

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4662325号
(P4662325)

(45) 発行日 平成23年3月30日(2011.3.30)

(24) 登録日 平成23年1月14日(2011.1.14)

(51) Int.Cl.		F I			
HO4N	1/387	(2006.01)	HO4N	1/387	101
GO6T	3/40	(2006.01)	HO4N	1/387	
GO6T	3/60	(2006.01)	GO6T	3/40	A
			GO6T	3/60	

請求項の数 7 (全 26 頁)

(21) 出願番号	特願2003-4947 (P2003-4947)	(73) 特許権者	000001007
(22) 出願日	平成15年1月10日(2003.1.10)		キヤノン株式会社
(65) 公開番号	特開2003-309719 (P2003-309719A)		東京都大田区下丸子3丁目30番2号
(43) 公開日	平成15年10月31日(2003.10.31)	(74) 代理人	100076428
審査請求日	平成18年1月10日(2006.1.10)		弁理士 大塚 康德
審判番号	不服2009-5334 (P2009-5334/J1)	(74) 代理人	100112508
審判請求日	平成21年3月11日(2009.3.11)		弁理士 高柳 司郎
(31) 優先権主張番号	特願2002-35434 (P2002-35434)	(74) 代理人	100115071
(32) 優先日	平成14年2月13日(2002.2.13)		弁理士 大塚 康弘
(33) 優先権主張国	日本国(JP)	(74) 代理人	100116894
			弁理士 木村 秀二
		(74) 代理人	100130409
			弁理士 下山 治
		(74) 代理人	100134175
			弁理士 永川 行光

最終頁に続く

(54) 【発明の名称】 画像処理装置及び画像処理方法

(57) 【特許請求の範囲】

【請求項1】

画像データをブロックごとに画像処理する画像処理手段と、
前記画像データの各ブロック単位において画像処理を実行するか否かの実行情報を入力する実行情報入力手段と、

前記ブロックごとに、前記実行情報を利用して前記画像データを画像処理するか否かを判定する判定手段とを備え、

前記画像処理手段が、画像処理を実行すると判定されたブロックの画像データについて画像処理を行うとともに、前記判定手段が、前記画像処理を実行すると判定されたブロックを示す第2の実行情報を生成することを特徴とする画像処理装置。

10

【請求項2】

前記画像データに対して画像処理を実行する範囲を指定する指定手段をさらに備えることを特徴とする請求項1記載の画像処理装置。

【請求項3】

前記画像データを解像度変換する解像度変換手段をさらに備えることを特徴とする請求項1又は2に記載の画像処理装置。

【請求項4】

前記画像データを構成するブロックの位置を並び替える並び替え手段をさらに備えることを特徴とする請求項1から3までのいずれか1項に記載の画像処理装置。

【請求項5】

20

画像データをブロックごとに画像処理する画像処理方法であって、
 前記画像データの各ブロック単位において画像処理を実行するか否かの実行情報を入力させる実行情報入力工程と、
 前記ブロックごとに、前記実行情報を利用して前記画像データを画像処理するか否かを判定する判定工程と、
 画像処理を実行すると判定されたブロックの画像データについて画像処理を行う画像処理工程とを有し、
前記判定工程では、前記画像処理を実行すると判定されたブロックを示す第2の実行情報を生成することを特徴とする画像処理方法。

【請求項6】

画像データをブロックごとに画像処理する画像処理装置を制御するためのコンピュータプログラムであって、
 前記画像データの各ブロック単位において画像処理を実行するか否かの実行情報を入力させる実行情報入力手順と、
 前記ブロックごとに、前記実行情報を利用して前記画像データを画像処理するか否かを判定する判定手順と、
 画像処理を実行すると判定されたブロックの画像データについて画像処理を行う画像処理手順とを有し、
前記判定手順では、前記画像処理を実行すると判定されたブロックを示す第2の実行情報を生成することを特徴とするコンピュータプログラム。

【請求項7】

請求項6に記載のコンピュータプログラムを格納したことを特徴とする記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、ブロック化された画像データに対し、クリッピング処理や解像度変換、回転処理等の画像処理を実施するのに好適な画像処理装置及び画像処理方法に関する。

【0002】

【従来の技術】

従来、ブロック化された画像データに対して、クリッピング処理、解像度変換処理及び回転処理のような画像処理を行って所望のデータを生成する画像処理装置においては、入力されたブロック状の画像データをラスタ化した後、それぞれの画像処理が行われていた。

【0003】

【発明が解決しようとする課題】

しかしながら、従来の画像処理方法では、一旦ラスタ化した後に画像処理が行われるために、全体の画像処理に要する時間が長くなるだけでなく、画像処理の際に必要な処理用のメモリを画像処理装置に多く搭載する必要があるという問題があった。また、画像の回転等の並び替え処理を行う場合、従来の方法では並び替え処理のためのバッファが必要であった。また、並び替え処理のためのバッファを削減するために、書き込みアドレスにおいて並び替えを実行した場合は、連続したアドレスにならないため、バーストモードを使用できず、アクセスに時間がかかるという問題もあった。

【0004】

本発明は、このような事情を考慮してなされたものであり、ブロック化された画像データに対して画像処理を実施するに際し、画像データ全体に対する画像処理速度を高速化するとともに、画像処理に必要なとされるメモリの容量を小さくすることができる画像処理装置及び画像処理方法を提供することを目的とする。

【0005】

【課題を解決するための手段】

上記課題を解決するため、本発明に係る画像処理装置は、画像データをブロックごとに画像処理する画像処理手段と、前記画像データの各ブロック単位において画像処理を実行

10

20

30

40

50

するか否かの実行情報をする実行情報手段と、前記ブロックごとに、前記実行情報を利用して前記画像データを画像処理するか否かを判定する判定手段とを備え、前記画像処理手段が、画像処理を実行すると判定されたブロックの画像データについて画像処理を行うとともに、前記判定手段が、前記画像処理を実行すると判定されたブロックを示す第2の実行情報を生成することを特徴とする。

【0008】

【発明の実施の形態】

以下に本願発明の実施形態を示す。もちろん以下の実施形態は、本願発明の技術分野における当業者による実施を容易にするために開示を提供するものであり、特許請求の範囲によって確定される本願発明の技術的範囲に含まれるほんの一部の実施形態にすぎない。従って、本願明細書に直接的に記載されていない実施形態であっても、技術思想が共通する限り本願発明の技術的範囲に包含されることは当業者にとって自明であろう。

10

【0009】

なお、以下には便宜上複数の実施形態を記載するが、これらは個別に発明として成立するだけでなく、もちろん、複数の実施形態を適宜組み合わせることで発明が成立することは、当業者であれば容易に理解できよう。

【0010】

[第1の実施形態]

以下、図面を参照して、本発明の第1の実施形態による画像処理装置について説明する。

【0011】

図1は、本発明の第1の実施形態による画像処理装置の概要を説明するためのブロック図である。図1において、入力部1から入力される画像データとして、一定サイズ(例えば8×8画素など)にブロック状にスキャンされた画像データが入力される。このようなデータはJPEGのようにブロック符号化した画像データを復号化した際に生成される。

20

【0012】

画像処理部2は入力部1に接続し、ブロック化された画像データに各種画像処理をブロック単位に行った後、ラスタ化した画像データに変換する前段画像処理部21と、ラスタ変換された画像データに対して画素単位の画像処理が行われる後段画像処理部22とから構成される。また、出力部3は画像処理部2に接続され、画像処理部2で処理された画像データを外部に出力する、或いは記憶装置等に記憶するための出力装置である。尚、画像処理部2内における画像処理部は上述した2つに限定されるものではなく、1つであっても3つ以上であってもよい。すなわち、本発明は、画像処理後の画像データに対して、所定ブロックごとに画像処理を実行する第1の画像処理手段(前段画像処理部21)を備えることを特徴とする。また、本発明は、第2の画像処理手段(後段画像処理部22)が、1画素単位で画像データを画像処理することを特徴とする。

30

【0013】

図2は、図1における前段画像処理部21の細部構成を説明するためのブロック図である。図2に示すように、本実施形態における前段画像処理部21は、入力された画像データの範囲から処理範囲を指定するクリッピング処理部211と、画像データの解像度を変換する解像度変換処理部212と、画像データの並び替えを行う並び替え処理部213と、ラスタ化のためのバンドバッファとして使用するSDRAM214とから構成される。

40

【0014】

尚、図2において、最初の処理部であるクリッピング処理部211へ入力される画像データをData_in、また、各処理部から出力される画像データをDataで示す。そして、説明のために出力される画像データである「Data」の前には、その処理を表す名称を付与している。さらに、最初の処理部であるクリッピング処理部211へ入力された画像データに関する有効信号をValid_in、各処理部で処理される画像データに関する有効信号をValidとする。このValidの前にも、その処理を表す名称を付与している。本実施形態による前段画像処理部21は、入力される画像データData_inと、入力される画像データに関する有効信号Valid_inを用いて、入力された画像データが有効である場合(有効信号Valid_in =

50

Hが入力された場合) についてのみ、入力された画像データを取り込んでそれぞれの処理を行うようにしている。

【 0 0 1 5 】

まず、図 1 に示す入力部 1 から前段画像処理部 2 1 のクリッピング処理部 2 1 1 に対して、画像データData_inと有効信号Valid_inが入力される。上述したように本実施形態では、入力される画像データはブロック化された画像であるとする。例えばクリッピング処理部 2 1 1 には、ブロック符号化する際に用いられた 8 × 8 画素のブロックごとの画像データが画像データData_inとして入力され、同時に、入力された画像データが有効データであれば有効信号Valid_in=Hが入力される。なお、無効データの場合はValid_in=Lが入力される。

10

【 0 0 1 6 】

クリッピング処理部 2 1 1 では、入力された画像データがクリッピングされる範囲(クリッピング有効範囲)であるかどうかの判定をブロック単位で行う。その結果、入力された画像データが有効データで、かつ、クリッピング有効範囲内の画像データであると判定された場合は、有効信号(Crip_Valid=H)が出力される。また、当該画像データがクリッピング有効範囲以外あるいは無効データの場合は、有効信号を出力しない(Crip_Valid=L)。あるいは当該画像データがクリッピング有効範囲以外の場合に他の無効信号を出力してもよい。クリッピング処理部 2 1 1 におけるこの動作によって、クリッピング処理部 2 1 1 に接続された解像度変換処理部 2 1 2 では、クリッピング処理部 2 1 1 が有効であると判定したブロックのみについて解像度変換処理を実施することが可能となる。

20

【 0 0 1 7 】

図 3 は、クリッピング処理部 2 1 1 で実施されるクリッピング処理について説明するための概略図である。図 3 (a) に示すように、画像データの横方向を X、縦方向を Y とする。このとき、ブロック単位でクリッピングする範囲を次のようにして指定する。すなわち、クリッピング範囲を矩形領域として、まず領域の左上の頂点をStart_Pointとして、その座標(X S , Y S) が指定される。同様に、矩形領域の右下の頂点をEnd_Pointとして、その座標(X E , Y E) が指定される。この 2 点で図 3 に示すようにして定義することができる矩形領域の範囲で囲まれた部分を本実施形態ではクリッピングし、当該範囲内のブロックについて有効信号Crip_Validを出力する。なお、クリッピング範囲の指定方法は、上記に限らず始点(または終点)とサイズ(幅、高さ)を用いても良い。また、画素単位のクリッピングパラメータからブロック単位のクリッピングパラメータを生成しても良い。

