、前景画像生成部 1 0 5 により判定される。マスクデータが “ 1 ” である場合、第 1 縮小ブロック画像は前景画像として有効となる。一方、マスクデータが “ 0 ” である場合、第 1 縮小ブロック画像は前景画像として無効となる。

10

【 0 2 4 6 】

ステップ S 2 4 2 において、YES ならば、ステップ S 2 4 4 A に進む。一方、ステップ S 2 4 2 において、NO ならば、ステップ S 2 4 4 B に進む。

【 0 2 4 7 】

ステップ S 2 4 4 A では、前景画像生成部 1 0 5 が、受信した第 1 縮小ブロック画像のデータを、前景画像符号化部 1 0 8 へ送信する。その後、この前景画像生成処理は終了し、図 1 1 の画像符号化処理 A に戻り、ステップ S 1 4 0 A の次のステップ S 1 5 0 に進む。

【 0 2 4 8 】

ステップ S 2 4 4 B では、第 1 縮小ブロック画像と同じサイズの画像（たとえば、1 6 × 1 6 のサイズの画像）の全ての画素を透過色（画素値 “ 0 ” ）とした画像（以下、透過色ブロック画像ともいう）のデータを、前景画像符号化部 1 0 8 へ送信する。なお、透過色の画素値（以下、透過色値ともいう）は、後述する背景画像生成処理で生成される黒ブロック画像の画素値と同一の値である。

20

【 0 2 4 9 】

その後、透過色設定部 1 1 2 が、前景画像の画素値 “ 0 ” が透過色であることを示す情報を、一時記憶部 1 3 0 に記憶させることにより設定する。その後、この前景画像生成処理 A は終了し、図 1 1 の画像符号化処理 A に戻り、ステップ S 1 4 0 A の次のステップ S 1 5 0 に進む。

【 0 2 5 0 】

再び、図 1 2 を参照して、前景画像 1 3 1 G A は、前景画像生成部 1 0 5 が、前景画像符号化部 1 0 8 へ送信する全てのブロック画像のデータから構成される画像である。前景画像 1 3 1 G A の解像度は、第 1 縮小画像 1 1 1 G の解像度と同じで、3 0 0 d p i である。前景画像生成処理 A により、前景画像として有効と判定された第 1 縮小ブロック画像に対応する、前景画像 1 3 1 G A 内のブロック画像は、当該判定された第 1 縮小ブロック画像となる。一方、前景画像生成処理 A により、前景画像として無効と判定された第 1 縮小ブロック画像に対応する、前景画像 1 3 1 G A 内のブロック画像は、ブロック画像内の全ての画素が透過色（画素値 “ 0 ” ）となる。なお、前景画像 1 3 1 G A は、透過色を示す画素値 “ 0 ” に対応する部分を黒色で表現している。

30

【 0 2 5 1 】

ステップ S 2 4 4 A で送信される画像は、たとえば、前景画像 1 3 1 G A 内の領域 1 3 1 R A 内の画像である。また、ステップ S 2 4 4 B で送信される画像は、たとえば、前景画像 1 3 1 G A 内の透過色（黒色）のブロック画像である。

40

【 0 2 5 2 】

再び、図 1 1 を参照して、ステップ S 1 5 0 では、前述したのと同様、前景画像符号化処理が行なわれる。その後、ステップ S 1 5 2 に進む。

【 0 2 5 3 】

ステップ S 1 5 2 では、前述したのと同様、画像縮小処理 B が行なわれる。

再び、図 1 2 を参照して、第 2 縮小画像 1 1 2 G は、第 2 画像縮小部 1 0 3 により、全ての入力ブロック画像が、それぞれ縮小された複数の縮小ブロック画像から構成される画

50

像である。第2縮小画像112Gについては、前述したので詳細な説明は繰り返さない。

【0254】

再び、図11を参照して、ステップS152（画像縮小処理B）では、第2画像縮小部103が、生成した第2縮小ブロック画像のデータを、背景画像生成部106へ送信する。その後、ステップS160に進む。

