

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第4171902号
(P4171902)

(45) 発行日 平成20年10月29日(2008.10.29)

(24) 登録日 平成20年8月22日(2008.8.22)

(51) Int.Cl.		F I			
HO4N	1/387	(2006.01)	HO4N	1/387	1 O 1
GO6F	3/12	(2006.01)	GO6F	3/12	C
GO6T	3/40	(2006.01)	GO6F	3/12	L
			GO6T	3/40	C

請求項の数 4 (全 53 頁)

(21) 出願番号	特願2003-157882 (P2003-157882)	(73) 特許権者	000002369
(22) 出願日	平成15年6月3日(2003.6.3)		セイコーエプソン株式会社
(62) 分割の表示	特願2001-13560 (P2001-13560) の分割	(74) 代理人	100096703 弁理士 横井 俊之
原出願日	平成11年3月29日(1999.3.29)	(72) 発明者	鎌田 直樹 長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内
(65) 公開番号	特開2004-96715 (P2004-96715A)	審査官	松永 隆志
(43) 公開日	平成16年3月25日(2004.3.25)		
審査請求日	平成18年3月28日(2006.3.28)		
(31) 優先権主張番号	特願平10-93739		
(32) 優先日	平成10年4月6日(1998.4.6)		
(33) 優先権主張国	日本国(JP)		
前置審査			

最終頁に続く

(54) 【発明の名称】 オブジェクト画素抽出装置、オブジェクト画素抽出方法、オブジェクト画素抽出プログラムを記録した媒体およびオブジェクト画素抽出プログラム

(57) 【特許請求の範囲】

【請求項1】

ドットマトリクス状の画素で表現した画像の画像データを取得する画像データ取得手段と、

上記画像データについて、画像の中の中央部分から多くの画素が抽出されるように画素を抽出し、抽出した画素を対象として輝度分布を求めるとともに、輝度がとりうる階調範囲の全輝度値のうち当該輝度分布における分布数が0ではない輝度値の数を色数とみなすことにより利用色数を算出する手段とを具備することを特徴とするオブジェクト画素抽出装置。

【請求項2】

ドットマトリクス状の画素で表現した画像の画像データを取得する工程と、

上記画像データについて、画像の中の中央部分から多くの画素が抽出されるように画素を抽出し、抽出した画素を対象として輝度分布を求めるとともに、輝度がとりうる階調範囲の全輝度値のうち当該輝度分布における分布数が0ではない輝度値の数を色数とみなすことにより利用色数を算出する工程とを具備することを特徴とするオブジェクト画素抽出方法。

【請求項3】

ドットマトリクス状の画素で表現した画像の画像データに基づいてコンピュータにてオブジェクト画素を抽出させるオブジェクト画素抽出プログラムを記録した媒体であって、上記画像データを取得する機能と、

上記画像データについて、画像の中の中央部分から多くの画素が抽出されるように画素を抽出し、抽出した画素を対象として輝度分布を求めるとともに、輝度がとりうる階調範囲の全輝度値のうち当該輝度分布における分布数が0ではない輝度値の数を色数とみなすことにより利用色数を算出する機能とをコンピュータに実現させることを特徴とするオブジェクト画素抽出プログラムを記録した媒体。

【請求項4】

ドットマトリクス状の画素で表現した画像の画像データに基づいてコンピュータにてオブジェクト画素を抽出させるオブジェクト画素抽出プログラムであって、

上記画像データを取得する機能と、

上記画像データについて、画像の中の中央部分から多くの画素が抽出されるように画素を抽出し、抽出した画素を対象として輝度分布を求めるとともに、輝度がとりうる階調範囲の全輝度値のうち当該輝度分布における分布数が0ではない輝度値の数を色数とみなすことにより利用色数を算出する機能とをコンピュータに実現させることを特徴とするオブジェクト画素抽出プログラム。

10

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、オブジェクト画素抽出装置、オブジェクト画素抽出方法、オブジェクト画素抽出プログラムを記録した媒体およびオブジェクト画素抽出プログラムに関する。

【0002】

20

【従来の技術】

コンピュータなどで画像を扱う際には、画像をドットマトリクス状の画素で表現し、各画素を階調値で表している。例えば、コンピュータの画面で水平方向に640ドット、垂直方向に480ドットの画素で写真やコンピュータグラフィックスを表示することが多い。

【0003】

一方、カラープリンタの性能向上がめざましく、そのドット密度は720dot/inch(dpi)というように極めて高精度となっている。すると、640×480ドットの画像をドット単位で対応させて印刷させようとするとき極めて小さくなってしまふ。この場合、階調値も異なる上、解像度の意味合い自体が異なるのであるから、ドット間を補間して印刷用のデータに変換しなければならない。

30

【0004】

従来、このような場合にドットを補間する手法として、最近隣内挿法(ニアリストネイバ補間:以下、ニアリスト法と呼ぶ)や、3次たまたみ込み内挿法(キュービックコンボリューション補間:以下、キュービック法と呼ぶ)などの手法が知られている。また、特開平6-225140号公報にはドットを補間したときの縁部のスムージングを行うにあたり、予め縁部がスムーズとなるような拡大形態となるようにドットパターンを用意しておく技術が開示されている。

【0005】

【発明が解決しようとする課題】

上述した従来の補間技術においては、次のような課題があった。

40

ニアリスト法やキュービック法などの各種の手法にはそれぞれに得失があるが、利用者がそれを選択するのは難しく、また、どちらか一方に固定したとすれば、不得手な画像に対して補間結果の品質が低下する。

特開平6-225140号公報に開示された発明においては、予めパターンを用意しておくことから補間倍率が固定的にならざるを得ないし、カラーの画像を前提とするとパターンの数が膨大となって予め用意しておくこと自体が困難である。

【0006】

本発明は、上記課題にかんがみてなされたもので、カラー画像を含めて効率よく補間するに際してオブジェクトの画素を抽出することが可能なオブジェクト画素抽出装置、オブジェクト画素抽出方法、オブジェクト画素抽出プログラムを記録した媒体およびオブジェ

50

クト画素抽出プログラムの提供を目的とする。

【0007】

【課題を解決するための手段】

本発明で提供されるオブジェクト画素抽出装置は、ドットマトリクス状の画素で表現した画像の画像データを取得する画像データ取得手段と、上記画像データについて、画像の中の中央部分から多くの画素が抽出されるように画素を抽出し、抽出した画素を対象として輝度分布を求めるとともに、輝度がとりうる階調範囲の全輝度値のうち当該輝度分布における分布数が0ではない輝度値の数を色数とみなすことにより利用色数を算出する手段とを具備する構成としている。本発明によれば、画像の中央部分から多くの画素が抽出されるような仕組みとすることでオブジェクトの画素がより抽出されやすくなる。

10

また、本願で提供される画像データ補間装置は、画像をドットマトリクス状の画素で表現した画像データを取得する画像データ取得手段と、上記画像データにおける構成画素数を増やす補間処理を行うにあたり複数の補間処理の中から選択して実行可能な画素補間手段と、上記画像データについての上記補間処理に関連する特徴量を取得する特徴量取得手段と、この特徴量取得手段によって取得された特徴量に対応して最適な補間結果を得ることが可能な補間処理を選択して上記画素補間手段に実行させる補間処理選択手段とを具備する構成とされている。

【0008】

このように構成した本発明においては、画像をドットマトリクス状の画素で表現した画像データの構成画素数を増やす補間処理を行うにあたり、画素補間手段は複数の補間処理の中からいずれかを選択して実行可能となっており、画像データ取得手段が対象となる画像データを取得すると、特徴量取得手段は同画像データについての上記補間処理に関連する特徴量を取得し、補間処理選択手段はこの特徴量取得手段によって取得された特徴量に対応して最適な補間結果を得ることが可能な補間処理を選択して上記画素補間手段に実行させる。すなわち、自ら画像の特徴量を取得して最適な補間処理を選択する。