30

【 0 0 1 8 】

尚、本実施形態においてクリッピング処理の際に用いられるブロック形状は、図 3 (a) に示すような正方形だけに限定せず、図 3 (b) に示す例のように横長形状のブロックや、図 3 (c) に示す例のような縦長形状のブロックであってもよい。即ち、任意のブロック形状に適用できる。

【 0 0 1 9 】

上述したように、J P E G 画像における D C T ブロック(8 × 8 画素) 単位のように、ブロック化された画像データについては、第 1 のクリッピング処理にてブロック単位のクリッピングを行うことでブロック単位の出力画素の削減を行い、また、その後の画像処理にて第 2 クリッピング処理を行うことで画素単位のクリッピングが可能である。そして、このクリッピング処理部 2 1 1 にてブロック単位で画像データを有効データ又は無効データの判定を行い、有効データのみを後段の処理部で処理するようにすることによって、その後の処理において無駄な処理を削減でき、画像処理全体の負荷の軽減を図ることが可能となる。

40

【 0 0 2 0 】

特に、前段画像処理部 2 1 の出力をバッファに格納する場合は、バッファ容量の削減も可能である。なお、ブロック単位のクリッピングであるので、ブロック内の画素数などのパラメータは全てのブロックで同一となり、後段の処理も含めてクリッピングのために増加

50

する処理はない。また、上述したようにクリッピング処理は有効信号の操作により行うので、後段処理のインタフェース（I/F）も変更不要である。なお、本実施形態では、以降の処理へクリッピング終了信号（Crip_Finish）を各処理におけるレイテンシ（latency）を補償して伝播することにより、以降の処理を適切なタイミングで終了させるようにしている。この構成により、最終段の処理にて最終画素のタイミングを検出でき、最終処理の終了を割り込み信号等によってCPU等へ通知することで無駄な処理（時間）の削減が可能である。

【0021】

解像度変換処理部212では、有効信号Crip_Validに対応したブロックに対して解像度変換（Resize）処理が行われる。本実施形態では、解像度変換処理部212はブロックサイズの縮小を実行する。例えば、8×8画素のブロックを4×4画素のブロックに変更する。即ち、ブロックサイズの縮小にて出力画素の削減を行うのである。解像度変換処理をブロック内で行うことにより、単純な解像度変換を可能にしている。例えば、入力画像が8×8画素の大きさのブロック状に入力される場合、解像度変換後のブロックサイズは、8×8、7×7、6×6、5×5、4×4、3×3、2×2、1×1の8通りしかなく、係数の生成が容易である。また、横方向・縦方向に同じ縮小回路を用いることができるだけでなく、ラインメモリも不要となる。但し、線形補間の場合は横方向の縮小回路と縦方向の縮小回路との間にブロックの横方向画素分のレジスタが必要となる。

【0022】

尚、7×7画素以下に縮小する場合は、適切なタイミングで有効信号出力を無効（Resize_Valid=L）とすることで解像度変換処理部をパイプライン化できる。即ち、パイプライン化した場合、所定のタイミングで不要信号が発生するが、有効信号出力を無効（Resize_Valid=L）とするだけで間引き処理を容易に実行できる。この操作により、縮小率による制御は係数の切り替えと有効信号出力の切り替えのみで良く、処理が簡略化できる。

【0023】

このように、入力される画像データのブロックサイズにあわせてブロック内で解像度変換処理を行うことにより、簡単に画像データの解像度変換が実現できる。もちろん、さらに細かな解像度変換を後段画像処理部22等で行うことも可能である。そして、解像度変換されたブロックが、有効画像データの場合は同様にして、解像度変換された画像データResize_Data_outと当該画像データに関する有効信号Resize_validを出力することにより、次に行われる並び替え処理の処理対象をさらに削減することが可能となる。なお、拡大方向の解像度変換は、出力画素数が増加するので、ここでは実行しない。

【0024】

並び替え処理部213における並び替え処理では、YCrCb信号の同時化（点順次化）及び回転、ラスタ化の処理が行われる。通常、ブロック符号化においては符号化効率向上のため、輝度データYと色差データCrCbに変換されて符号化される。従って、本実施形態の並び替え処理部213には輝度データYと色差データCrCbがブロック順に入力される。よって、後段でYCrCbデータをRGBデータに変換するにはYCrCbデータの同時化が必要になる。本実施形態では、このYCrCbデータの同時化のためのバッファを用いてYCrCbデータの同時化及びブロック内回転、ブロック内ラスタ化の処理を同時に実行する。

【0025】

まず、Yブロックデータが入力されているときはYブロックバッファ（不図示）に格納し、Crブロックデータが入力されているときはCrブロックバッファ（不図示）に格納し、Cbブロックデータが入力されているときはCbブロックバッファ（不図示）に格納する。次に、Yブロックバッファ、Crブロックバッファ、Cbブロックバッファの画素位置を合せて同時に読み出すことで同時化する。輝度データYと色差データCrCbのサンプリングレートが異なる場合は、サンプリングレートの低い方（通常は、色差データ）のデータを補間して出力する。また、複数ブロックで構成されている場合は、複数ブロックを結合した状態でラスタ化して出力する。

10

20

30

40

50

【 0 0 2 6 】

画像全体のラスタ化はバンドバッファ（SDRAM 2 1 4 上に構成）に対する格納時に行われる。なお、ここでは上記 Y C r C b の点順次データを R G B データに変換した後、上記バンドバッファに格納するものとする。上記ブロック内ラスタ化及び Y C r C b 同時化（点順次化）された画像データは上記バンドバッファの該当位置にブロック状に格納される。その結果、上記バンドバッファにはクリッピング範囲内のラスタ化された画像が格納される。

【 0 0 2 7 】

また、ラスタ変換の実行と同時に、画像データの回転処理を行う場合には、ブロック内の回転を上記ブロックバッファにて実行した後、回転モードに合わせてブロックの先頭アドレスを変更しながらデータをバンドバッファに保存すればよい。予めブロック内の回転を上記ブロックバッファにて実行しておくことで、回転モードによるバンドバッファへの書き込みシーケンスの違いは、ブロックの先頭アドレスの設定のみに集約される。よって、後段画像処理部 2 2 等では、ラスタ変換及び回転処理がされた R G B 点順次画像データを用いて画像処理を実施することができる。

10

【 0 0 2 8 】

図 4 は、並び替え処理部 2 1 3 で行われる回転処理を説明するための概略図である。図 4 において、回転処理の例として画像データの 0 度、9 0 度、1 8 0 度及び 2 7 0 度の 4 つの角度での回転例が示されている。この例で用いられている画像データは、横成分が 4 ブロック、縦成分が 3 ブロックの 4 × 3 ブロックの画像である。この例に示すように、画素単位での並び替え（回転）処理と、ブロック単位での並び替え（回転）処理は同様にして実施することができる。

20

【 0 0 2 9 】

図 4 に示すように、並び替え（回転）処理の対象となる画像 4 0 において、画像内の各ブロックには番号が付与されている。この番号は右側の数字が画像の X 成分、左側の数字が画像の Y 成分としている。すなわち、X = 0、Y = 0 のブロックは、「0 0」で表され、X 方向に 1 ずれた、すなわちブロック 0 0 の右隣のブロックは「0 1」で表現されている。

【 0 0 3 0 】

画像 4 0 に対して回転角 0 度の並び替え処理を実施した場合、画像 4 0 と同じ配置の画像 4 1 が例えば S D R A M 2 1 4 に記憶される。

30

【 0 0 3 1 】

回転角 1 8 0 度の並び替えをした場合、画像 4 3 に示すように、画像 4 0 において左上に位置したブロック 0 0 を右下へ移動しなければならない。このように、ブロック 0 0 からブロック 2 3 までのすべてのブロックを移動する。また、回転角 9 0 度、2 7 0 度の場合も同様の処理をブロック単位で行うことにより、それぞれ画像 4 2、4 4 に示すような結果となる。なお、9 0 度及び 2 7 0 度の場合は縦横のサイズが逆になる。

【 0 0 3 2 】

さらに、並び替え処理部 2 1 3 における処理について説明する。図 5 は、第 1 の実施形態における並び替え処理部 2 1 3 の処理動作の具体例を説明するための図である。図 5 を用いて、画像データのアドレスの連続性を増加させることと、転送効率を向上させる効果が得られる理由について説明する。ここでは並び替え処理部 2 1 3 で、4 × 4 画素のブロックで構成される画像 5 0 を 1 8 0 度に回転する処理を行って画像 5 1 が生成されるものとする。従来の処理では、入力画素順に画像 5 1 の位置に対応するアドレスを生成する処理を行っていたので、画像データが S D R A M 2 1 4 に記録されるような場合、アドレスは減少する方向になるのでバーストモードが使用できず、S D R A M 2 1 4 へのアクセスは 1 画素ごとになり、転送効率が非常に悪くなっていた。

40

【 0 0 3 3 】

しかし、本実施形態では、回転処理を行った後の画像データを、一旦、ブロックバッファ 5 2 へ保存する。これによって、アドレスが増加する方向に連続した複数のデータ（図 5

50

においては4つのデータ)を一度にSDRAM214へ記録させることが可能になる。すなわち、従来1ブロック16回のアクセスが必要であったのを4回のアクセスで済むようになった。つまり、ブロック内の並び替え処理を行うことによって、アドレスの連続性が増加し、これによって転送効率を向上させることが可能になったのである。

【0034】

図6は、クリッピング処理以降の解像度変換処理と並び替え処理を詳細に説明するための図である。ここでは、解像度変換部212に対して1ブロックが8×8画素で構成された画像データが入力され、解像度変換によって解像度が4×4画素のブロックに変換されるものとする。そして、並び替え処理部213において、180度回転の並び替え処理が行われるものとする。

10

【0035】

まず、1ブロック8×8画素(0~63)のデータが、輝度信号Yの0番目のデータから順番に有効信号Crip_Validと同時に、解像度変換部212に入力される。そして、解像度変換部212において解像度変換処理が行われ、1ブロック4×4画素の画像データに変換される。そして、変換された画像データは上述したような並び替え処理が行われ、一旦、Yブロックバッファ60、Crブロックバッファ61、Cbブロックバッファ62にそれぞれ格納される。1ブロック分のYCrCbデータが格納されると、Yブロックバッファ60、Crブロックバッファ61、Cbブロックバッファ62のそれぞれの先頭より順番に読み出され、RGBデータに変換されてバンドバッファの該当アドレスに格納されて並び替えが完了する。