【0255】

ステップS160では、前述したのと同様、背景画像生成処理が行なわれる。

再び、図12を参照して、背景画像132Gは、背景画像生成部106が、背景画像符号化部109へ送信する全てのブロック画像のデータから構成される画像である。背景画像132Gの解像度は、第2縮小画像112Gの解像度と同じで、150dpiである。背景画像生成処理により、前景画像として無効と判定された第2縮小ブロック画像に対応する、背景画像132G内のブロック画像は、当該判定された第2縮小ブロック画像となる。一方、前景画像生成処理により、前景画像として有効と判定された第2縮小ブロック画像に対応する、背景画像132G内のブロック画像（黒ブロック画像）は、ブロック画像内の全ての画素が黒（画素値“0”）となる。

10

【0256】

再び、図11を参照して、ステップS160の処理の後、ステップS170に進む。

ステップS170では、前述したのと同様、背景画像符号化処理が行なわれる。その後、ステップS172に進む。

【0257】

ステップS172では、前述したのと同様、入力画像全体に対し処理が終了したか否かが判定される。ステップS172において、YESならば、ステップS180Aに進む。一方、ステップS172において、NOならば、再度、ステップS101の処理が繰り返される。

20

【0258】

ステップS180Aでは、フォーマット処理Aが行なわれる。フォーマット処理Aでは、フォーマット部110が、複数の前景画像ブロック符号化データと、複数の背景画像ブロック符号化データとの関連付けを行なう。当該関連付けは、汎用性の高いデータフォーマットを使用して行なわれる。

【0259】

ここで、複数の前景画像ブロック符号化データは、前述したので詳細な説明は繰り返さない。前述したように、複数の前景画像ブロック符号化データから構成されるデータを前景画像符号化データともいう。

30

【0260】

また、複数の背景画像ブロック符号化データは、前述したので詳細な説明は繰り返さない。前述したように、複数の背景画像ブロック符号化データから構成されるデータを背景画像符号化データともいう。

【0261】

汎用性の高いデータフォーマットの一例としては、Adobe（登録商標）社のPDFである。なお、汎用性の高いデータフォーマットは、Adobe（登録商標）社のPDFに限定されることなく、他のデータフォーマットであってもよい。

40

【0262】

たとえば、前景画像符号化データおよび背景画像符号化データがJPEG形式である場合、フォーマット部110は、以下の関連付け処理Aを行なう。

【0263】

関連付け処理Aでは、フォーマット部110が、Adobe（登録商標）社のPDF形式に基づいて、前景画像符号化データおよび背景画像符号化データを関連付けた関連付けデータAを作成する。そして、フォーマット部110は、前景画像符号化データ、背景画像符号化データおよび関連付けデータAを一つの符号化画像データとして、汎用統合データAを生成する。

50

【 0 2 6 4 】

汎用統合データ A を復号化する場合は、たとえば、前景画像符号化データを J P E G により復号化した復号化前景画像と、背景画像符号化データを J P E G により復号化した復号化背景画像とを使用して行なわれる。

【 0 2 6 5 】

具体的には、まず、復号化前景画像と、復号化背景画像とのサイズが異なる場合、復号化背景画像を、復号化前景画像と同じサイズに拡大する処理が行なわれる。そして、復号化前景画像に対し、透過色が設定された復号化前景画像内のブロック画像に対応する、復号化背景画像内のブロック画像を、復号化前景画像内の対応する位置に上書きする。上記処理を繰返し行なうことにより、汎用統合データ A を復号化して、符号化する前の入力画像を得ることができる。

10

【 0 2 6 6 】

なお、前述の説明では、ステップ S 1 0 1 ~ ステップ S 1 7 0 までの処理をブロック画像単位で、一連の処理とすることを前提に説明した。しかしながら、本発明の一つの特徴であるブロック画像単位での像域判定が実施され、前景画像および背景画像が非可逆符号化されていれば、前述の処理の順に限定されるものではない。

【 0 2 6 7 】

たとえば、入力画像全体について、まず、前景画像生成処理、背景画像生成処理を行ない、各処理により得られたデータを、一時記憶部 1 3 0 に記憶させる。その後、前景画像、背景画像を符号化してもよい。すなわち、S 1 5 0、S 1 7 0 の処理を、ステップ S 1 7 2 の後で、かつ、ステップ S 1 8 0 A の前に行なうようにしてもよい。

20

【 0 2 6 8 】

また、ステップ S 1 8 0 A のフォーマット処理 A をブロック画像単位で処理するようにしてもよい。すなわち、ステップ S 1 8 0 A の処理を、ステップ S 1 7 0 の後で、かつ、ステップ S 1 7 2 の前に行なうようにしてもよい。