20

【0009】

このように、画像の特徴に応じて最適な補間処理を選択する手法は必ずしも実体のある装置に限られる必要はなく、その方法としても機能することは容易に理解できる。このため、本発明で提供される画像データ補間方法は、画像をドットマトリクス状の画素で表現した画像データについて構成画素数を増やす補間処理を行うにあたり、上記画像データを取得する工程と、上記画像データについての上記補間処理に関連する特徴量を取得する工程と、この取得された特徴量に対応して複数の補間処理の中から最適な補間結果を得ることが可能な補間処理を選択する工程と、この複数の補間処理の中から選択された補間処理を実行する工程とを具備する構成とされている。

30

【0010】

すなわち、必ずしも実体のある装置に限らず、その方法としても有効であることに相違はない。

このような画像データ補間装置は単独で存在する場合もあるし、ある機器に組み込まれた状態で利用されることもあるなど、発明の思想としてはこれに限らず、各種の態様を含む。従って、ソフトウェアであったりハードウェアであったりするなど、適宜、変更可能である。

40

発明の思想の具現化例として画像データ補間装置のソフトウェアとなる場合には、かかるソフトウェアを記録した記録媒体上においても本発明は当然に存在し、利用されるといわざるをえない。

【0011】

その一例として、本発明で提供されるコンピュータにて補間処理を実行させる画像データ補間プログラムを記録した媒体は、画像をドットマトリクス状の画素で表現した画像データについて構成画素数を増やす補間処理を行うにあたり、上記画像データを取得するステップと、上記画像データについての上記補間処理に関連する特徴量を取得するステップと、この取得された特徴量に対応して複数の補間処理の中から最適な補間結果を得ることが

50

可能な補間処理を選択するステップと、この複数の補間処理の中から選択された補間処理を実行するステップとをコンピュータに実行させる構成としている。

【0012】

むろん、その記録媒体は、磁気記録媒体であってもよいし光磁気記録媒体であってもよいし、今後開発されるいかなる記録媒体においても全く同様に考えることができる。また、一次複製品、二次複製品などの複製段階については全く問う余地無く同等である。その他、供給方法として通信回線を利用して行なう場合でも本発明が利用されていることにはかわりない。

さらに、一部がソフトウェアであって、一部がハードウェアで実現されている場合においても発明の思想において全く異なるものではなく、一部を記録媒体上に記憶しておいて必要に応じて適宜読み込まれるような形態のものとしてあってもよい。

10

なお、本発明をソフトウェアで実施する場合、発明がプログラムを記録した媒体として実現されるのみならず、本発明がプログラム自体として実現されるのは当然であり、プログラム自体も本発明に含まれる。

【0013】

ここで、画像データは画像をドットマトリクス状の画素で表現したものであり、各画素についてデータで表したものであればよく、カラー画像であってもよいし、モノクロ画像であってもよい。また、階調値は二階調のものであってもよいし、多階調のものであってもよい。

画像データ取得手段は、かかる画像データを取得するものであり、上記画素補間手段が構成画素を増やすための補間処理を行うにあたり、対象となる画像データを保持するようなものであればよい。従って、その取得手法は特に限定されるものではなく、各種のものを採用可能である。例えば、インターフェイスを介して外部機器から取得するものであってもよいし、撮像手段を備えて画像を撮像するものであってもよい。また、コンピュータグラフィックアプリケーションを実行してマウスやキーボードから入力するものであってもよい。

20

【0014】

画素補間手段は、各種の手法により複数の補間処理を選択可能なものであればよく、これは実質的に補間処理結果が異なる選択肢を備えるものであればよい。従って、全ての選択肢が独立した補間手法である必要はない。

30

その一例として、上記画素補間手段は、上記補間処理の選択肢として複数の補間処理を重ねて実行する補間処理を有する構成としてあるすることもできる。

このように構成した場合には、上記画素補間手段が補間処理の選択肢の一つとして複数の補間処理を重ねて実行する。すなわち、最初にある補間処理で画素を補間した後、重ねて別の補間処理で画素を補間する。これにより独立した補間処理としては二種類であっても、それを重ねて行う処理も含めて三種類の選択肢といえる。

【0015】

従って、補間処理の対応例として複数の補間処理を重ねて実行することが可能であるので、補間処理のバリエーションを増やすことができる。

また、実行する補間処理は全画素について一定のものに限る必要もない。その一例として、上記特徴量取得手段は、画像の一部の領域ごとに特徴量を取得し、上記補間処理選択手段は、当該領域毎に取得された特徴量に基づいて上記画素補間手段における補間処理を選択して実行させる構成とすることもできる。

40

【0016】

このように構成した場合には、上記特徴量取得手段が画像の一部の領域ごとに特徴量を取得する。各補間処理の得失によっては一律に適用することが好ましいものもあればそうでないものもある。そして、一律でなくても構わない場合にはそれぞれの特長を活かすことが好ましい。このため、一部の領域毎に取得された特徴量に基づいて上記補間処理選択手段は上記画素補間手段における補間処理を選択して実行させる。

【0017】

50

例えば、画像に変化のないところでは単純な補間処理を採用したとしても画質の劣化は分かりにくい、変化の激しいところでは補間処理の手法が単純であると画質の劣化が分かりやすくなってしまふ。従って、画素の補間は既存の画素の間に画素を増やすことになるので増やす画素について補間処理を適宜変更することが可能である。

従って、画像全体に一律の補間処理を施す必要が無くなり、総合的な意味で最適な結果を得ることが可能となる。

【 0 0 1 8 】

補間処理を選択するのは補間処理に応じて補間結果が異なるからであり、補間手法を選択する鍵となる特徴量もさまざまである。その一例として、上記画素補間手段は、自然画と非自然画のそれぞれに最適な補間処理を実行可能であり、上記特徴量取得手段は、上記画像データが表す画像が自然画か非自然画であるかを判定する特徴量を取得し、上記補間処理選択手段は、同特徴量に基づいて画像が自然画であると判定されるときに上記画素補間手段にて自然画に最適な補間処理を実行させるとともに非自然画であると判定されるときに上記画素補間手段にて非自然画に最適な補間処理を実行させる構成とすることもできる。

10

【 0 0 1 9 】

このように構成した場合には、その前提として上記画素補間手段が自然画と非自然画のそれぞれに最適な補間処理を実行可能であることとすると、上記画像データが表す画像が自然画か非自然画であるかを判断し得る特徴量を上記特徴量取得手段が取得し、上記補間処理選択手段は、同特徴量に基づいて画像が自然画であると判定されるときに上記画素補間手段にて自然画に最適な補間処理を実行させるとともに非自然画であると判定されるときに上記画素補間手段にて非自然画に最適な補間処理を実行させる。

20

【 0 0 2 0 】

従って、身近な処理対象である自然画と非自然画とに応じた最適な補間処理を選択することができるようになる。

自然画とはいわゆる実写画像のようなものを意味し、非自然画はビジネスグラフに代表されるコンピュータグラフィックスなどを意味する。特徴的には自然画であれば多色を利用していると考えられるし、非自然画であれば利用色数は少ないといえる。むろんこの他に見出される特徴に対応するものであってもよいが、画素補間手段はこのような特徴的な差異を有する自然画と非自然画のそれぞれに最適な補間処理を実行可能となっている。

30

【 0 0 2 1 】

より具体的には、上記画素補間手段は、非自然画に対して最近隣内挿法による補間処理を実行するとともに自然画に対して3次たたまみ込み内挿法による補間処理を実行する構成とすることができる。

このようにすれば、非自然画の場合に最近隣内挿法による高速処理というメリットが得られ、自然画の場合に3次たたまみ込み内挿法によるシャープさの保持というメリットが得られる。