20

【0036】

なお、上述したような画像処理部2における各処理の順番は任意である。さらに、何れかの処理が欠落していてもよい。さらにまた、各処理において入力信号をそのまま出力するモード(スルーモード)を設け、各処理の実行を任意に選択することができるようにもよい。

【0037】

また、上述した実施形態では、ブロック符号化のようにブロック化された画像を例としてSDRAM等の記憶装置を用いた装置による説明を行ったが、これは、ソフトウェア処理で行ってもよい。

【0038】

上述したように、本発明は、画像データに対して所定ブロックごとに画像処理を実行する画像処理手段を備える画像処理装置であって、ブロックごとに画像データを画像処理するか否かを判定する判定手段を備え、画像処理手段が、画像処理を実行すると判定されたブロックの画像データについて画像処理を行うことを特徴とする。

30

【0039】

また、本発明は、画像データに対して画像処理を実行する範囲を指定する指定手段(クリッピング処理部211)を備えることを特徴とする。さらに、本発明は、画像データを所定解像度に変換する解像度変換手段(解像度変換処理部212)を備えることを特徴とする。さらにまた、本発明は、画像データを構成するブロックの位置を所定位置に並び替える並び替え手段(並び替え処理部213)を備えることを特徴とする。

40

【0040】

さらに、本発明は、入力画像データがラスタ化される前のJPEG復号化データであって、画像処理に用いられるブロックが、JPEG復号化が行われた際に用いられたブロック(8×8画素またはMCU)であることを特徴とする。さらにまた、本発明は、画像データがラスタ化される前のMPEG復号化データであって、画像処理に用いられるブロックが、MPEG復号化が行われた際に用いられたブロック(8×8画素またはMB)であることを特徴とする。

【0041】

次に、上述した構成による画像処理装置の処理動作について説明する。図8は、第1の実施形態による前段画像処理部21の動作手順を説明するためのフローチャートである。

50

【 0 0 4 2 】

まず、画像処理装置の入力部 1 から処理対象となる画像データを入力するとともに、不図示の操作部等を用いて上述した各画像処理に必要なパラメータ設定が行われる（ステップ S 8 1）。ここで、画像処理におけるパラメータとは、画像をクリッピング処理する範囲の指定、解像度変換処理のサイズ、並び替え処理に必要な回転モード、サンプリングモード（J P E G での 4 : 4 : 4、4 : 2 : 2、4 : 2 : 0 等）、出力バンドバッファの領域等のことである。

【 0 0 4 3 】

次に、入力された画像データに対して、クリッピング処理部 2 1 1 においてクリッピング処理が行われる（ステップ S 8 2）。図 9 は、ステップ S 8 2 でのクリッピング処理手順を詳細に説明するためのフローチャートである。

10

【 0 0 4 4 】

ここでは、図 3 に示したように、クリッピングの範囲始点を（X S , Y S ）とし、終点を（X E , Y E ）として指定した場合を例に挙げる。上述したように、本発明においては、あるブロック単位でのクリッピングが行われる。尚、より細かいクリッピングを後段処理部で行うことが可能である。このブロック単位とは、例えば J P E G データであれば、D C T ブロックである 8 × 8 画素単位、あるいは Y , C r , C b をひとつにまとめた M C U を単位としてもよい。このようなブロック単位でのクリッピング処理を行うことにより、バンドバッファに格納されるデータ量を削減する。これにより後段画像処理部 2 2 での処理に要する負荷を軽減することができるだけでなく、画像処理装置全体の処理速度も速くなる。

20

【 0 0 4 5 】

ステップ S 9 1 では、クリッピング処理が終了したか否かの判定が行われる。即ち、クリッピング範囲の指定を始点（X S , Y S ）と終点（X E , Y E ）で示される矩形領域とした場合、入力画像ブロックの座標 X , Y が X E , Y E の範囲を共に越えている場合（Y E S）、すでにクリッピングしたい範囲を越えていることを示すので、ここで終了処理を行う（ステップ S 9 3）。この終了処理により、後段画像処理部 2 2 に対して終了したことを示す指示（これにより、後段画像処理部 2 2 が起動する）や、終了信号（Crip_Finish）の伝播により、前段画像処理部 2 1 におけるその後の処理を終了させることも可能である。

30

【 0 0 4 6 】

一方、ステップ S 9 1 においてクリッピング処理が終了していない場合（N O 側）、指定した範囲がクリッピング範囲内かの判定が行われる（ステップ S 9 2）。その結果、クリッピング範囲内の場合（Y E S 側）、有効データ処理を行って後続する画像処理部（ここでは解像度変換処理ステップ S 8 3）に対して有効なデータであることを示す（ステップ S 9 4）。一方、ステップ S 9 2 において有効範囲でないと判定された場合（N O 側）、無効データであることを示す（ステップ S 9 5）。なお、後続する画像処理部に信号等を出力しないことによって、無効データ処理とすることも可能である。入力ブロックに対する有効データ処理ステップ S 9 4 及び無効データ処理ステップ S 9 5 が終了すると、次のブロックの判定のため、終了判定ステップ S 9 1 に戻り、上記操作を繰り返す。

40

【 0 0 4 7 】

ステップ S 8 2 においてクリッピング処理にて有効と判断されたブロックには、解像度変換処理が行われる（ステップ S 8 3）。図 1 0 は、ステップ S 8 3 で行われる解像度変換処理の詳細を説明するためのフローチャートである。解像度変換処理部 2 1 2 では、まず解像度変換があるか否かの判定が行われる（ステップ S 1 0 1）。その結果、解像度変換がないと判定された場合（N O 側）、解像度変換処理を行うことなく終了し、後続する画像処理部に入力された画像データをそのまま渡す。一方、解像度変換があると判定された場合（Y E S 側）、解像度変換のパラメータ設定が行われる（ステップ S 1 0 2）。そして、設定されたパラメータに従い解像度変換処理が行われ、所望のサイズになるように間引き処理や線形補間処理等の処理が行われる（ステップ S 1 0 3）。なお、図 8 のパラメ

50

ータ設定ステップS 8 1にて設定された解像度変換処理後のサイズより、解像度変換処理判定ステップS 1 0 1を行い、パラメータ設定ステップS 1 0 2を省略することも可能である。例えば、入力画像のブロックの横サイズが8で、解像度変換後の横サイズが8未満の場合、解像度変換を行う。この時、解像度変換後の横サイズが解像度変換のパラメータになっており、たとえば4に設定されている場合、解像度変換によって8×8画素のブロックが4×4のブロックに縮小される。

【0048】

ステップS 8 3で解像度変換処理が行われた後、並び替え処理部2 1 3において、ブロック内の並び替え(回転等)処理が行われる(ステップS 8 4)。そして、処理された画像データがSDRAM 2 1 4に保存される(ステップS 8 5)。ステップS 8 4において、並び替え処理が実行されることによって保存時にアドレス生成が簡単になるだけでなく、従来の1画素ごとの保存(シングルライト)ではなく、複数個単位の保存(バーストライト)が可能となり、1画素あたりのアクセス速度を早くすることができる。

10

【0049】

すなわち、保存先がSDRAMとすると、SDRAMへの書き込みアドレスの連続性を増加させて、転送効率を上げることが可能となる。さらに、保存先が組み込み機器のメインメモリのように、多くのバスマスタからのアクセスが生じるような場合においては、アクセス効率をあげることによって後段画像処理部2 2の画像処理だけでなく、他の処理への影響も少なくすることができる。

【0050】

さらにまた、JPEG画像を例にあげて説明すると、クリッピング処理において本実施形態のようにブロック単位で大まかな切り抜きを行い、その後、切り抜いたブロックに対して解像度変換を行い、ラスタ化されたRGB点順次データとして保存することにより、後段処理部における画素単位のクリッピングや、より詳細な解像度変換等の処理をより簡単に行うことができる。

20

【0051】

すなわち、本発明に係る画像処理装置は、画像データを画像処理するための条件を設定する条件設定手段と、画像データに対して画像処理を実行する範囲を指定する指定手段と、画像データを所定解像度に変換する解像度変換手段と、画像データを構成するブロック内での順番を並び替える並び替え手段と、並び替えたブロックを記録デバイスの所定位置に記録する記録手段とを備えることを特徴とする。

30

【0052】

上述したように、ブロック単位でクリッピング処理、解像度変換処理、並び替え処理を行うことにより、処理動作そのものが簡単になるだけでなく、後段処理部への負荷を軽減できるといえる。また、各処理の途中で画像データを保存する場合、その保存に必要な容量も少なくすることが可能となる。

【0053】

また、ブロック単位ですでに並び替えを行っているので保存先等へのアドレス生成が簡単になり、また並び替えを行っているので、連続データとして一度に複数のデータを格納することが可能となり、転送効率が良くなる。

40

【0054】

[第2の実施形態]

図1 1は、本発明の第2の実施形態に係る画像処理装置の構成を示すブロック図である。図中、1 1 0はJPEG復号化回路、1 1 1はMCUクリッピング回路、1 1 2はMCU縮小回路、1 1 3はブロック内回転回路、1 1 4はブロックバッファ、1 1 5は点順次化回路、1 1 6はRGB変換回路、1 1 7はページバッファ、1 1 8は画素クリッピング回路、1 1 9はリサイズ回路である。以下、第1の実施形態と異なる部分のみ説明する。

【0055】

入力部1よりJPEG圧縮画像がJPEG復号化回路1 1 0に入力され、復号化されてMCU単位に出力される。MCUクリッピング回路1 1 1はMCU単位に有効・無効を判定

50

し、有効MCUはMCU縮小回路112で所望のサイズに縮小され、ブロック内回転回路113のアドレス生成によりブロックバッファ114の回転後の所定位置に格納される。MCU1つ分のデータがブロックバッファに格納されると、点順次化回路115はブロックバッファ114よりY、Cr、Cbのデータを同時に読み出しRGB回路116にてRGBデータに変換する。なお、Cr、Cbデータがサブサンプリングされている場合は、点順次化時に補間して出力する。変換されたRGBデータはページバッファ117にMCU単位で回転モードに対応したアドレスに格納される。

【0056】

1ページ分のデータがページバッファ117に格納されると、不図示のプリントエンジンの同期信号に従ってページバッファ117のページ先頭よりRGBデータが読み出される。画素クリッピング回路118は上記MCUクリッピング回路111でクリッピングされなかった画素をクリップしリサイズ回路119にて所望のサイズに変換した後、出力部よりプリントエンジンに出力される。

10

【0057】

次に、具体的な動作について説明する。

【0058】

図12は、第2の実施形態に係る画像処理装置による画像処理の具体例を説明するための図である。図2に示すように、本実施形態では2272×1704画素、サンプリングモードを4:2:2のJPEG画像から1600×1200画素をトリミングして1ページが680×480画素のプリンタにて出力するものとする。なお、トリミング範囲は左上の頂点を(0,0)としたとき、始点(300,300)と終点(1899,1499)で囲まれる矩形範囲(境界含む)とする。