【 0 2 6 9 】

また、処理を並列化して同時に複数の処理がなされるようにしてもよい。たとえば、ステップ S 1 3 2 ~ S 1 5 0 の処理と、ステップ S 1 5 2 ~ S 1 7 0 の処理とを、ステップ S 1 2 0 の後に並列に行なってもよい。

30

【 0 2 7 0 】

また、ステップ S 1 3 2 の画像縮小処理 A と、ステップ S 1 5 2 の画像縮小処理 B とを、ステップ S 1 0 1 で読み出すブロック画像単位ではなく、異なる処理単位で行なってもよい。

【 0 2 7 1 】

以上説明したように、本実施の形態では、マスク符号化データを使用せずに汎用統合データ A を生成するため、第 1 の実施の形態の奏する効果に加え、汎用統合データ A のデータ容量をさらに小さくすることができるという効果を奏する。

(画像の色空間)

40

なお、本発明に従う上述の実施の形態は、入力画像がカラー画像の場合において特に効果を発揮するが、グレースケール (輝度成分のみ) 画像などに適用することも可能である。

(無効なブロック内の画素値)

第 1 の実施の形態においては、無効なブロック (複数のドント・ケア画素から構成されるブロック画像) の画素値について、符号化効率が向上するように任意の値を設定することが可能である。たとえば、ある無効ブロックの画素値は “ 0 ” に設定し、他の無効ブロックの画素値は “ 2 5 5 ” に設定することが可能である。

【 0 2 7 2 】

しかしながら、本発明の第 2 の実施の形態においては、無効なブロック (複数のドント

50

・ケア画素から構成されるブロック画像)の画素値について、以下のような注意が必要である。

【0273】

すなわち、透過色を、画像の表現に必要な画素値(たとえば、画素値“0”)に割り当てて表現する場合、有効なブロック画像中の画素値“0”についても透過色として扱われる可能性があるため、全ての層(マスク層、前景層、背景層)のデータまたは画像において、無効なブロック画素の画素値として“0”を設定しておくことが好ましい。

【0274】

なお、透過色を、画像の表現に必要な画素値以外の値で表現することが可能である場合には、無効なブロック画像中の画素値について、符号化効率が向上するような任意の値を設定することが可能である。

【0275】

第1の実施の形態および第2の実施の形態のいずれにおいても、上述のように無効なブロック画素中の画素値について任意の値を設定する場合、以下のようにすることができる。たとえば、非可逆符号化をJPEG方式を用いて行なう場合、全ての無効なブロック画素中の全ての画素について、画素値“0”を設定すれば、複雑な処理を実行することなく、一般的に圧縮効率を高めることが可能である。

(マスク生成基準)

また、上述の説明では、処理対象のブロック画像が、前景画像として有効または無効であるかを示すマスクデータは、像域判定処理により設定された線分ブロックデータに基づいて生成される。線分ブロックデータは、処理対象のブロック画像が、文字・線を含む線分ブロック画像であるか否かを示すデータである。なお、線分ブロック画像は、文字、線に加え、エッジ部分を含む画像であってもよい。すなわち、線分ブロックデータは、処理対象のブロック画像が、文字、線またはエッジ部分を含む線分ブロック画像であるか否かを示すデータであってもよい。このとき、エッジ部分の有無の判定には、たとば、Sobelフィルタなどによる公知のエッジ抽出方法を用いればよい。

(画像縮小処理を行わない場合)

また、上述の説明では、前景画像および背景画像を縮小する場合について説明した。しかしながら、前景画像および背景画像のどちらも縮小しない場合、あるいは前景画像および背景画像のいずれかのみを縮小する場合も考えられる。

【0276】

すなわち、図2の画像符号化処理および図11の画像符号化処理Aにおいて、ステップS132(画像縮小処理A)およびステップS152(画像縮小処理B)のいずれか1つ、または両方の処理を行わなくてもよい。

【0277】

たとえば、図6に示した画像の解像度とブロックサイズとの関係を示すデータテーブルT100に示される、組合せ番号“2”、“3”、“4”のいずれかに対応するデータで、処理が行なわれる場合、入力画像の解像度と前景画像の解像度は同じとなる。したがって、画像符号化処理および画像符号化処理Aにおいて、ステップS132(画像縮小処理A)を省略することができる。