非自然画と自然画とでは画素あたりの表現量の多少に差が出てくることが多く、最近隣内挿法は処理が単純であるために高速処理が可能である反面、自然画に対しては不自然な感じが表れることになりかねない。一方、3次たたまみ込み内挿法はシャープさを保持できるものの処理は複雑であり、補間処理に要する演算負荷が大きく、処理時間が多くかかるといえる。

40

【 0 0 2 2 】

このような得失を前提として、非自然画に対して最近隣内挿法による補間処理を実行することになるので、処理の高速化が期待でき、自然画に対して3次たたまみ込み内挿法による補間処理を実行するのでシャープさが確保されることになる。

特徴量取得手段は画像データについての特徴量を取得するが、この特徴量自体は補間処理に関連する性質のものである。上述したようにその一例としては自然画であるか非自然画であるかといったことを判定することにより得られるが、その取得手法はさまざまである。

50

【 0 0 2 3 】

その一例として、上記特徴量取得手段は、取得された画像データにおける各画素のデータを所定の基準で集計して当該画像データが表す画像の種類を判定する特徴量を取得する構成とすることもできる。

このように構成した場合には、特徴量取得手段が各画素のデータを所定の基準で集計する。すなわち、画像データ自身からその画像の内容の種類を判定する特徴量を取得する。むしろ、集計手法は各種の統計手法などを採用可能であるが、ヒストグラムなどもその一例である。

【 0 0 2 4 】

このようにすれば、画像データを集計して画像の種類を判定するので、他の情報が無い場合でも確実に最適な補間処理を選択することができる。

集計するデータとしては、例えば、利用色数といったものが上げられる。上述したようにコンピュータグラフィックスでは描画する人間が色を指定すると考えると、さほど沢山の色を利用できない。特に、ビジネスグラフなどではその傾向が強く表れる。一方、自然画であれば一色のものであっても光線の加減で多色となり得、利用色数は極端に多くなる。従って、画像データについて利用色数を集計すればその画像が自然画か非自然画かといったことを判定できることになる。

【 0 0 2 5 】

この場合の利用色数は厳密に集計する必要もない。例えば、各画素についての輝度のヒストグラムを集計し、各輝度の利用頻度から利用色数の多少を判定するようにしても良い。自然画について輝度のヒストグラムと利用色数との関連性は見出しにくいですが、ビジネスグラフなどを想定すると簡易に理解できる。ビジネスグラフのようなものは利用色数が少ないし、塗りつぶしのグラフなどを考えても数色しか利用しないことが多い。この場合、輝度のヒストグラムはスペクトル状となって表れる。異なる色において輝度が一致することもあり得るので、必ずしも輝度の利用頻度が利用色数とは一致しないが、多いか少ないかは容易に理解できる。例えば、自然画であるとする、陰影などをがあるので利用される輝度は確実に多くなり、ヒストグラムもスペクトル状というよりは、連続した曲線のように表れるはずである。

【 0 0 2 6 】

また、この場合に輝度自体も正確でなくて構わないのは、同様の趣旨から明らかである。例えば、RGBの画像データについての正確な輝度は、通常、演算だけで求められるものではない。しかしながら、テレビ放送などでは単純な重み付け加算でRGBから輝度を得ている。すなわち、このような簡易的な手法も利用可能である。

ヒストグラムがスペクトル状となるか曲線状となるかといったことも集計結果から導き出されるものであり、それに基づいて自然画であるか非自然画であるかを判定して特徴量と判断することもできる。また、自然画であるか否かという観点以外にも明るい画像であるか否かといった特徴量として取得することも可能である。すなわち、明るい画像に対して好適な補間処理と、暗い画像に対して好適な補間処理がある場合にはこれらを選択することが可能となる。

【 0 0 2 7 】

また、各画素のデータの変化度合いというものを集計することも可能である。コンピュータグラフィックスで絵を描いた場合、利用者が描いただけの変化度合いとなるが、風景を撮影した自然画であればその情報量は各段に多く、それは画素の変化度合いの大きさに表れてくる。従って、画素間の変化度合いからも画像の種類を判定することが可能となる。一方、特徴量取得手段は画像を判定するのが目的であるから、判定するのに必要な程度でだけ画像データを集計すればよいといえる。この意味で、上記特徴量取得手段は、上記画像データを構成する一部の画素についての上記データを抽出して集計する構成とすることもできる。

【 0 0 2 8 】

このように構成した場合には、上記画像データを構成する一部の画素についてだけ上記階

10

20

30

40

50

調値を抽出し、集計する。

ここで、抽出する手法については、各種の手法を採用可能である。単純なものとしてランダムに画素を選択して集計することも可能であるし、特定のオブジェクトを見出して集計するようにしても良い。また、集計しながら集計結果の傾向を予測するようなものでもよい。自然画か非自然画かを判定する例であれば、利用色がなだらかにその周辺を含むようなことを見出されればその時点で自然画であると判定することも可能である。また、極端に色数が多そうであることが分かればその時点で判定しても良い。

【0029】

このようにした場合には、画像データの全てを集計しないで行うため、処理を高速化することができる。

10

また、コンピュータなどを想定すると、画像データはファイルとして処理されるが、そのファイル形式から画像の種類が判定できることも少なくない。従って、上記画像データの形式に基づいて特徴量を取得する構成とすることもできる。例えば、自然画であればJPEG方式で圧縮されることが多い。従って、画像データの形式がJPEG方式であれば自然画と判定することができる。また、ビジネスグラフについてはそれを出力するアプリケーションを表すファイルの拡張子が付されていることが多い。従って、このような拡張子も一形式と判断して画像の種類を判定できる。

【0030】

このようにすれば、画像データの形式に基づいて画像の種類を判定するので、画像データを解析する場合に比べて処理量を少なくすることができる。

20

一方、補間処理に関連する特徴量というのは必ずしも画像の種類といった類のものでなければならぬわけでもない。従って、上記特徴量取得手段は、実行する補間倍率を取得して特徴量とする構成とすることもできる。

補間倍率も補間結果に関連する。例えば、補間倍率が小さい間は画像が荒くならないが、補間倍率が上がると画像が荒くなるということもある。この場合、後者のものであれば補間処理を複雑なものにしなければならぬとも言える。このため、特徴量取得手段が補間倍率を取得する。

【0031】

むしろ、この場合、選択する補間処理に応じて補間倍率のしきい値などは変化することになるし、ある補間処理から他の補間処理へ移行するにあたって段階的に変化させるといったことも可能である。

30

このようにすれば、画像データの種類以外の要素に基づいて最適な補間処理を選択することができる。

【0032】

【発明の効果】

以上説明したように、本発明によれば、画像の中央部分から多くの画素が抽出されるような仕組みとすることでオブジェクトの画素が抽出されやすくなる。

また、本願によれば、自ら画像の特徴量を取得して最適な補間処理を選択するようしており、画像の特徴に応じた補間処理が実施されるので、最適な補間結果を極めて容易に得ることが可能な画像データ補間装置、画像データ補間方法、画像データ補間プログラムを記録した媒体を提供することができる。

40

【0033】

さらに、画像データの簡易な分類として自然画と非自然画とに分類して最適な補間処理を実行することが可能となる。

また、非自然画に対してはその性質上最も好適な最近隣内挿法による補間処理を実行し、自然画に対してはその性質上最も好適な3次たたみ込み内挿法による補間処理を実行することが可能となる。