20

【0059】

サンプリングモードは4:2:2であるので、MCUサイズは16×8となる。従って横方向のMCU数は2272/16=142、縦方向のMCU数は1704/8=213(いずれも8ビットで良い)となり、MCUクリッピング回路111の横方向MCUカウンタは142進にセットされる。また、トリミングの始点座標は(300,300)、終点座標は(1899,1499)であるから、MCU有効範囲は(XS,YS)=(18,37)、終点座標は(XE,YE)=(119,187)となる。なお、トリミング領域が必ずしもMCU境界と一致するとは限らないので、始点(XS,YS)は切り捨て、終点(XE,YE)は切り上げとなる。具体的には、XSは右4ビットシフトした値、YSは右3ビットシフトした値、XEは右4ビットシフトした値にシフト前の下位4ビットをORした結果(0または1)を加算した値、YEは右3ビットシフトした値にシフト前の下位3ビットをORした結果を加算した値となる。

30

【0060】

MCUクリッピング回路111は上記(XS,YS)=(18,37)及び(XE,YE)=(119,187)をMCUカウンタの値と比較し、有効範囲のMCUのときのみ、前述した有効信号Crip_Validを"H"にする。

【0061】

MCU縮小回路112は有効範囲内のMCUサイズを所定サイズに縮小する。図13は、MCU縮小回路112の細部構成を示すブロック図である。同図に示したように、MCU縮小回路112は、横サイズ縮小回路131と縦サイズ縮小回路132の2つの縮小回路、有効信号生成回路133にて構成される。

40

【0062】

有効信号生成回路133は入力される上記有効信号Crip_Validに従い、内部の横カウンタ、縦カウンタをインクリメントする。この2つのカウンタにより、ブロック内での画素位置を把握する。本実施形態ではブロックサイズは8×8画素なので、横カウンタ、縦カウンタは共に3ビットで、横カウンタのキャリーで縦カウンタがインクリメントされる。横カウンタの値xは横サイズ縮小回路131に、縦カウンタの値yは縦サイズ縮小回路132に供給され、各々補間係数の選択に使用される。また、横カウンタの値x及び縦カウン

50

タの値 y は各々有効信号テーブルに入力され、有効信号に変換される。有効信号テーブルは縮小サイズ別に 8 ビットの出力を持ち、各カウンタの値より 1 ビットを選択する。なお、ここでは説明の簡略化ため、テーブル構成としているが、後述するようにカウンタ値より有効信号を直接生成しても良い。生成された縦、横の有効信号は、入力された有効信号 Crip_Valid にタイミングを合せて AND し、MCU 縮小回路有効信号 Resize_Valid として出力する。

【 0 0 6 3 】

図 1 4 は、横サイズ縮小回路 1 3 1 又は縦サイズ縮小回路 1 3 2 の具体的な構成を示すブロック図である。横サイズ縮小回路と縦サイズ縮小回路との違いは、バッファ 1 4 4 の遅延量で、横サイズ縮小回路の場合は 1 画素分で、縦サイズ縮小回路の場合は 1 ライン (8 画素) 分である。ここでは隣接する画素と補間画素との位置より線形補間で求める。例えば、縮小後のサンプリングポイントが画素 A と画素 B の間で、サンプリングポイントから画素 B までの距離とサンプリングポイントから画素 A までの距離の比が $(1 - P)$ のとき、補間ポイントの値 P は、

$$P = (1 - P) A + P B \quad (1)$$

となる。

【 0 0 6 4 】

図 1 5 は、MCU 縮小後のサンプリングポイントを具体的に示す図である。網点がかかっている丸が縮小後のサンプリングポイントを表している。図 1 5 において、(a) は 7×7 画素、(b) は 6×6 画素、(c) は 5×5 画素、(d) は 4×4 画素、(e) は 3×3 画素、(f) は 2×2 画素に縮小した場合を示している。なお、 1×1 画素に縮小した場合は縮小前の $(x , y) = (3 , 3)$ の位置に一致する。図から明らかなように、縦方向、横方向共に 8 画素周期の繰返しになっている。つまり、ブロック内で縮小は完結している。なお、折り返しノイズ除去のため、JPEG 復号化回路 1 1 0 では Q テーブルを操作し、折り返しノイズとなる高周波成分を排除してから逆 DCT するものとし、 1×1 画素、 2×2 画素、 4×4 画素では補間せずに該当する位置のデータがそのまま出てくるものとする。また、図中の円で囲まれた数字は、有効となるタイミングを示している。

【 0 0 6 5 】

図 7 は、図 1 5 に示したそれぞれのサンプリングポイントの場合における補間係数 (2 5 6 倍して整数化した値) を示す図である。ここで $(2 5 6)$ は縮小後のサンプリングポイントと入力画素の位置が一致している場合を表し、この場合は補間せずに入力画素データをそのまま出力する。また、上述したように、横サイズ縮小回路と縦サイズ縮小回路の補間回路は同じであるので、補間係数も同一となる。従って、縦サイズ縮小回路 1 3 2 においては、図 7 における x は y に置き換わる。

【 0 0 6 6 】

なお、図 7 で背景がドットになっている欄は無効データを示しており、このタイミングのときの有効信号は " L " (無効) となる。従って、補間係数はどのような値を設定してもかまわない。ここでは、回路の簡略化のため、無効データのときの補間係数も規定している。例えば、全ての係数は x 4 で出現順が逆転 ($(1 - P)$ も入れ替え) するようになっている。また、縮小後のサイズ W が 6 の時と 3 の時の係数を同一にしている。例えば $W = 7$ の時の係数テーブルは (2 5 5 , 2 1 9 , 1 8 3 , 1 4 6 , 1 1 0 , 7 3 , 3 7 , 0) に設定し、 $(1 - P)$ は左から ($1 - P$) は右から読み出せばよい。なお、係数テーブルのビット数を削減するため、係数が 2 5 5 の時は補間せずに入力データをそのまま出力する (スルーモード) 。

【 0 0 6 7 】

また、縮小後のサイズ W が 2 のべき乗の時は、全てスルーモードとしているので、係数は不要である。さらに、 $(1 - P)$ は $(1 - P)$ の 2 の補数となるので、 $(1 - P)$ のビット反転 + 1 より生成しても良い。この場合、例えば上記 $W = 7$ の時の係数テーブルは (0 , 3 7 , 7 3 , 1 1 0) の 4 つの係数のみで良い。 $(1 - P)$ は $(1 - P)$ のビット反転 + 1 より生成し、 x 4 では逆順に読み出した値を $(1 - P)$ に、読み出した値のビット反転 + 1 を $(1 - P)$ にセットする

10

20

30

40

50

。

【 0 0 6 8 】

なお、 $x = 7$ （または $y = 7$ ）の時、無効データとしているのは、バッファ144による遅延を補償するためである。通常、横サイズ縮小回路では1画素分、縦サイズ縮小回路では8画素分のレイテンシ（遅延）が発生する。このため、MCU縮小回路112の補間出力は9画素分のレイテンシ（遅延）で出力され、入力有効信号及び横サイズ縮小の有効信号は、縦サイズ縮小の有効信号とタイミングを合せてからANDをしなければならない。しかし、上述のように $x = 7$ （または $y = 7$ ）の時、無効データとすることで、内部の動作タイミングを入力有効信号に同期化することができ、上記タイミング調整は不要になる。但し、上記横サイズ縮小回路、縦サイズ縮小回路の出力をラッチする場合は、2画素分のレイテンシ補償が必要である。さらに、 $x = 0$ （または $y = 0$ ）のデータも可能な限り使用しないようにしている。これは、DCTの高周波成分削減による歪の影響は周辺部に大きく現れるので、周辺部の画素の使用を避けて上記歪の影響を少なくするためである。

10

【 0 0 6 9 】

上記縮小回路の有効信号は簡単に生成できる。即ち、

$W = 1$ の時は $x = 3$ で"H"、

$W = 2$ の時は x の下位2ビット = 01で"H"、

$W = 3$ の時は $x = 1, 4, 6$ で"H"、

$W = 4$ の時は x のLSB = 0で"H"、

$W = 5$ の時は $x = (2, 5, 7)$ 以外で"H"、

$W = 6$ の時は $x = (3, 7)$ 以外で"H"、

$W = 7$ の時は $x = (7)$ 以外で"H"、

となる。

20

【 0 0 7 0 】

$x = 7$ で常に有効信号を"L"とすれば、さらに簡略化できる。例えば、 $W = 7$ では判定は不要になり、 $W = 6$ は $W = 1$ の反転で良い。また、 $W = 5$ は $W = 3$ に+1したものの反転であり、2と5（ $W = 3$ においては1と6）は1の補数の関係にあるので、 x のMSBと下位の2ビットそれぞれをEXORすることによって判定すればよい。

【 0 0 7 1 】

上述したように、ブロックバッファ114ではブロック内の回転及び点順次化が実行される。JPEGの場合、サンプリングモードが4:2:0のときMCUの画素サイズが最大となる。このとき、JPEG復号化回路110からはMCUとして輝度信号が4ブロック（ Y_0, Y_1, Y_2, Y_3 ）、色差信号が各1ブロック（ C_r, C_b ）の6ブロックが8×8画素のブロック毎に順次入力される。ブロックバッファ114には3つのブロックバッファ（Yブロックバッファ、 C_r ブロックバッファ、 C_b ブロックバッファ）があり、各々個別のRAMより構成されている（但し、 C_r ブロックバッファと C_b ブロックバッファのアドレスは共通化できるので、データバスの上位・下位に分離して格納することで同一RAM上に構成しても良い）。

30

【 0 0 7 2 】

ブロック内回転回路113には入力される上記有効信号Resize_Validに従い、内部の書き込み用横カウンタ、縦カウンタ、ブロックカウンタをインクリメントする。上記縦・横2つのカウンタにより、ブロック内での画素位置を把握し、ブロックカウンタにてブロックの位置を把握する。本実施形態ではブロックサイズは最大8×8画素なので、横カウンタ、縦カウンタは共に3ビットとなる。また、MCU縮小回路112でサイズが変更されている場合は、横カウンタ、縦カウンタはサイズ分のカウンタとなる。即ち、縮小後のサイズを W とすると、カウンタが $W - 1$ の時にキャリーが発生し、0に戻る（ W 進カウンタ）。なお、縦カウンタは、横カウンタのキャリーでインクリメントされる。また、MCU内のYブロックの数は最大4なので、ブロックカウンタは2ビットとなり、横カウンタのキャリーと縦カウンタのキャリーのANDでインクリメントされる。

40

【 0 0 7 3 】

50

書き込み用横カウンタの値を X_w 、縦カウンタの値を Y_w 、ブロックカウンタの値を B_w とすると、Yブロックバッファへの書き込みアドレスは (B_w, Y_w, X_w) となる（ここで、"はビット結合を示す）。同様に C_r ブロックバッファと C_b ブロックバッファの書き込みアドレスは (Y_w, X_w) となる。

【0074】

図16は、図12の具体例における輝度信号データがYブロックバッファに格納される様子を示した図である。なお、四角の中の番号は格納されるタイミングを示している。本実施形態では、MCU縮小した場合においても、並び替えせずに縮小前の 8×8 画素単位に格納している。これにより、格納時のアドレスが回転モードに依存しなくなり、アドレス生成が容易になる（アドレス生成部での制御は、Y、 C_r 、 C_b のブロックシーケンス及び縮小後のサイズWによるキャリア制御のみとなる）。