【0278】

また、本発明は、画像符号化装置1000および画像符号化装置1000Aにおいて、第1画像縮小部102および第2画像縮小部103を省略しても実現可能である。この場合、入力画像、前景画像および背景画像の解像度が全て同じとなる。そして、ブロック画像単位の像域情報(線分ブロックデータ)に基づいて前景画像および背景画像を生成する。これにより、像域の特徴に応じた適切な符号化パラメータ(たとえば、JPEG方式における量子化パラメータなど)を用いて符号化することが可能である。

【0279】

前述の通り、画像符号化装置1000および画像符号化装置1000Aにおいて、第1画像縮小部102および第2画像縮小部103のいずれか1つまたは両方を省略した場合

10

20

30

40

50

、画像符号化処理および画像符号化処理Aにおいて、対応する処理（たとえば、ステップS132の画像縮小処理Aなど）は、適宜省略すればよい。

（3層以上の場合）

また、前述の説明では一つの入力画像を、ブロック画像単位で、前景層と背景層の二つの層に分類することを前提としているが、本発明は二つの層に分類する場合のみに留まらず、三つ以上の層に分類する場合においても同様に実施することが可能である。

【0280】

たとえば、文字または線またはエッジを表現する層（前景画像A）、写真を表現する層（前景画像B）、その他の成分を表現する層（背景画像）の三つに分類する場合においても、分類する単位がブロック画像単位であれば、本発明の主旨に則って処理することができる。

10

【0281】

たとえば、ステップS110の像域判定処理において、文字、線またはエッジを含む像域、写真の像域、その他の像域の三つの像域に分類して、像域判定結果を示す像域判定データとして三種の値のいずれかを示し得るようにすればよい。たとえば、“0”は文字、線またはエッジ、“1”は写真、“2”はその他を示すようにすればよい。

【0282】

また、ステップS120のマスク生成処理において、たとえば、前景画像Aについては、像域判定データに基づいて、“0”に対応するブロック画像は“前景画像Aとして有効”、“1”または“2”に対応するブロック画像は“前景画像Aとして無効”と設定して、前景画像A用のマスク層データを生成すればよい。

20

【0283】

さらに、前景画像Bについては、像域判定データに基づいて、“1”に対応するブロック画像は“前景画像Bとして有効”、“0”または“2”に対応するブロック画像は“前景画像Bとして無効”と設定して、前景画像B用のマスク層データを生成すればよい。なお、N種類の像域を設定した場合についても同様に、N-1個のマスク層データを生成すればよい。

【0284】

また、ステップS140の前景画像生成処理またはステップS140Aの前景画像生成処理Aでは、たとえば、前景画像Aの有効無効情報に基づいて、前景画像Aを生成し、前景画像Bの有効無効情報に基づいて、前景画像Bを生成すればよい。なお、N種類の像域を設定した場合についても同様に、N-1個の前景画像を生成すればよい。

30

【0285】

また、ステップS160の背景画像生成処理において、処理対象のブロック画像が、たとえば、“前景画像Aとして無効”で且つ“前景画像Bとして無効”であるブロック画像は、背景画像として有効なブロック画像として第2縮小ブロック画像データを設定する。処理対象のブロック画像が、“前景画像Aとして有効”または“前景画像Bとして有効”のどちらか一つでも満たす場合、背景画像として無効なブロック画像として、たとえば、第2縮小ブロック画像と同じサイズの画像の全ての画素値を“0”に設定すればよい。

【0286】

なお、N種類の像域を設定した場合においても同様に、処理対象のブロック画像が、N-1個の前景画像全てにおいて無効な場合、背景画像として有効なブロック画像とする。また、N-1個の前景画像のいずれか一つでも有効であるブロック画像は、背景画像として無効なブロック画像とすればよい。

40

【0287】

三つ以上の層に分類する他の実施の形態として、ステップS120のマスク生成処理において、前景画像の数と同じ数のマスク層のデータを生成する代わりに、一つのマスク層のデータが、2値ではなく、N種類の値のいずれかを示し得るように構成してもよい。この場合、ステップS174のマスク符号化処理では、多値画像の可逆符号化に適したFLATEなどの符号化方式を用いればよい。

50

(入力画像の部分領域の層構造)