【0034】

さらに、画素の変化度合いであるシャープさに基づいて補間処理を選択することにより、最適な補間結果を得ることが可能となる。

50

さらに、画素の変化度合の小さい領域では極めて簡易な処理で補間処理を実行することが可能となる。

さらに、変化度合の大きい領域では画像データをなだらかに変化させて画質を向上させることが可能となる。

【 0 0 3 5 】

さらに、複数の補間処理を重ねて実行することにより実質的に利用可能な補間処理を増やしてきめ細かな処理を行うことが可能となる。

さらに、画像の領域ごとに最適な補間処理を選択して実行させることが可能となる。

さらに、演算パラメータを調整するだけで、一つの補間処理を実質的に複数の画素補間処理として利用することが可能となる。

【 0 0 3 6 】

さらに、画像の拡大処理とともに画像のシャープさの変更処理とが共に選択された場合に極めて効率的に処理することとなる。

さらに、同様の効果を奏する画像データ補間処理用コンピュータを提供することができる。

【 0 0 3 7 】

【 発明の実施の形態 】

以下、図面にもとづいて本発明の実施形態を説明する。

図 1 は、本発明の画像データ補間装置の主要構成を示すブロック図である。

デジタル処理を前提とすると、画像はドットマトリクス状の画素で表現することになり、各画素を表すデータの集まりで画像データが構成される。そして、画素単位で処理する系においては、画像の拡大縮小は画素単位で実施することになる。本画像データ補間装置はこのような画素単位での拡大処理を実施するものであり、画像データ取得手段 C 1 は、このような画像データを取得し、画素補間手段 C 2 はこの画像データにおける構成画素数を増やす補間処理を行う。ここで、画素補間手段 C 2 は補間処理として複数の補間処理を実行可能となっており、特徴量取得手段 C 3 が上記画像データについての上記補間処理に関連する特徴量を取得すると、補間処理選択手段 C 4 はその特徴量に対応して最適な補間結果を得ることが可能な補間処理を選択し、上記画素補間手段 C 2 に実行させる。

【 0 0 3 8 】

本実施形態においてはこのような画像データ補間装置を実現するハードウェアの一例としてコンピュータシステム 1 0 を採用している。

図 2 は、同コンピュータシステム 1 0 をブロック図により示している。

本コンピュータシステム 1 0 は、画像入力デバイスとして、スキャナ 1 1 a とデジタルスチルカメラ 1 1 b とビデオカメラ 1 1 c とを備えており、コンピュータ本体 1 2 に接続されている。それぞれの入力デバイスは画像をドットマトリクス状の画素で表現した画像データを生成してコンピュータ本体 1 2 に出力可能となっており、ここで同画像データは R G B の三原色においてそれぞれ 2 5 6 階調表示することにより、約 1 6 7 0 万色を表現可能となっている。

【 0 0 3 9 】

コンピュータ本体 1 2 には、外部補助記憶装置としてのフロッピーディスクドライブ 1 3 a とハードディスク 1 3 b と C D - R O M ドライブ 1 3 c とが接続されており、ハードディスク 1 3 b にはシステム関連の主要プログラムが記録されており、フロッピーディスクや C D - R O M などから適宜必要なプログラムなどを読み込み可能となっている。

また、コンピュータ本体 1 2 を外部のネットワークなどに接続するための通信デバイスとしてモデム 1 4 a が接続されており、外部のネットワークに同公衆通信回線を介して接続し、ソフトウェアやデータをダウンロードして導入可能となっている。この例ではモデム 1 4 a にて電話回線を介して外部にアクセスするようにしているが、L A N アダプタを介してネットワークに対してアクセスする構成とすることも可能である。この他、コンピュータ本体 1 2 の操作用にキーボード 1 5 a やマウス 1 5 b も接続されている。

【 0 0 4 0 】

10

20

30

40

50

さらに、画像出力デバイスとして、ディスプレイ 17 a とカラープリンタ 17 b とを備えている。ディスプレイ 17 a については水平方向に 800 画素と垂直方向に 600 画素の表示エリアを備えており、各画素毎に上述した 1670 万色の表示が可能となっている。むろん、この解像度は一例に過ぎず、640 × 480 画素であったり、1024 × 768 画素であるなど、適宜、変更可能である。

【0041】

また、カラープリンタ 17 b はインクジェットプリンタであり、CMYK の四色の色インクを用いて記録媒体たる印刷用紙上にドットを付して画像を印刷可能となっている。画像密度は 360 × 360 DPI や 720 × 720 DPI といった高密度印刷が可能となっているが、階調表現については色インクを付すか否かといった 2 階調表現となっている。

一方、このような画像入力デバイスを使用して画像を入力しつつ、画像出力デバイスに表示あるいは出力するため、コンピュータ本体 12 内では所定のプログラムが実行されることになる。そのうち、基本プログラムとして稼働しているのはオペレーティングシステム (OS) 12 a であり、このオペレーティングシステム 12 a にはディスプレイ 17 a での表示を行わせるディスプレイドライバ (DSP DRV) 12 b とカラープリンタ 17 b に印刷出力を行わせるプリンタドライバ (PRT DRV) 12 c が組み込まれている。これらのドライバ 12 b, 12 c の類はディスプレイ 17 a やカラープリンタ 17 b の機種に依存しており、それぞれの機種に応じてオペレーティングシステム 12 a に対して追加変更可能である。また、機種に依存して標準処理以上の付加機能を実現することもできるようになってきている。すなわち、オペレーティングシステム 12 a という標準システム上で共通化した処理体系を維持しつつ、許容される範囲内での各種の追加的処理を実現できる。

【0042】

むろん、このようなプログラムを実行する前提として、コンピュータ本体 12 内には CPU 12 e と RAM 12 f と ROM 12 g と I/O 12 h などが備えられており、演算処理を実行する CPU 12 e が RAM 12 f を一時的なワークエリアや設定記憶領域として使用したりプログラム領域として使用しながら、ROM 12 g に書き込まれた基本プログラムを適宜実行し、I/O 12 h を介して接続されている外部機器及び内部機器などを制御している。

【0043】

ここで、基本プログラムとしてのオペレーティングシステム 12 a 上でアプリケーション 12 d が実行される。アプリケーション 12 d の処理内容は様々であり、操作デバイスとしてのキーボード 15 a やマウス 15 b の操作を監視し、操作された場合には各種の外部機器を適切に制御して対応する演算処理などを実行し、さらには、処理結果をディスプレイ 17 a に表示したり、カラープリンタ 17 b に出力したりすることになる。

【0044】

かかるコンピュータシステム 10 では、画像入力デバイスであるスキャナ 11 a などで画像データを取得し、アプリケーション 12 d による所定の画像処理を実行した後、画像出力デバイスとしてのディスプレイ 17 a やカラープリンタ 17 b に表示出力することが可能である。この場合、単に画素同士の対応に着目すると、カラープリンタ 17 b における画素密度とスキャナ 11 a の画素密度が一致する場合にはスキャンした元画像の大きさと印刷される画像の大きさとが一致するが、両者にずれがあれば画像の大きさが異なることになる。スキャナ 11 a の場合はカラープリンタ 17 b の画素密度と近似するものも多いが、高画質化のために画素密度の向上が図られているカラープリンタ 17 b の画素密度の方が一般的な画像入力デバイスにおける画素密度よりも高密度であることが多い。特に、ディスプレイ 17 a の表示密度と比較すると各段に高密度であり、ディスプレイ 17 a 上での表示を画素単位で一致させて印刷させるとなると極めて小さな画像になりかねない。

【0045】

このため、オペレーティングシステム 12 a で基準となる画素密度を決定しつつ実際のデバイスごとの画素密度の相違を解消するために解像度変換が実施される。例えば、ディス

10

20

30

40

50

プレイ 17 a の解像度が 720 DPI であるとするときに、オペレーティングシステム 12 a で 360 DPI を基準とするならば、ディスプレイドライバ 12 b が両者の間の解像度変換を実施する。また、同様の状況でカラープリンタ 17 b の解像度が 720 DPI であればプリンタドライバ 12 c が解像度変換を実施する。

【0046】

解像度変換は画像データにおける構成画素数を増やす処理にあたるので補間処理に該当し、これらのディスプレイドライバ 12 b やプリンタドライバ 12 c がその機能の一つとして補間処理を実施する。ここにおいて、ディスプレイドライバ 12 b やプリンタドライバ 12 c は上述した画素補間手段 C2 はもとより、以下に述べるように特徴量取得手段 C3 や補間処理選択手段 C4 を実行し、解像度変換にて画質が劣化しないようにしている。