10

【0075】

図17は、図12の具体例における色差信号データが C_r ブロックバッファ及び C_b ブロックバッファ（ C_r ブロックバッファ及び C_b ブロックバッファは同一のメモリ上に構成しても良い。）に格納される様子を示した図である。なお、四角の中の番号は格納されるタイミングを示している。図中に示したように本実施形態では、MCU縮小にて $1/2$ 以下（ $W/4$ ）に縮小した場合、サブサンプリングされている方向の縮小後の画素サイズを2倍にする。即ち、輝度信号の縮小後の画素サイズをWとした時、サブサンプリングされている方向の画素データの縮小後のサイズは $2W$ となる。この構成により、縮小による色差信号の解像度の劣化を最小にする。

20

【0076】

一方、読み出し時は、読み出し用の横カウンタ、縦カウンタ及びブロックカウンタによってアドレスを生成する。上記カウンタの構成はサンプリングモード及び回転モードにより異なる。ここでは、各サンプリングモードに対応するため、Yブロックバッファ用は横カウンタを3ビット、縦カウンタを4ビット、ブロックカウンタを1ビットにて構成し、C（ C_r 、 C_b 共通）ブロックバッファ用は横カウンタを4ビット、縦カウンタを3ビットにて構成する。

【0077】

例えば、Yブロックバッファ用カウンタの場合、 $4:4:4$ モードでは横カウンタ、縦カウンタ共にW進アップダウンカウンタ（3ビット）として動作し、 $4:2:2$ モードでは、横カウンタはW進アップダウンカウンタ、縦カウンタは $2W$ 進のアップダウンカウンタ（ 0° 、 180° ）またはW進のアップダウンカウンタ+1ビットのブロックカウンタ（ 90° 、 270° ）として動作し、 $4:2:0$ モードでは、横カウンタはW進アップダウンカウンタ、縦カウンタは $2W$ 進のアップダウンカウンタ+1ビットのブロックカウンタとして動作する。上記カウンタ構成の切り替えはキャリア（ポロー）の生成方法によって行う。なお、ダウンカウント時の初期値は $W-1$ （W進）または $2W-1$ （ $2W$ 進）である。

30

【0078】

図18は、回転モードとYブロックバッファの読み出し位置との関係を示す図である。図中の四角は 8×8 画素のブロックを、四角中の数字はブロックの入力順を、小さな四角は縮小後のブロック位置を示している。また、矢印はデータの読み出される方向を示している。また、ドットで示されているブロックは $4:2:2$ モードの時の対象ブロックを示している。

40

【0079】

Yブロックバッファの読み出し用横カウンタの値を X_r 、縦カウンタの値を Y_r （または Y_r' ）、ブロックカウンタの値を B_r とし、また、C（ C_r 、 C_b ）ブロックバッファの読み出し用横カウンタの値を X_{cr} （または X_{cr}' ）、縦カウンタの値を Y_{cr} （または Y_{cr}' ）、とすると、図18より、各ブロックバッファの読み出しアドレスは下記の式で表される。尚、以下では、"、"はビット結合を、"!はダウンカウントを示すものとする。

50

(1) サンプリングモード = 4 : 4 : 4 の時 (Y、C r、C b 共通)

0° のとき : Y r , X r

90° のとき : ! X r , Y r

180° のとき : ! Y r , ! X r

270° のとき : X r , ! Y r

(2) サンプリングモード = 4 : 2 : 2 の時

[Y ブロックアドレス]、(Y r ' は 2 W 進)

0° のとき : Y r '[0] , Y r '[3..1] , X r

90° のとき : B r , ! X r , Y r

180° のとき : ! Y r '[0] , ! Y r '[3..1] , ! X r

270° のとき : ! B r , X r , ! Y r

[C r , C b ブロックアドレス] (W > 4 の時、X c r '、Y c r ' は 2 W 進)

0° のとき : Y c r , X c r '[3..1]

90° のとき : ! X c r , Y c r '[3..1]

180° のとき : ! Y c r , ! X c r '[3..1]

270° のとき : X c r , ! Y c r '[3..1]

[C r , C b ブロックアドレス] (W = 4 の時、X c r '、Y c r ' は 2 W 進)

0° のとき : Y c r , X c r '

90° のとき : ! X c r , Y c r '

180° のとき : ! Y c r , ! X c r '

270° のとき : X c r , ! Y c r '

(3) サンプリングモード = 4 : 2 : 0 の時

[Y ブロックアドレス] (Y r ' は 2 W 進)

0° のとき : B r , Y r '[0] , Y r '[3..1] , X r

90° のとき : ! Y r '[0] , B r , ! X r , Y r '[3..1]

180° のとき : ! B r , ! Y r '[0] , ! Y r '[3..1] , ! X r

270° のとき : Y r '[0] , ! B r , X r , ! Y r '[3..1]

[C r , C b ブロックアドレス] (W > 4 の時、X c r '、Y c r ' は 2 W 進)

0° のとき : Y c r '[3..1] , X c r '[3..1]

90° のとき : ! X c r '[3..1] , Y c r '[3..1]

180° のとき : ! Y c r '[3..1] , ! X c r '[3..1]

270° のとき : X c r '[3..1] , ! Y c r '[3..1]

[C r , C b ブロックアドレス] (W = 4 の時、X c r '、Y c r ' は 2 W 進)

0° のとき : Y c r '[3..1] , X c r '

90° のとき : ! X c r '[3..1] , Y c r '

180° のとき : ! Y c r '[3..1] , ! X c r '

270° のとき : X c r '[3..1] , ! Y c r '

なお、上記カウンタの内、所定カウンタの機能 (アップカウントとダウンカウント) を逆転させることで、縦及び横の任意の鏡像を得ることができる。

【 0 0 8 0 】

また、有効信号を操作することにより、各ブロックバッファのカウンタを共通にすることも可能である。この場合、読み出し用の横カウンタ、縦カウンタ及びブロックカウンタは各々 3 ビット、4 ビット、1 ビットのバイナリカウンタでよく、Y ブロックバッファの読み出しアドレスは、

0° のとき : B r , Y r [0] , Y r [3..1] , X r

90° のとき : ! Y r [0] , B r , ! X r , Y r [3..1]

180° のとき : ! B r , ! Y r [0] , ! Y r [3..1] , ! X r

270° のとき : Y r [0] , ! B r , X r , ! Y r [3..1]

のようになる。但し、" , " はビット結合を、"! " はビット反転を示すものとする (以下同じ)。

10

20

30

40

50

【 0 0 8 1 】

また、C (C r , C b) ブロックバッファの読み出しアドレスは、

4 : 2 : 2 モードで $W > 4$ の時、

0 ° のとき : Y r [3 . . 1] , Y r [0] , X r [2 . . 1]

9 0 ° のとき : ! Y r [0] , ! X r [2 . . 1] , Y r [3 . . 1]

1 8 0 ° のとき : ! Y r [3 . . 1] , ! Y r [0] , ! X r [2 . . 1]

2 7 0 ° のとき : Y r [0] , X r [2 . . 1] , ! Y r [3 . . 1]

4 : 2 : 0 モードで $W > 4$ の時、

0 ° のとき : B r , Y r [3 . . 2] , Y r [0] , X r [2 . . 1]

9 0 ° のとき : ! Y r [0] , ! X r [2 . . 1] , B r , Y r [3 . . 2]

1 8 0 ° のとき : ! B r , ! Y r [3 . . 2] , ! Y r [0] , ! X r [2 . . 1]

2 7 0 ° のとき : Y r [0] , X r [2 . . 1] , ! B r , ! Y r [3 . . 2]

4 : 2 : 2 モードで $W = 4$ の時、

0 ° のとき : Y r [3 . . 1] , Y r [0] , X r [1 . . 0]

9 0 ° のとき : ! X r , B r , Y r [2 . . 1]

1 8 0 ° のとき : ! Y r [3 . . 1] , ! Y r [0] , ! X r [1 . . 0]

2 7 0 ° のとき : X r , ! B r , ! Y r [2 . . 1]

4 : 2 : 0 モードで $W = 4$ の時、

0 ° のとき : B r , Y r [2 . . 1] , Y r [0] , X r [1 . . 0]

9 0 ° のとき : ! Y r [0] , ! X r [1 . . 0] , B r , Y r [2 . . 1]

1 8 0 ° のとき : ! B r , ! Y r [2 . . 1] , ! Y r [0] , ! X r [1 . . 0]

2 7 0 ° のとき : Y r [0] , X r [1 . . 0] , ! B r , ! Y r [2 . . 1]

となる。

【 0 0 8 2 】

また、横方向のアドレス (図 1 8 の X 方向) を反転するだけで横方向の鏡像を、縦方向のアドレス (図 1 8 の Y 方向) を反転するだけで縦方向の鏡像を出力させることができる。

【 0 0 8 3 】

なお、この場合の有効信号は、

0 ° のとき : $X r < W$ かつ $Y r < W$ のとき有効

9 0 ° のとき : ! $X r < W$ かつ $Y r < W$ のとき有効

1 8 0 ° のとき : ! $X r < W$ かつ ! $Y r < W$ のとき有効

2 7 0 ° のとき : $X r < W$ かつ ! $Y r < W$ のとき有効

となる。

【 0 0 8 4 】

さらに、4 : 2 : 2 モードで、無効なブロックの読み出しを削減するため、Y ブロックバッファの読み出しアドレスを、

0 ° のとき : 0 , Y r [0] , Y r [3 . . 1] , X r

9 0 ° のとき : 0 , Y r [3] , ! X r , Y r [2 . . 0]

1 8 0 ° のとき : 0 , ! Y r [0] , ! Y r [3 . . 1] , ! X r

2 7 0 ° のとき : 0 , ! Y r [3] , X r , ! Y r [2 . . 0]

C (C r , C b) ブロックバッファの読み出しアドレスを $W > 4$ の時、

0 ° のとき : Y r [3 . . 1] , Y r [0] , X r [2 . . 1]

9 0 ° のとき : ! X r , Y r [3 . . 1]

1 8 0 ° のとき : ! Y r [3 . . 1] , ! Y r [0] , ! X r [2 . . 1]

2 7 0 ° のとき : X r , ! Y r [3 . . 1]

C (C r , C b) ブロックバッファの読み出しアドレスを $W = 4$ の時、

0 ° のとき : Y r [3 . . 1] , Y r [0] , X r [1 . . 0]

9 0 ° のとき : ! X r , Y r [3] , Y r [1 . . 0]

1 8 0 ° のとき : ! Y r [3 . . 1] , ! Y r [0] , ! X r [1 . . 0]

2 7 0 ° のとき : X r , ! Y r [3] , ! Y r [1 . . 0]