また、前述の説明では、層データ(たとえば、前景画像、背景画像、マスク層データ)が、入力画像全体を表現する(解像度のみ異なる)場合について説明した。しかしながら、本発明では、たとえば、背景画像が入力画像全体を表現する領域(解像度は低くてよい)を表現し、前景画像とマスク層データは、入力画像の部分領域を表現するように構成してもよい。

【0288】

ただし、部分領域を表現するように構成する場合、ステップS180のフォーマット処理において、前景画像とマスク層データが背景画像上のどの部分領域に対応しているのかを示す情報を関連付けておく必要がある。

10

【0289】

今回開示された実施の形態はすべての点で例示であって制限的なものではないと考えられるべきである。本発明の範囲は上記した説明ではなくて特許請求の範囲によって示され、特許請求の範囲と均等の意味および範囲内でのすべての変更が含まれることが意図される。

【産業上の利用可能性】

【0290】

本発明は、たとえば、複写機や複合機のように画像を電子的に読み込み、データ容量を圧縮して、蓄積または伝送するような装置に適用することができる。

【図面の簡単な説明】

20

【0291】

【図1】第1の実施の形態における画像符号化装置の構成を示すブロック図である。

【図2】第1の実施の形態において、画像符号化装置で実行される画像符号化処理のフローチャートを示す。

【図3】第1の実施の形態において、画像符号化装置で処理される画像を示す図である。

【図4】像域判定処理のフローチャートである。

【図5】マスク生成処理のフローチャートである。

【図6】データテーブルを示す図である。

【図7】各画像の解像度と、各画像の処理単位となるブロック画像の相対的な大小関係を例示した図である。

30

【図8】前景画像生成処理のフローチャートである。

【図9】背景画像生成処理のフローチャートである。

【図10】第2の実施の形態における画像符号化装置の構成を示すブロック図である。

【図11】第2の実施の形態において、画像符号化装置で実行される画像符号化処理Aのフローチャートを示す。

【図12】第2の実施の形態において、画像符号化装置で処理される画像を示す図である。

【図13】前景画像生成処理Aのフローチャートである。

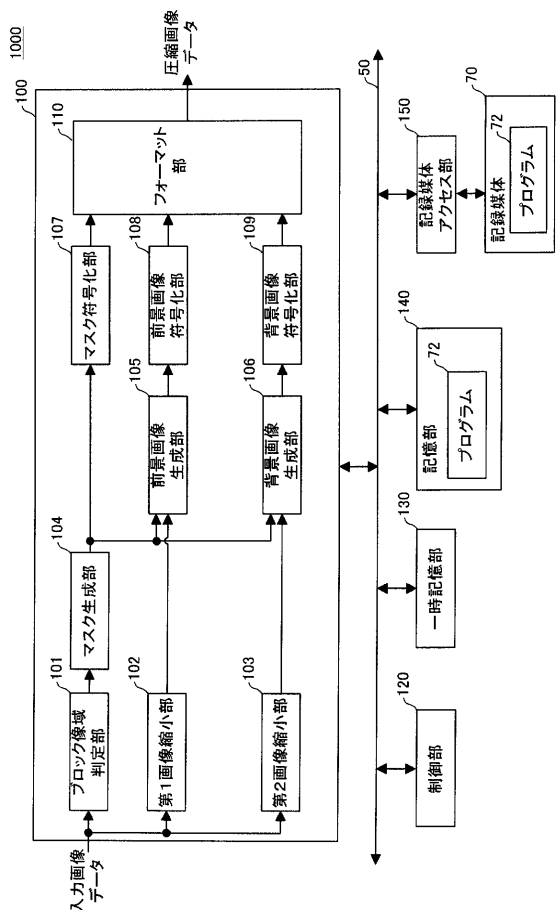
【符号の説明】

【0292】

40

70 記録媒体、100, 100A 画像処理部、101 ブロック像域判定部、102 第1画像縮小部、103 第2画像縮小部、104 マスク生成部、105 前景画像生成部、106 背景画像生成部、107 マスク符号化部、108 前景画像符号化部、109 背景画像符号化部、110 フォーマット部、112 透過色設定部、120 制御部、130 一時記憶部、140 記憶部、150 記録媒体アクセス部、1000, 1000A 画像符号化装置。