10

【0047】

なお、かかるディスプレイドライバ 12 b やプリンタドライバ 12 c は、ハードディスク 13 b に記憶されており、起動時にコンピュータ本体 12 にて読み込まれて稼働する。また、導入時には CD-ROM であるとかフロッピーディスクなどの媒体に記録されてインストールされる。従って、これらの媒体は画像データ補間プログラムを記録した媒体を構成する。

本実施形態においては、画像データ補間装置をコンピュータシステム 10 として実現しているが、必ずしもかかるコンピュータシステムを必要とするわけではなく、同様の画像データに対して補間処理が必要なシステムであればよい。例えば、図 3 に示すようにデジタルスチルカメラ 11 b 1 内に補間処理する画像データ補間装置を組み込み、補間処理した画像データを用いてディスプレイ 17 a 1 に表示させたりカラープリンタ 17 b 1 に印字させるようなシステムであっても良い。また、図 4 に示すように、コンピュータシステムを介することなく画像データを入力して印刷するカラープリンタ 17 b 2 においては、スキャナ 11 a 2 やデジタルスチルカメラ 11 b 2 あるいはモデム 14 a 2 等を介して入力される画像データについて自動的に解像度変換を行って印刷処理するように構成することも可能である。

20

【0048】

この他、図 5 に示すようなカラーファクシミリ装置 18 a や図 6 に示すようなカラーコピー装置 18 b といった画像データを扱う各種の装置においても当然に適用可能である。図 7 および図 8 は、上述したプリンタドライバ 12 c が実行する解像度変換に関連するソフトウェアフローを示している。ここで、前者は汎用的なフローを示しており、後者は本実施形態の具体的なフローを示している。

30

【0049】

ステップ ST102 は元画像データを取得する。アプリケーション 12 d にてスキャナ 11 a から画像を読み込み、所定の画像処理を行った後で印刷処理すると、所定の解像度の印刷データはオペレーティングシステム 12 a を介してプリンタドライバ 12 c が取得するので、この段階が該当する。むしろ、スキャナ 11 a にて画像を読み込むものであってもよい。この処理はソフトウェアとしてみるときに画像データ取得ステップということになるが、当該画像データ取得ステップを含めてコンピュータに実行させる各種のステップは、オペレーティングシステム 12 a 自体やハードウェアを直接に含めないものとして理解することができる。これに対して、CPU などのハードウェアと有機一体的に結合したものと考えると画像データ取得手段 C1 に該当する。

40

【0050】

ステップ ST104 は、読み込んだ画像データにおける特徴量を抽出する処理である。特徴量の抽出処理の詳細は後述する。ステップ ST106 では得られた特徴量に基づいて当該画像データに最適な補間処理を選択し、同補間処理を表すフラグをセットする。そして、ステップ ST108 にて同フラグを参照してステップ ST110, ST112, ST114 の各補間処理 1 ~ N を実行させることになる。従って、ステップ ST110, ST112, ST114 に示す各補間処理 1 ~ N が具体的に画像補間ステップに相当するし、ステップ ST106, ST108 が特徴量に基づいて補間処理を選択するので補間処理選択

50

ステップに相当する。むしろ、これらがCPUなどのハードウェアと有機一体的に結合したものと考えると画像補間手段C2や補間処理選択手段C4を構成することになる。

【0051】

ステップST116, ST118は補間処理が全てのブロックについて終了するまでブロック移動させる処理を示している。補間処理は必ずしも画像全体にわたって一律の処理である必要はなく、ブロック単位の領域毎に補間処理を変更することも可能である。

従って、ブロック毎に最適な補間処理を実施しようとする場合には、各ブロックでの補間処理を終了したらステップST104にて次のブロックの特徴量を抽出する処理から実行し直す。一方、画像データ全体にわたって一律に処理する場合にはステップST108以下の処理を繰り返す。

10

【0052】

そして、全てのブロックが終了すればステップST120にて補間された画像データを出力する。プリンタドライバ12cの場合、解像度変換だけで印刷データが得られるわけではなく、色変換であるとか、ハーフトーン処理が必要になる。従って、ここで画像データを出力するとは、次の段階へのデータの受け渡しを意味する。

次に、以上のような汎用的なフローに対してより具体的な処理について説明する。本実施形態においては、元画像がコンピュータグラフィックス（非自然画）であるか写真（自然画）であるかを判定し、判定結果に基づいて補間処理を選択することにする。ステップST202ではステップST102と同様にして元画像データを入力する。

20

【0053】

元画像の種類がどのようなものであるかを判定する手法としてはいくつか採用可能であるが、本実施形態においては画像データを用いて集計処理を行う。

具体的には画像データで利用されている色数を求め、多ければ自然画と判定し、少なければ非自然画と判定する。写真の場合、一色の物体が映っているとしても光線の加減や陰影によって明るいところから暗いところまで幅ができてしまい、色数は多くなる。このような特性があるので色数を見れば自然画か非自然画かを判断できる。ただし、1670万色の中で実際にどれだけの色を利用しているかを集計するのはプログラムでの処理において効率的でない。また、自然画でさえそのうちの一部しか利用していないことが多く、非自然画との見極めがつきにくい。

30

【0054】

このため、本実施形態においては、各画素の輝度を求め、輝度が取りうる範囲において画素数のヒストグラムを集計して利用色数の傾向を判定する。1670万色のうちで同じ輝度となる色が複数存在するのは当然であるが、非自然画との比較だけに着目すれば、色であっても輝度であっても多いか少ないかの比較は可能である。さらに、非自然画がせいぜい64色程度ぐらいしか利用していないと考えれば輝度の取りうる範囲が256階調であったとしても十分に判定可能であると考えられる。

【0055】

一方、輝度の集計は以上のように画像データの概略の傾向を判定するのみであるから、必ずしも全画素について集計する必要はない。すなわち、集計対象となる画素を選択すべく間引き処理を実行する。

40

図9に示すように、ビットマップの画像であれば、縦方向に所定ドットと横方向に所定ドットからなる二次元のドットマトリクスとして成り立っており、正確な輝度の分布を求めらるるのであれば全画素について輝度を調べる必要がある。しかしながら、ここでは必ずしも正確である必要はない。従って、ある誤差の範囲内となる程度に間引きを行うことが可能である。統計的誤差によれば、サンプル数Nに対する誤差は概ね $1 / (2^{**} (1/2))$ と表せる。ただし、**は累乗を表している。従って、1%程度の誤差で処理を行うためには $N = 10000$ となる。

【0056】

ここにおいて、図9に示すビットマップ画面は(w i d t h) × (h e i g h t)の画素数となり、サンプリング周期r a t i oは、

50

$ratio = \min(width, height) / A + 1$

とする。ここにおいて、 $\min(width, height)$ は $width$ と $height$ のいずれか小さい方であり、 A は定数とする。また、ここでいうサンプリング周期 $ratio$ は何画素ごとにサンプリングするかを表しており、図10の印の画素はサンプリング周期 $ratio = 2$ の場合を示している。すなわち、縦方向及び横方向に二画素ごとに一画素のサンプリングであり、一画素おきにサンプリングしている。 $A = 200$ としたときの1ライン中のサンプリング画素数は図11に示すようになる。

【0057】

同図から明らかなように、サンプリングしない場合のサンプリング周期 $ratio = 1$ の場合を除いて、200画素以上の幅があるときには最低でもサンプル数は100画素以上となることが分かる。従って、縦方向と横方向について200画素以上の場合には(100画素) × (100画素) = (10000画素)が確保され、誤差を1%以下にできる。ここにおいて $\min(width, height)$ を基準としているのは次のような理由による。例えば、図12(a)に示すビットマップ画像のように、 $width \gg height$ であるとする、長い方の $width$ でサンプリング周期 $ratio$ を決めてしまった場合には、同図(b)に示すように、縦方向には上端と下端の2ラインしか画素を抽出されないといったことが起こりかねない。しかしながら、 $\min(width, height)$ として、小さい方に基づいてサンプリング周期 $ratio$ を決めるようにすれば同図(c)に示すように少ない方の縦方向においても中間部を含むような間引きを行うことができるようになる。