10

20

30

40

50

に設定するようにしても良い。この場合は図 18 の灰色のブロックのみ読み出される。

【 0 0 8 5 】

なお、本実施形態では、ブロックバッファの読み出し時にブロック内回転を行う構成としたが、図 19 及び図 20 に示すようにブロックバッファへのデータ書き込み時にブロック内回転した位置へデータを格納するようにしても良い。すなわち、図 19 は、書き込み時にブロック内回転した時の輝度信号データが Y ブロックバッファに格納される様子を示す図である。また、図 20 は、書き込み時にブロック内回転した時の色差信号データが Cr ブロックバッファ及び Cb ブロックバッファに格納される様子を示す図である。

【 0 0 8 6 】

また、本実施形態では、読み出し時に 8 × 8 画素単位の識別が可能であるので、読み出し時の有効信号を操作することにより、8 × 8 画素単位のクリッピングを行うことも可能である。この場合、MCU 単位、DCT ブロック (8 × 8 画素) 単位、画素単位の 3 段階のクリッピングになる。あるいは MCU 単位のクリッピングを省略して、DCT ブロック (8 × 8 画素) 単位、画素単位の 2 段階のクリッピングとしても良い。

【 0 0 8 7 】

さらに、点順次化された YCrCb データは RGB 変換回路 116 にて RGB データに変換され、ページバッファ 117 に格納される。

【 0 0 8 8 】

さらにまた、ページバッファ 117 への格納は、上記 MCU 単位で行われる。

【 0 0 8 9 】

MCU の先頭アドレスは、上記ブロック内の並び替えと同様に、横方向の MCU カウンタ及び縦方向の MCU カウンタより生成できる。但し、この場合、複数の乗算が必要となり、アドレス生成が複雑になる。そこで、現画素のアドレスに差分アドレスを加算して次画素のアドレスを求める方法 (以下、「差分アドレッシング法」と記す) によりページバッファの格納アドレスを生成する。即ち、直前画素のアドレスを $A(n-1)$ 、差分アドレスを D としたとき、現画素のアドレス $A(n)$ を、

$$A(n) = A(n-1) + D \quad (2)$$

とする。

【 0 0 9 0 】

差分アドレス D は、回転やサンプリングモードにより下記に示すシーケンスにて切り替わる。ここで、MCU ラインエンド (右端の MCU の最終画素) における差分アドレスを D_l 、MCU ラインエンド以外の MCU エンド (MCU の最終画素) における差分アドレスを D_m 、MCU エンド以外のブロック内ラインエンド (MCU 内の右端画素) における差分アドレスを D_b とする。また、MCU 内の横方向カウンタの値を x 、MCU 内の縦方向カウンタの値を y 、横方向の MCU カウンタの値を M_x 、縦方向の MCU カウンタの値を M_y 、回転後の MCU サイズを $W_x \times W_y$ 、横方向の MCU 数を W_m 、縦方向の MCU 数を H_m とする。この時の各差分アドレスの切り替えタイミングを下記に示す。

【 0 0 9 1 】

$D = D_l$: $x = W_x - 1$ 、 $y = W_y - 1$ 、 $z = m - 1$ の時

$D = D_m$: $x = W_x - 1$ 、 $y = W_y - 1$ 、 $z = m - 1$ の時

$D = D_b$: $x = W_x - 1$ 、 $y = W_y - 1$ の時

$D = 1$: 上記以外

ここで、 $z = M_x$ 、 $m = W_m$: 0° 、 180° 回転時、

$z = M_y$ 、 $m = H_m$: 90° 、 270° 回転時である。

【 0 0 9 2 】

また、各差分アドレス及びアドレスの初期値 $A(0)$ は、ページバッファの先頭アドレスを A_s とすると、

$D_b = W_m \times W_x - W_x + 1$: 各回転モード共通

[0° 回転]

$A(0) = A_s$

10

20

30

40

50

$$D1 = 1$$

$$Dm = -Wm \times Wx \times (Wy - 1) + 1$$

[90°回転]

$$A(0) = As + (Wm - 1) \times Wx$$

$$D1 = -Wm \times Wx \times (Hm \times Wy - 1) - 2Wx + 1$$

$$Dm = Wm \times Wx - Wx + 1$$

[180°回転]

$$A(0) = As + Wm \times Wx \times (Hm - 1) \times Wy + (Wm - 1) \times Wx$$

$$D1 = -Wm \times Wx \times (2Wy - 1) + (Wm - 2) \times Wx + 1$$

$$Dm = -Wm \times Wx \times (Wy - 1) - 2Wx + 1$$

[270°回転]

$$A(0) = As + Wm \times Wx \times (Hm - 1) \times Wy$$

$$D1 = Wm \times Wx \times ((Hm - 2) \times Wy + 1) + 1$$

$$Dm = -Wm \times Wx \times (2Wy - 1) - Wx + 1$$

となる。

【 0093 】

例えば、図12の例では、MCUクリッピング回路により101×151MCUの画像になり、MCU縮小回路にて16×8画素のMCUは6×3画素のMCUサイズに縮小され、ブロック内回転回路にて90°回転されて出力される。従って、MCU縮小回転後のMCUサイズは3×6となる。即ち、 $Wx = 3$ 、 $Wy = 6$ 、 $Wm = 151$ 、 $Hm = 101$ である。よって、ページバッファの先頭アドレス $As = 0$ とおくと、 $A(0) = 450$ 、 $D1 = -274070$ 、 $Dm = Db = 451$ となる。MCU内の横方向カウンタ x は3進カウンタで、このカウンタのキャリーでMCU内の縦方向カウンタ y がカウントアップする。MCU内の縦方向カウンタ y は6進カウンタで、このカウンタのキャリーでMCUカウンタ My がカウントアップする。MCUカウンタ My は101進カウンタで、0から100までカウントアップし、0に戻る。なお、横方向のMCUカウンタは不要である。

【 0094 】

差分アドレス D は、1, 1, 451, 1, 1, 451, ..., 1, 1, -274070, ... のように、3回に1回451に、1818回(101×6×3回)に1回-274070に切り替わる。即ち、上記カウンタのキャリーに同期して切り替わる。

【 0095 】

上述のように、回転モード、MCU縮小回転後のMCUサイズ、画像の縦及び横のMCU数によって上記パラメータ Wx 、 Wy 、 Wm 、 Hm が決定され、MCU内の横方向カウンタ、MCU内の縦方向カウンタ、MCUカウンタの構成が決まる。また、上記差分アドレス $D1$ 、 Dm 、 Db は上記パラメータで一意に決まり、パラメータに変更がない限り一定である。したがって、上記差分アドレス $D1$ 、 Dm 、 Db 及びアドレスの初期値 $A(0)$ をレジスタに格納することにより、複雑な演算なしにアドレス生成が可能である。また、上述したように、通常は $D = 1$ となる(アドレスが増加する方向に連続する)ので、バーストライトによるメモリの使用効率の向上が可能である。

【 0096 】

ページバッファ上には回転された状態で格納されているので、ページバッファ117からの読み出しは、モードに関係なくページバッファの先頭アドレス As より順次(連続して)行われる。読み出された画像データは、上記MCUクリッピング回路111でクリッピングされなかった画素を画素クリッピング回路118にてクリップし、リサイズ回路119にて所望のサイズに変換した後、出力部よりプリントエンジンに出力される。例えば、図12の例では、ページバッファ上に453×606画素のサイズで格納されている。これをユーザ設定領域である450×600画素(1200×1600の3/8倍)にクリップして、さらに、出力画像サイズである480×640画素に拡大(16/15倍)して出力部より出力する。

【 0097 】

10

20

30

40

50

本実施形態ではMCU縮小回路にて目的のサイズ付近にまで縮小する。図12の例では1200×1600画素を480×640画素にしたいので、縮小率は480/1200=0.4となる。この値は1/8(=0.125)単位の縮小率の中では3/8(=0.375)が最も近い。従って、MCU縮小回路にて3/8に縮小し、後段のリサイズ回路で16/15倍することで、 $3/8 \times 16/15 = 0.4$ を実現する。

【0098】

上述したプリントエンジンの同期信号に従って画像信号を出力するような場合、1/2以下の縮小を1度で実行すると縮小後のサンプリングポイントが離れてしまうので、補間に用いる画素の読み出しは不連続になってしまう。従って、ページバッファをDRAM上に構成した場合は、メモリの使用効率が下がる。一方、本実施形態では1/16以上の縮小率においては、リサイズ回路119での縮小率は1/2以上となり、連続読み出しにて対応可能となる(リサイズ回路119にラインメモリを装備した場合は先頭より画像データを読み出すだけでよい)。即ち、メモリアクセスの連続性が向上し、メモリ使用効率が向上する。また、JPEG復号化回路のQテーブルを操作し、折り返しノイズとなる高周波成分を排除してから逆DCTする構成により、折り返しノイズ除去のためのプリフィルタが不要となる。また、縮小率が1/2以下においては色差信号のサブサンプリングによる補間を行わないようにしたので、色差信号の解像度の劣化が抑えられる。

10

【0099】

なお、上記説明ではページバッファにて並び替えを行っているが、これに限らず、バンドバッファ(ページバッファの一部)でも良い。

20

【0100】

以上、本発明の実施形態を説明したが、これに限らず本発明は、複数の機器(例えば、ホストコンピュータ、インタフェース機器、リーダ、プリンタ等)から構成されるシステムに適用しても、一つの機器からなる装置(例えば、複写機、ファクシミリ装置等)に適用してもよい。

【0101】

また、本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記録媒体(または記憶媒体)を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ(またはCPUやMPU)が記録媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言うまでもない。この場合、記録媒体から読み出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記録した記録媒体は本発明を構成することになる。また、コンピュータが読み出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているオペレーティングシステム(OS)などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

30

【0102】

さらに、記録媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張カードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張カードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

40

【0103】

本発明を上記記録媒体に適用する場合、その記録媒体には、先に説明したフローチャートに対応するプログラムコードが格納されることになる。

【0104】

【発明の効果】

以上説明したように、本発明によれば、画像データに対して処理に必要な範囲を一定のブロック単位で指定して必要な範囲にだけ画像処理を実施し、その後の画像処理より詳細な

50

画像処理を行うので、画像データ全体に対する画像処理速度を高速化するとともに、画像処理に必要とされるメモリも小さくすることができる。また、メモリアクセスの連続性を向上させ、メモリ使用効率を向上することができる。