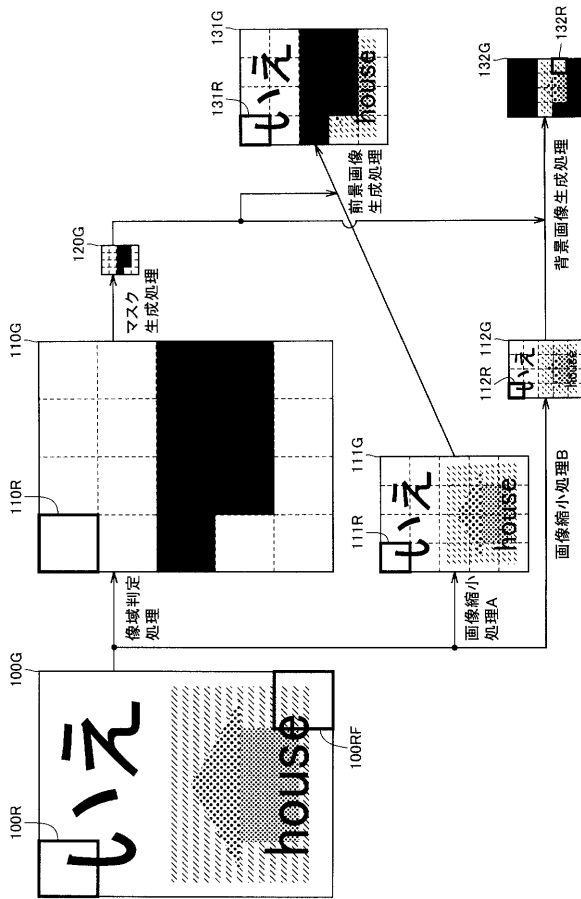
【図1】



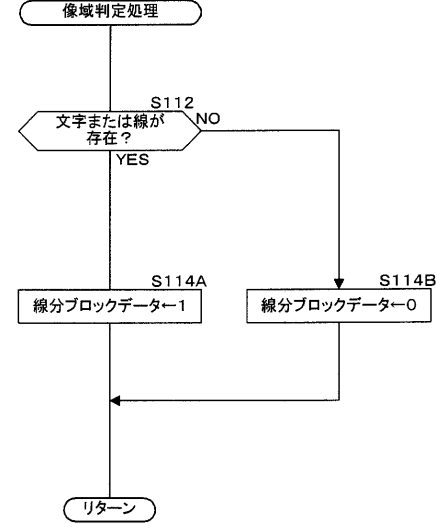
【図2】



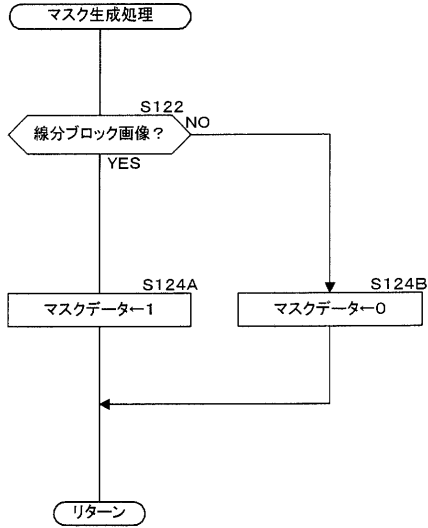
【図3】



【図4】



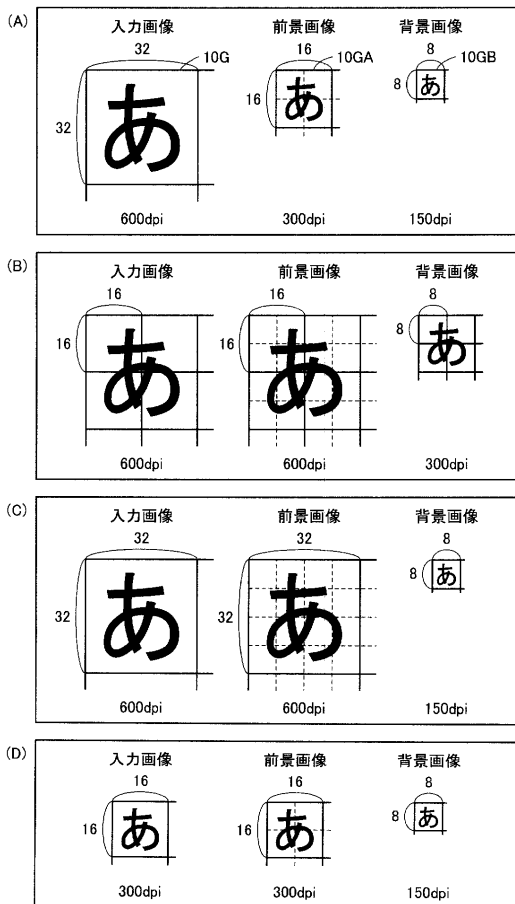
【図5】



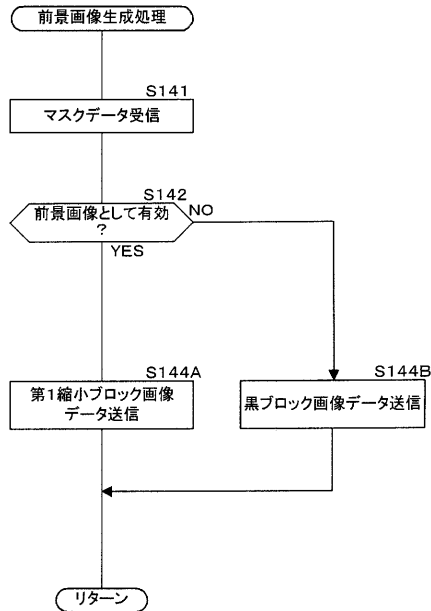
【図6】

組合せ 番号	解像度 (dpi)		ブロックサイズ(精画素数×縦画素数)	
	入力画像	前景画像	領域判定 背景画像	背景画像
1	600	300	32×32	16×16
2	600	600	16×16	16×16
3	600	600	32×32	32×32
4	300	300	16×16	16×16