【0058】

なお、この例では、縦方向と横方向の画素について正確なサンプリング周期で間引きを行うようにしている。これは、逐次入力される画素について間引きしながら処理する場合に好適である。しかし、全画素が入力されている場合には縦方向や横方向についてランダムに座標を指定して画素を選択するようにしても良い。このようにすれば、10000画素というような必要最低限の画素数が決まっている場合に10000画素となるまでランダムに抽出する処理を繰り返し、10000画素となった時点で抽出を止めればよくなる。

【0059】

このように選択した画素についての画素データがその成分要素として輝度を持っていればその輝度値を用いて分布を求めることが可能である。しかしながら、輝度値が直接の成分値となっていない画像データの場合でも、間接的には輝度を表す成分値を備えている。従って、輝度値が直接の成分値となっていない表色空間から輝度値が直接の成分値となっている表色空間への変換を行えば輝度値を得ることができる。

【0060】

異なる表色空間の間での色変換は変換式によって一義的に定まるものではなく、それぞれの成分値を座標とする色空間について相互に対応関係を求めておき、この対応関係を記憶した色変換テーブルを参照して逐次変換する必要がある。すると、厳密には1670万個の要素の色変換テーブルを持たなければならない。効率的な記憶資源の利用を考えた結果、すべての座標値についての対応関係を用意しておくのではなく、通常は適当なとびとびの格子点について対応関係を用意しておき、補間演算を併用するようにしている。しかし、かかる補間演算はいくつかの乗算や加算を経て可能となるものであるため、演算処理量は膨大となってくる。

【0061】

すなわち、フルサイズの色変換テーブルを使用するのであれば処理量としては少なくなるもののテーブルサイズが非現実的な問題となり、テーブルサイズを現実的なサイズにすれば演算処理量が非現実的となることが多い。

このような状況に鑑み、本実施形態においては、テレビジョンなどの場合に利用されているように、RGBの三原色から輝度を求める次式の変換式を採用している。すなわち、P点での輝度 y_p についてはRGBの成分値(R_p, G_p, B_p)から、

$$y_p = 0.30 R_p + 0.59 G_p + 0.11 B_p$$

10

20

30

40

50

とする。このようにすれば、三回の乗算と二回の加算だけで輝度値を求めることができるようになる。

【 0 0 6 2 】

本実施形態においては、R G Bの表色空間を対象としている結果、このような変換式を採用しているが、その背景には各成分値が色の明るさを示しているので、それぞれの成分値を単独で見た場合に輝度に線形に対応しているという性質がある。従って、よりおおざっぱに言えばそれぞれの加算割合を考慮することなく単に

$$y_p = (R_p + G_p + B_p) / 3$$

というように簡略化することも不可能ではない。

【 0 0 6 3 】

ステップ S T 2 0 4 では、以上のようにした間引き処理した画素についての輝度をヒストグラム化する。集計後、ステップ S T 2 0 6 では色数を計数する。色数が少なれば輝度の分布もまばらであり、ビジネスグラフのような非自然画では図 1 3 に示すように線スペクトル状に表れるし、写真のような自然画であれば図 1 4 に示すようになだらかな曲線状になることが予想される。

このようなことから、ステップ S T 2 0 6 では 2 5 6 階調の輝度のうち分布数が「 0 」でない輝度値がいくつ表れているかカウントし、ステップ S T 2 0 8 では「 6 4 」色未満であるときに自然画でないと判断し、「 6 4 」色以上であるときに自然画であると判断する。一方、分布が線スペクトル状か否かは分布数が「 0 」でない輝度値の隣接割合で判断することも可能である。すなわち、分布数が「 0 」でない輝度値であって隣接する輝度値に分布数があるか否かを判断する。隣接する二つの輝度値のうち少なくとも一方で隣接していれば何もせず、両方で隣接していない場合にカウントを行い、その結果、「 0 」でない輝度値の数とカウント値との割合で判断すればよい。例えば、「 0 」でない輝度値の数が「 8 0 」であって、隣接しないものの数が「 8 0 」であれば線スペクトル状に分布していることが分かる。むろん、この利用色数が特徴量に該当する。

【 0 0 6 4 】

なお、特徴量の取得手法はこれらに限るものではなく他の手法も実現可能である。

まず、集計対象となる画素を間引きするにあたっては、上述したような均等間隔の間引きに限られるものではない。例えば、画像の中での本来のオブジェクト部分を見つけ、その画素についての特徴量を集計するといったことも可能である。

このようなオブジェクトは他の部分と比較して画像がシャープであるという経験的事実に基づいて、シャープな部分がオブジェクトの画素であると判断する。画像データがドットマトリクス状の画素から構成されている場合には、画像のエッジ部分では隣接する画素間での同データの差分は大きくなる。この差分は輝度勾配であり、これをエッジ度と呼ぶことにして各画素でのエッジ度を判定する。図 1 5 に示すような X Y 直交座標を考察する場合、画像の変化度合いのベクトルは X 軸方向成分と Y 軸方向成分とをそれぞれ求めれば演算可能となる。ドットマトリクス状の画素からなるデジタル画像においては、図 1 6 に示すように縦軸方向と横軸方向に画素が隣接しており、X 方向の差分値 f_x と Y 方向の差分値 f_y は、

$$f_x = f(x + 1, y) - f(x, y)$$

$$f_y = f(x, y + 1) - f(x, y)$$

のように表される。従って、これらを成分とするベクトルの大きさ $|g(x, y)|$ は、

$$|g(x, y)| = (f_x^2 + f_y^2)^{1/2}$$

のように表される。むろん、エッジ度はこの $|g(x, y)|$ で表される。なお、本来、画素は図 1 7 に示すように縦横に昇目状に配置されており、中央の画素に注目すると八つの隣接画素がある。従って、同様にそれぞれの隣接する画素との画像データの差分をベクトルで表し、このベクトルの和を画像の変化度合いと判断しても良い。

【 0 0 6 5 】

以上のようにして各画素についてエッジ度が求められるので、基本的にはあるしきい値と比較してエッジ度の方が大きい画素をオブジェクトの画素と判断すればよい。しかしなが

10

20

30

40

50

ら、経験的事実から考察すると、オブジェクトは構図の中央部分に位置することが多い。この事実は、中央部分から多くの画素が抽出されるような仕組みとすることによってオブジェクトの画素がより抽出されやすくなるということを裏付けるものである。

【 0 0 6 6 】

このため、図 1 8 に示すように、画像の中の部分毎に比較するしきい値 $T h 1$, $T h 2$, $T h 3$ を異ならせておくことも可能である。むろん、この例では、

$$T h 1 < T h 2 < T h 3$$

なる関係があり、中央に近い部分ほどしきい値は低く、エッジ度が比較的低くてもオブジェクトと判断されるようになっている。

むろん、このようにしてオブジェクトと判断された画素についての画像データを集計し、補間処理に対応した特徴量を得ることになる。

10

【 0 0 6 7 】

一方、特徴量は必ずしも画像データを集計して得る必要もない。補間処理に依存して補間結果が良好となるか否かを関連づけるものであればよい。

画像データの画像が自然画か否かは印刷元となる画像ファイルの形式からも判断できる。図 1 9 はプリンタドライバ 1 2 c がオペレーティングシステム 1 2 a に用意されたシステム関数を利用する状況を示しており、プリンタドライバ 1 2 c がファイル名を問合せる関数を利用すると、オペレーティングシステム 1 2 a は対応するファイル名を返答する。この場合、" X X X X . X L S " であればその拡張子からビジネスグラフであると分かり、非自然画であるとの判定を可能とする。また、" X X X X . J P G " であればその拡張子