【図面の簡単な説明】

【図 1】本発明の第 1 の実施形態による画像処理装置の概要を説明するためのブロック図である。

【図 2】図 1 における前段画像処理部 2 1 の細部構成を説明するためのブロック図である。

【図 3】クリッピング処理部 2 1 1 で実施されるクリッピング処理について説明するための概略図である。

10

【図 4】並び替え処理部 2 1 3 で行われる回転処理を説明するための概略図である。

【図 5】第 1 の実施形態における並び替え処理部 2 1 3 の処理動作の具体例を説明するための図である。

【図 6】クリッピング処理以降の解像度変換処理と並び替え処理を詳細に説明するための図である。

【図 7】図 1 5 に示したそれぞれのサンプリングポイントの場合における補間係数を示す図である。

【図 8】第 1 の実施形態による画像処理装置の動作手順を説明するためのフローチャートである。

【図 9】ステップ S 8 2 でのクリッピング処理手順を詳細に説明するためのフローチャートである。

20

【図 1 0】ステップ S 8 3 で行われる解像度変換処理の詳細を説明するためのフローチャートである。

【図 1 1】本発明の第 2 の実施形態に係る画像処理装置の構成を示すブロック図である。

【図 1 2】第 2 の実施形態に係る画像処理装置による画像処理の具体例を説明するための図である。

【図 1 3】MCU 縮小回路 1 1 2 の細部構成を示すブロック図である。

【図 1 4】横サイズ縮小回路 1 3 1 又は縦サイズ縮小回路 1 3 2 の具体的な構成を示すブロック図である。

【図 1 5】MCU 縮小後のサンプリングポイントを具体的に示す図である。

30

【図 1 6】図 1 2 の具体例における輝度信号データが Y ブロックバッファに格納される様子を示した図である。

【図 1 7】図 1 2 の具体例における色差信号データが Cr ブロックバッファ及び Cb ブロックバッファに格納される様子を示した図である。

【図 1 8】回転モードと Y ブロックバッファの読み出し位置との関係を示す図である。

【図 1 9】書き込み時にブロック内回転した時の輝度信号データが Y ブロックバッファに格納される様子を示す図である。

【図 2 0】書き込み時にブロック内回転した時の色差信号データが Cr ブロックバッファ及び Cb ブロックバッファに格納される様子を示す図である。

【符号の説明】

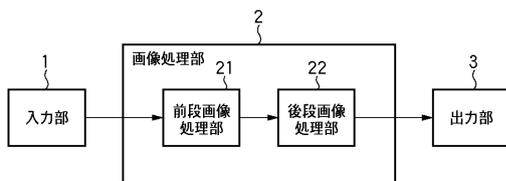
40

- 1 入力部
- 2 画像処理部
- 3 出力部
- 2 1 前段画像処理部
- 2 2 後段画像処理部
- 1 1 0 J P E G 復号化部
- 1 1 1 M C U クリッピング部
- 1 1 2 M C U 縮小部
- 1 1 3 ブロック内回転部
- 1 1 4 ブロックバッファ

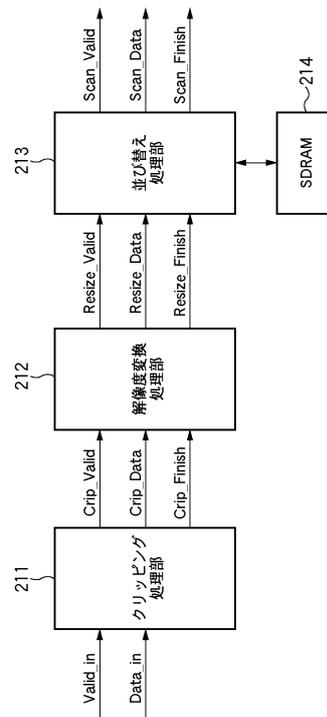
50

- 1 1 5 点順次化部
- 1 1 6 R G B 変換部
- 1 1 7 ページバッファ
- 1 1 8 画素クリッピング部
- 1 1 9 リサイズ部
- 1 3 1 横サイズ縮小部
- 1 3 2 縦サイズ縮小部
- 1 3 3 有効信号生成部
- 1 4 1 係数生成部
- 1 4 2、1 4 3 乗算器
- 1 4 4 バッファ
- 1 4 5 加算器
- 2 1 1 クリッピング処理部
- 2 1 2 解像度変換処理部
- 2 1 3 並び替え処理部
- 2 1 4 S D R A M

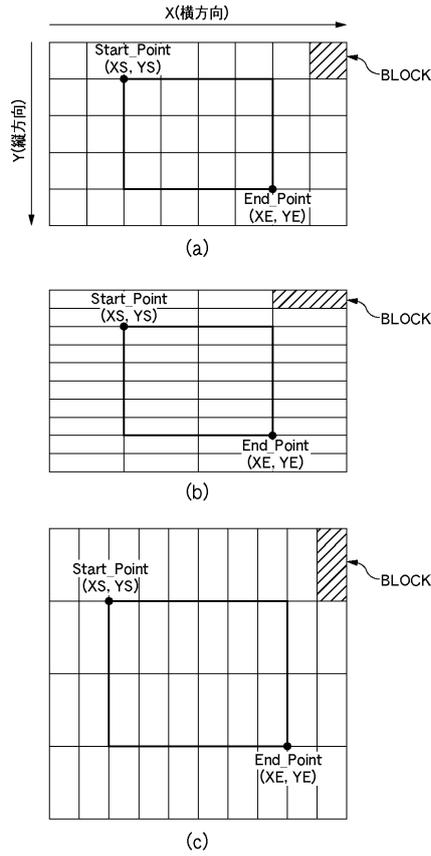
【図 1】



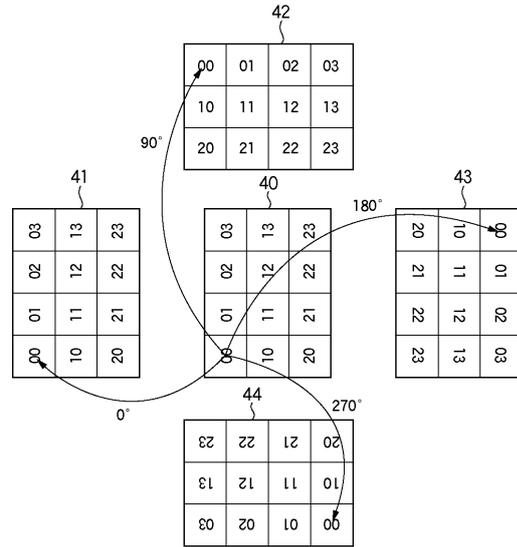
【図 2】



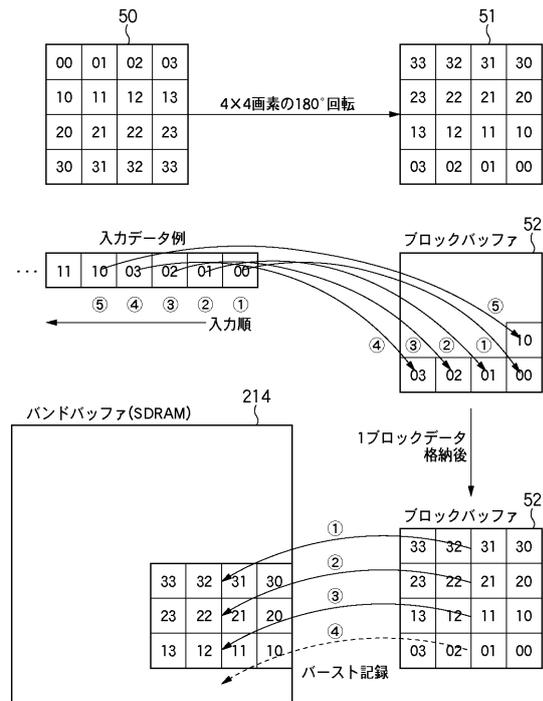
【図3】



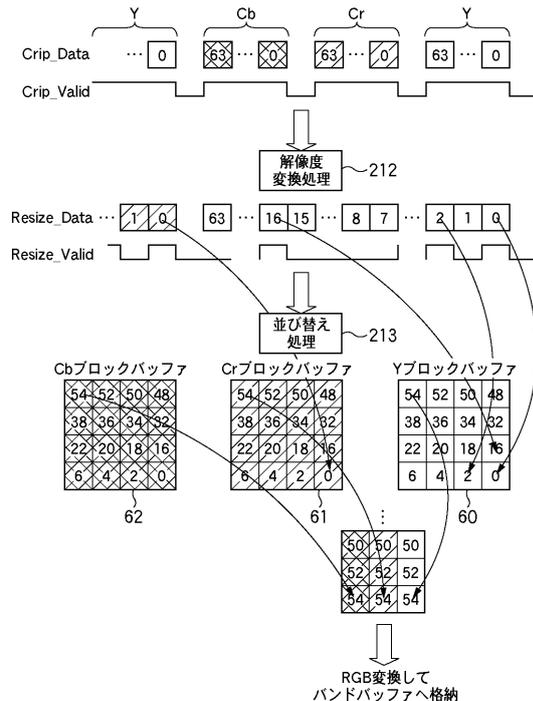
【図4】



【図5】



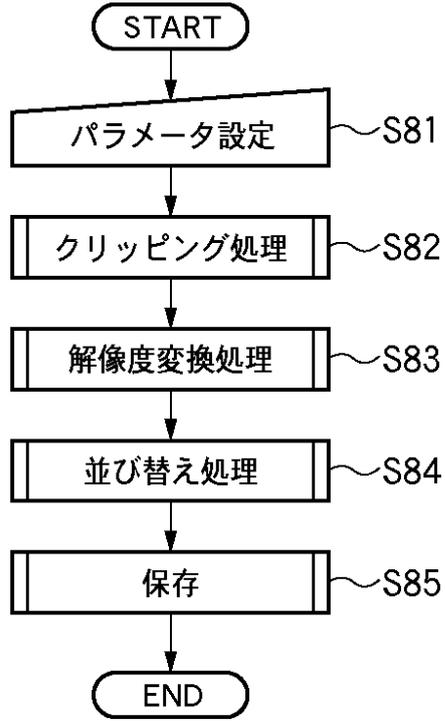
【図6】



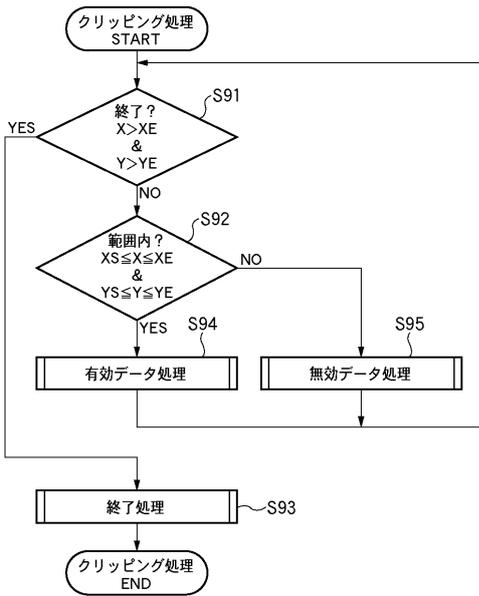
【図7】

		0	1	2	3	4	5	6	7
X	α	$1-\alpha$	α	$1-\alpha$	α	$1-\alpha$	α	$1-\alpha$	α
W	α	$1-\alpha$	α	$1-\alpha$	α	$1-\alpha$	α	$1-\alpha$	α
7	256	0	219	37	183	73	146	110	110
6	256	0	171	85	85	171	0	256	256
5	256	0	102	154	205	51	205	51	102
4	256	0	256	0	256	0	256	0	256
3	256	0	171	85	85	171	0	256	256
2	256	0	256	0	256	0	256	0	256
1	256	0	256	0	256	0	256	0	256