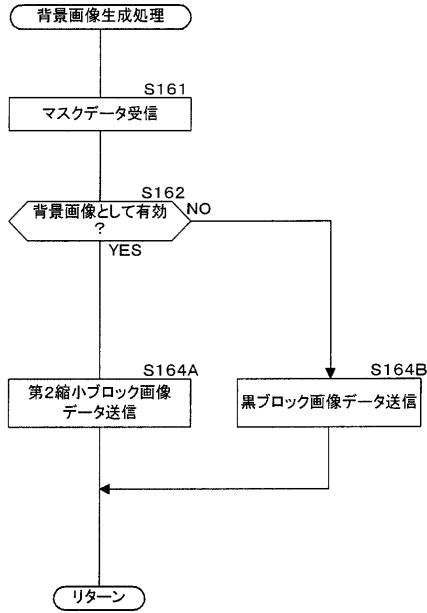
【図7】



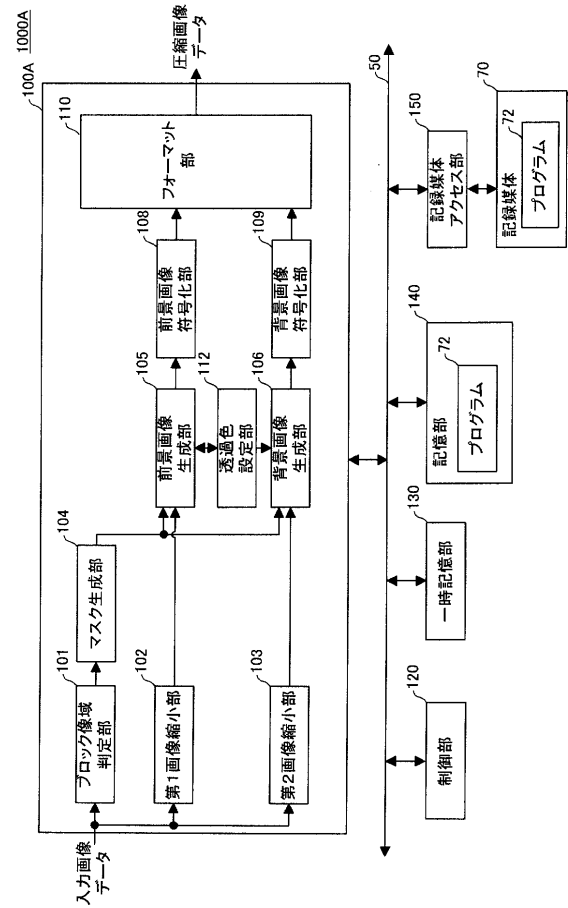
【図8】



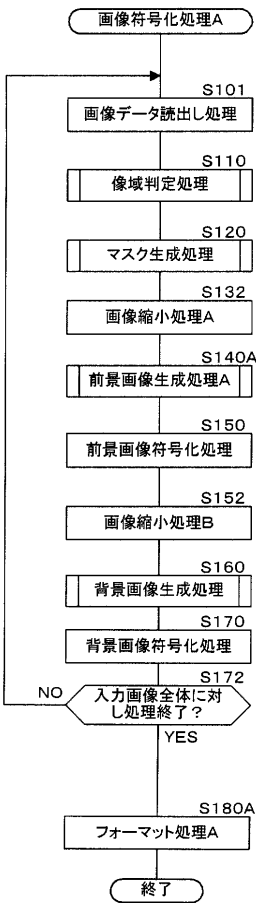
【図9】



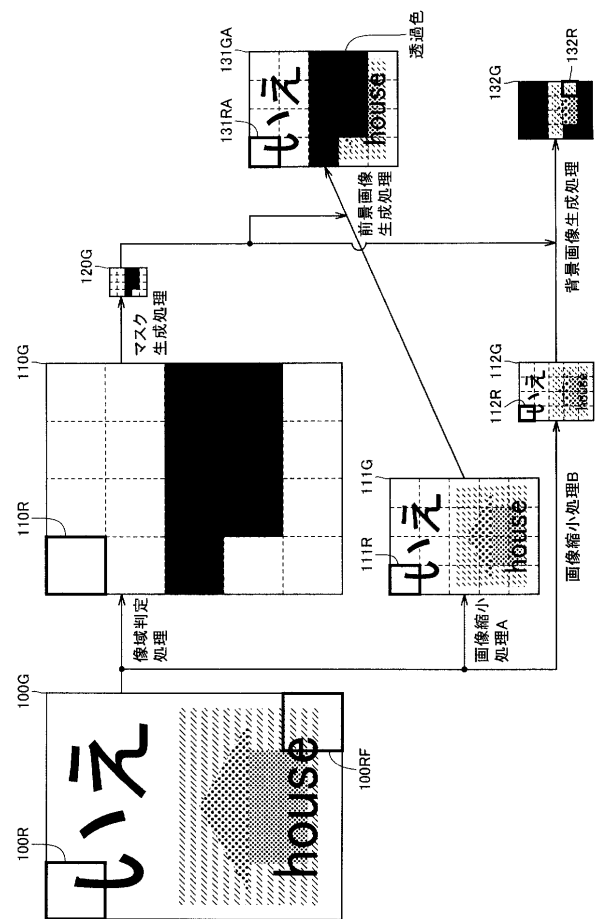
【図10】



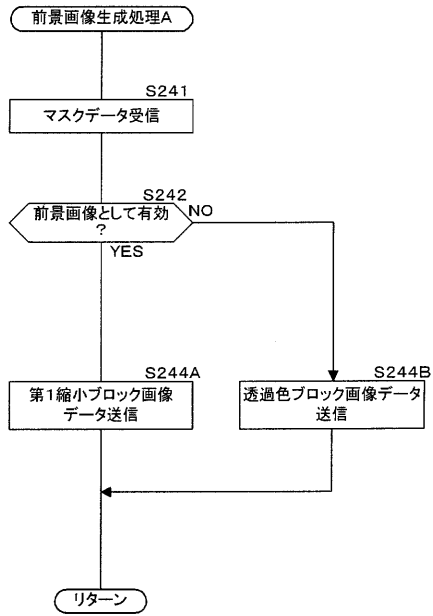
【図11】



【図12】



【図 13】



フロントページの続き

- (72)発明者 八幡 洋一郎
大阪府大阪市阿倍野区長池町2番2号 シャープ株式会社内
- (72)発明者 蔭地 謙作
大阪府大阪市阿倍野区長池町2番2号 シャープ株式会社内
- (72)発明者 齋鹿 尚史
大阪府大阪市阿倍野区長池町2番2号 シャープ株式会社内

審査官 松永 隆志

- (56)参考文献 特開2004-350182(JP,A)
特開2000-196895(JP,A)
特開平07-240845(JP,A)
特開平06-333033(JP,A)
特開平04-356873(JP,A)
特開2005-151382(JP,A)
特開2005-151097(JP,A)

(58)調査した分野(Int.Cl., DB名)

H04N 1/413
H04N 1/41
H04N 7/26