20

から写真画像の圧縮ファイルであると分かり、自然画であるとの判定を可能とする。むろん、拡張子からではなく、データファイルの先頭部分に含まれる情報からドロー系のファイル構造であるかビットマップ系のファイル構造であるかも判断できるから、自ずから自然画であるか非自然画であるかといった目安を取得できることになる。すなわち、このような画像の内容を推理可能となる目安であれば、特徴量を構成することになる。

【 0 0 6 8 】

以上のようにして、ステップ S T 2 0 2 にて入力された元画像データが自然画であるか非自然画であるかを判断したら、それぞれに応じた適切な補間処理を実行する。

ここで、本実施形態において実行する補間処理の各手法について説明する。

コンピュータグラフィックスのような非自然画に適した補間処理として、ステップ S T 2 1 0 ではニアリスト法の補間処理を実行可能となっている。ニアリスト法は図 2 0 に示すように、周囲の四つの格子点 $P i j$, $P i + 1 j$, $P i j + 1$, $P i + 1 j + 1$ と内挿したい点 $P u v$ との距離を求め、もっとも近い格子点のデータをそのまま移行させる。これを一般式で表すと、

30

$$P u v = P i j$$

ここで、 $i = [u + 0 . 5]$ 、 $j = [v + 0 . 5]$ である。なお、 $[]$ はガウス記号で整数部分を取ることを示している。

【 0 0 6 9 】

図 2 1 は、ニアリスト法で画素数を縦横 3 倍ずつに補間する状況を示している。補間する前には四隅の画素 () があるとして、補間して生成する画素にはこれらの画素のうちもっとも近い画素のデータをそのまま移行させている。すなわち、この例で言えば四隅の画素に隣接する画素についてそれぞれ複写することになる。また、かかる処理を行うと、図 2 2 に示すように白い画素を背景として黒い画素が斜めに配置される元画像は、図 2 3 に示すように黒の画素が縦横に 3 倍に拡大されつつ斜め方向に配置されることになる。

40

【 0 0 7 0 】

ニアリスト法においては、画像のエッジがそのまま保持される特徴を有する。それ故に拡大すればジャギーが目立つもののエッジはエッジとして保持される。これに対して他の補間処理では補間される画素を周りの画素のデータを利用してなだらかに変化するようにする。従って、ジャギーが目立たなくなる反面、本来の元画像の情報は削られていってしま

50

い、エッジがなくなることになってコンピュータグラフィックスなどの非自然画には適さなくなる。

【 0 0 7 1 】

一方、写真のような自然画に適した補間処理として、ステップ S T 2 1 2 ではキュービック法の補間処理を実行する。キュービック法は図 2 4 に示すように、内挿したい点 P u v を取り囲む四つの格子点のみならず、その一周り外周の格子点を含む計 1 6 の格子点のデータを利用する。

内挿点 P u v を取り囲む計 1 6 の格子点がそれぞれに値を備えている場合に、内挿点 P u v はそれらの影響を受けて決定される。例えば、一次式で補間しようとするれば、内挿点を挟む二つの格子点からの距離に反比例させて重みづけ加算すればよい。X 軸方向に注目すると、内挿点 P u v から上記 1 6 の格子点との距離は、図面上、左外側の格子点までの距離を x 1、左内側の格子点までの距離を x 2、右内側の格子点までの距離 x 3、右外側の格子点までの距離 x 4 と表しつつ、このような距離に対応した影響度合いを関数 f (x) で表すことにする。また、Y 軸方向に注目すると、内挿点 P u v から上記 1 6 の格子点との距離は、上方外側の格子点までの距離を y 1、上方内側の格子点までの距離を y 2、下方内側の格子点までの距離 y 3、下方外側の格子点までの距離 y 4 と表しつつ、同様に影響度合いは関数 f (y) で表せる。

10

【 0 0 7 2 】

1 6 の格子点は以上のような距離に応じた影響度合いで内挿点 P u v に寄与するので、全ての格子点にデータに対して X 軸方向と Y 軸方向のそれぞれの影響度合いを累積させる一般式は次式のようなになる。

20

【 0 0 7 3 】

【 数 1 】

$$P=[f(y1)f(y2)f(y3)f(y4)] \begin{pmatrix} P11 & P21 & P31 & P41 \\ P12 & P22 & P32 & P42 \\ P13 & P23 & P33 & P43 \\ P14 & P24 & P34 & P44 \end{pmatrix} \begin{pmatrix} f(x1) \\ f(x2) \\ f(x3) \\ f(x4) \end{pmatrix}$$

30

また、ここで距離に応じた影響度合いを 3 次たたま込み関数で表すとすると、 f (t) = { sin(t) } / t

となる。なお、上述した各距離 x 1 ~ x 4 , y 1 ~ y 4 は格子点 P u v の座標値 (u , v) について絶対値を利用して次のように算出することになる。

x1 = 1+(u-|u|) y1 = 1+(v-|v|)

x2 = (u-|u|) y2 = (v-|v|)

x3 = 1-(u-|u|) y3 = 1-(v-|v|)

x4 = 2-(u-|u|) y4 = 2-(v-|v|)

40

以上の前提のもとで P について展開すると、

【 0 0 7 4 】

【 数 2 】

$$P=[f(y_1) f(y_2) f(y_3) f(y_4)] \begin{Bmatrix} P_{11} \cdot f(x_1) + P_{21} \cdot f(x_2) + P_{31} \cdot f(x_3) + P_{41} \cdot f(x_4) \\ P_{12} \cdot f(x_1) + P_{22} \cdot f(x_2) + P_{32} \cdot f(x_3) + P_{42} \cdot f(x_4) \\ P_{13} \cdot f(x_1) + P_{23} \cdot f(x_2) + P_{33} \cdot f(x_3) + P_{43} \cdot f(x_4) \\ P_{14} \cdot f(x_1) + P_{24} \cdot f(x_2) + P_{34} \cdot f(x_3) + P_{44} \cdot f(x_4) \end{Bmatrix}$$

$$\begin{aligned} &= f(y_1) \{P_{11} \cdot f(x_1) + P_{21} \cdot f(x_2) + P_{31} \cdot f(x_3) + P_{41} \cdot f(x_4)\} \\ &+ f(y_2) \{P_{12} \cdot f(x_1) + P_{22} \cdot f(x_2) + P_{32} \cdot f(x_3) + P_{42} \cdot f(x_4)\} \\ &+ f(y_3) \{P_{13} \cdot f(x_1) + P_{23} \cdot f(x_2) + P_{33} \cdot f(x_3) + P_{43} \cdot f(x_4)\} \\ &+ f(y_4) \{P_{14} \cdot f(x_1) + P_{24} \cdot f(x_2) + P_{34} \cdot f(x_3) + P_{44} \cdot f(x_4)\} \end{aligned}$$

10

$$\begin{aligned} &= P_{11} \cdot f(x_1) \cdot f(y_1) + P_{21} \cdot f(x_2) \cdot f(y_1) + P_{31} \cdot f(x_3) \cdot f(y_1) + P_{41} \cdot f(x_4) \cdot f(y_1) \\ &+ P_{12} \cdot f(x_1) \cdot f(y_2) + P_{22} \cdot f(x_2) \cdot f(y_2) + P_{32} \cdot f(x_3) \cdot f(y_2) + P_{42} \cdot f(x_4) \cdot f(y_2) \\ &+ P_{13} \cdot f(x_1) \cdot f(y_3) + P_{23} \cdot f(x_2) \cdot f(y_3) + P_{33} \cdot f(x_3) \cdot f(y_3) + P_{43} \cdot f(x_4) \cdot f(y_3) \\ &+ P_{14} \cdot f(x_1) \cdot f(y_4) + P_{24} \cdot f(x_2) \cdot f(y_4) + P_{34} \cdot f(x_3) \cdot f(y_4) + P_{44} \cdot f(x_4) \cdot f(y_4) \end{aligned}$$