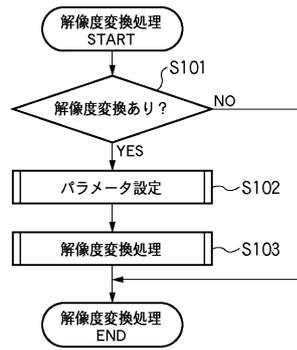
【図8】



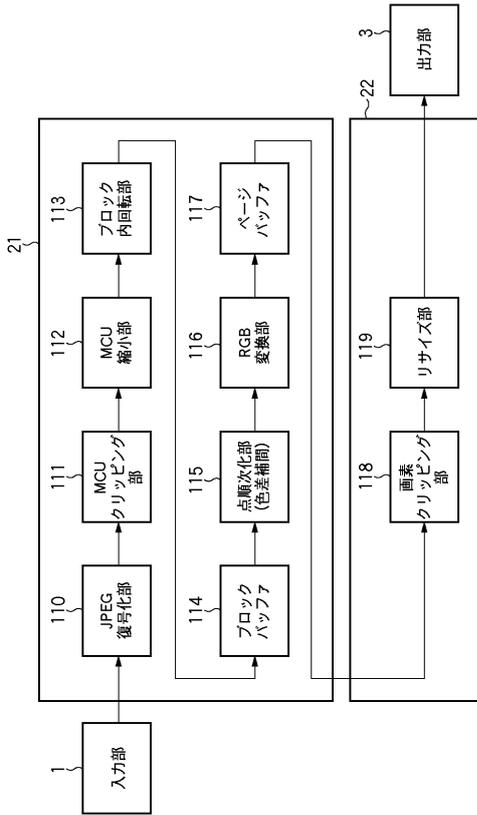
【図9】



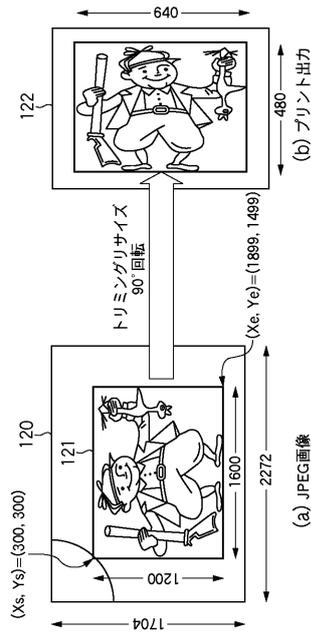
【図10】



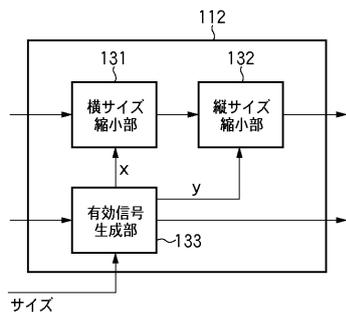
【図11】



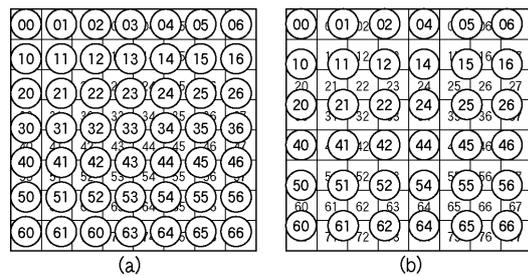
【図12】



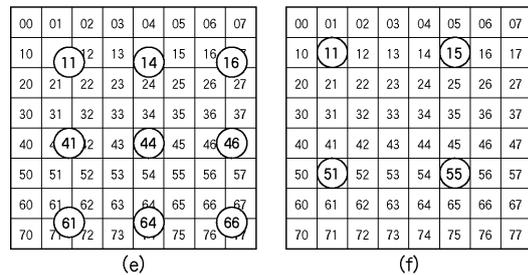
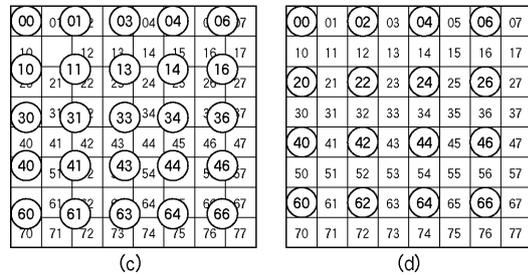
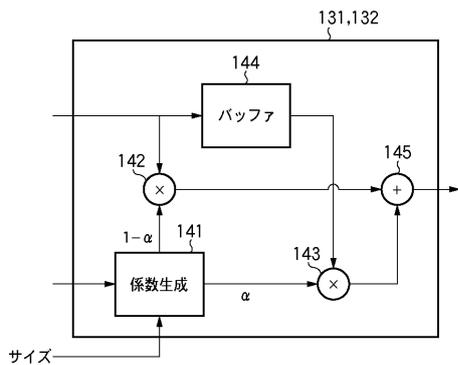
【図13】



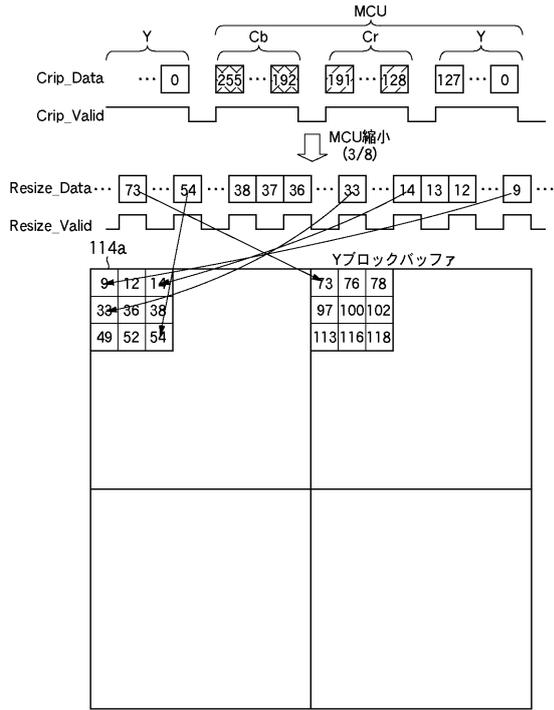
【図15】



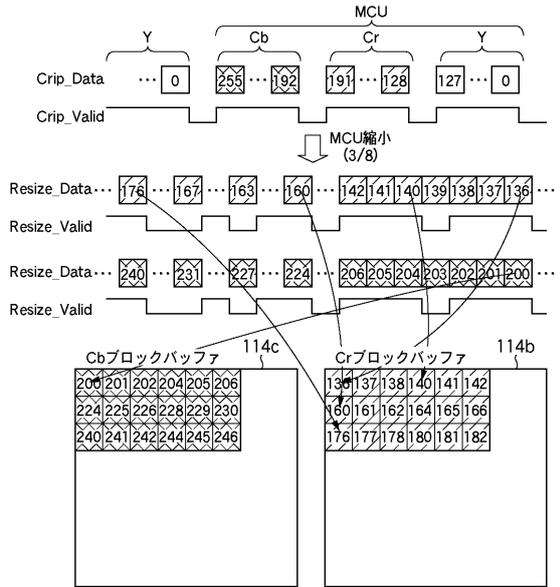
【図14】



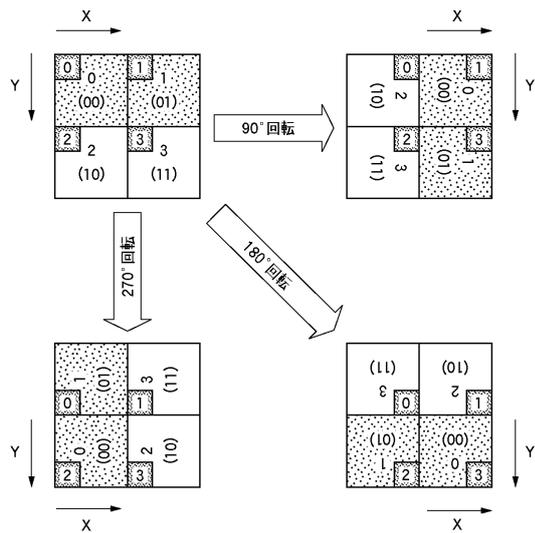
【図16】



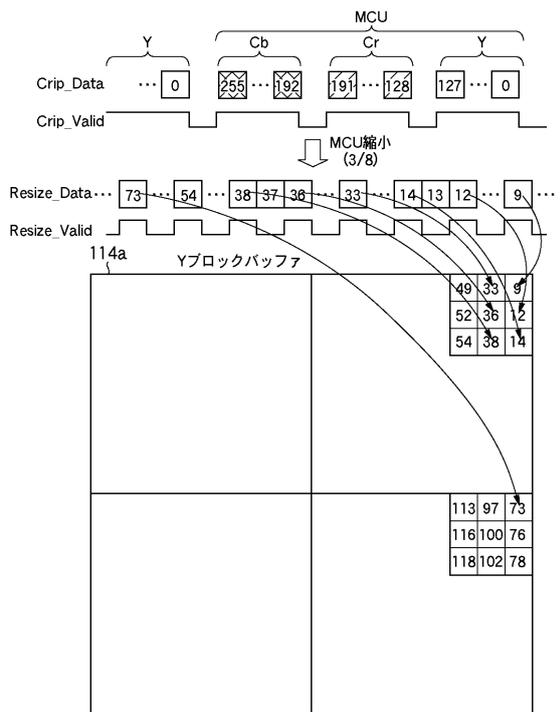
【図17】



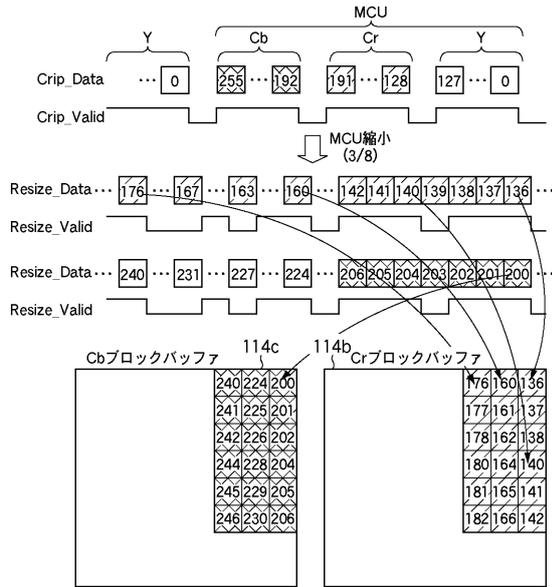
【図18】



【図19】



【 図 20 】



フロントページの続き

- (72)発明者 石川 尚
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
- (72)発明者 三瀬 良子
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

合議体

審判長 吉村 博之

審判官 千葉 輝久

審判官 溝本 安展

- (56)参考文献 特開平11-168610(JP,A)

- (58)調査した分野(Int.Cl., DB名)
H04N1/38-1/393