20

となる。なお、3次たたみ込み関数と呼ばれるように距離に応じた影響度合い $f(t)$ は次のような三次式で近似される。

【数3】

$$f(t) = \begin{cases} \sin(\pi t) / \pi t \\ \begin{cases} 1 - 2|t|^2 + |t|^3 & : 0 \leq |t| < 1 \\ 4 - 8|t| + 5|t|^2 - |t|^3 & : 1 \leq |t| < 2 \\ 0 & : 2 \leq |t| \end{cases} \end{cases}$$

30

このキュービック法では一方の格子点から他方の格子点へと近づくにつれて徐々に変化していき、その変化具合がいわゆる3次関数的になるという特徴を有している。

【0075】

図25と図26はキュービック法にて補間される際の実例を示している。理解を容易にするため、垂直方向についてのデータの変化はなく、水平方向についてエッジが生じているモデルについて説明する。また、補間する画素を3点とする。

40

まず、図26の具体的数値について説明する。補間前の画素の階調値を左列に「Original」として示しており、階調値「64」の画素（P0、P1、P2、P3）が4点並び、階調値「128」の画素（P4）を1点挟み、階調値「192」の画素（P5、P6、P7、P8、P9）が5点並んでいる。この場合、エッジは階調値「128」の画素の部分である。

【0076】

ここで各画素間に3点の画素（Pn1、Pn2、Pn3）を内挿することになると、内挿される画素間の距離は「0.25」となり、上述したx1～x4は内挿点毎に表の中程の列の数値となる。x1～x4に対応して $f(x_1) \sim f(x_4)$ も一義的に計算されるこ

50

とになり、例えば、 x_1, x_2, x_3, x_4 が、それぞれ「1.25」、「0.25」、「0.75」、「1.75」となる場合、それに対する $f(t)$ については、概略「-0.14」、「0.89」、「0.30」、「-0.05」となる。また、 x_1, x_2, x_3, x_4 が、それぞれ「1.50」、「0.50」、「0.50」、「1.50」となる場合、それに対する $f(t)$ については、「-0.125」、「0.625」、「0.625」、「-0.125」となる。また、 x_1, x_2, x_3, x_4 が、それぞれ「1.75」、「0.75」、「0.25」、「1.25」となる場合、それに対する $f(t)$ については、概略「-0.05」、「0.30」、「0.89」、「-0.14」となる。以上の結果を用いて内挿点の階調値を演算した結果を表の右列に示しているとともに、図25においてグラフで示している。なお、このグラフの意味するところについて後に詳述する。

10

【0077】

垂直方向についてのデータの変化がないものとみなすと、演算は簡略化され、水平方向に並ぶ四つの格子点のデータ (P_1, P_2, P_3, P_4) だけを参照しつつ、内挿点から各格子点までの距離に応じた影響度合い $f(t)$ を利用して次のように算出できる。

$$P = P_1 \cdot f(x_1) + P_2 \cdot f(x_2) + P_3 \cdot f(x_3) + P_4 \cdot f(x_4)$$

従って、内挿点 P_{21} について算出する場合には、

$$\begin{aligned} P_{21} &= 64 * f(1.25) + 64 * f(0.25) + 64 * f(0.75) + 128 * f(1.75) \\ &= 64 * (-0.14063) + 64 * (0.890625) + 64 * (0.296875) + 128 * (-0.04688) \\ &= 61 \end{aligned}$$

20

となる。

【0078】

キュービック法によれば3次関数的に表せる以上、そのカーブの形状を調整することによって補間結果の品質を左右することができる。

その調整の一例として、

$$0 < t < 0.5 \quad f(t) = -(8/7)t^3 - (4/7)t^2 + 1$$

$$0.5 < t < 1 \quad f(t) = (1-t)(10/7)$$

$$1 < t < 1.5 \quad f(t) = (8/7)(t-1)^3 + (4/7)(t-1)^2 - (t-1)$$

$$1.5 < t < 2 \quad f(t) = (3/7)(t-2)$$

30

としたものをハイブリッドバイキュービック法と呼ぶことにする。

【0079】

図27はハイブリッドバイキュービック法にて補間される際の具体例を示しており、キュービック法の場合と同じ仮定のモデルについて補間した結果を示している。また、図25にもハイブリッドバイキュービック法による補間処理結果を示しており、この例では3次関数的なカーブがわずかに急峻となり、画像全体のイメージがシャープとなる。

上述したニアリスト法やキュービック法やハイブリッドバイキュービック法の特性の理解のために他の補間手法である共1次内挿法(バイリニア補間:以下、バイリニア法と呼ぶ)について説明する。

40

【0080】

バイリニア法は、図28に示すように、一方の格子点から他方の格子点へと近づくにつれて徐々に変化していく点でキュービック法に近いが、その変化が両側の格子点のデータだけに依存する一次関数的である点で異なる。すなわち、内挿したい点 P_{uv} を取り囲む四つの格子点 $P_{ij}, P_{i+1j}, P_{ij+1}, P_{i+1j+1}$ で区画される領域を当該内挿点 P_{uv} で四つの区画に分割し、その面積比で対角位置のデータに重み付けする。これを式で表すと、

$$\begin{aligned}
 P = & \{ (i+1) - u \} \{ (j+1) - v \} P_{i j} \\
 & + \{ (i+1) - u \} \{ v - j \} P_{i j+1} \\
 & + \{ u - i \} \{ (j+1) - v \} P_{i+1 j} \\
 & + \{ u - i \} \{ v - j \} P_{i+1 j+1}
 \end{aligned}$$

となる。なお、 $i = [u]$ 、 $j = [v]$ である。

【0081】

二つのキュービック法とバイリニア法は一方の格子点から他方の格子点へと近づくにつれて徐々に変化していく点で共通するが、その変化状況が3次関数的であるか1次関数的であるかが異なり、画像としてみたときの差異は大きい。図29はニアリスト法とキュービック法とハイブリッドバイキュービック法とバイリニア法における補間結果の相違を理解しやすくするために二次元的に表した図である。同図において、横軸に位置を示し、縦軸に補間関数を示している。むろん、この補間関数は上述した距離に応じた影響度合いに該当する。 $t = 0$ 、 $t = 1$ 、 $t = 2$ の位置に格子点が存在し、内挿点は $t = 0 \sim 1$ の位置となる。

10

【0082】

バイリニア法の場合、隣接する二点間 ($t = 0 \sim 1$) で直線的に変化するだけであるので境界をスムージングすることになり、画面の印象はぼやけてしまう。すなわち、角部のスムージングと異なり、境界がスムージングされると、コンピュータグラフィックスでは、本来あるべき輪郭がなくなってしまうし、写真においてはピントが甘くなってしまう。一方、キュービックにおいては、隣接する二点間 ($t = 0 \sim 1$) においては山形の凸を描いて徐々に近接するのみならず、さらに同二点間の外側 ($t = 1 \sim 2$) において下方に押し下げる効果をもつ。すなわち、あるエッジ部分は段差が生じない程度に大きな高低差を有するように変化され、写真においてはシャープさを増しつつ段差が生じないという好適な影響を及ぼす。また、ハイブリッドバイキュービックではよりシャープさを増す影響を及ぼす。なお、キュービック法は演算処理量が大きく、補間倍率が大きくなって補間すべき画素数が大きくなれば多大な演算処理量を要することになる。

20

【0083】

画質の面を重視すれば、キュービック法のような三次関数を選びそうであるが、コンピュータの処理では速度と画質のバランスも大きい。すなわち、画質の向上程度に応じて処理速度の低下具合の許容度が大きくなるが、画質の向上が微量あるいは多少画質が向上落ちるとしても処理速度が高速である方を好むという場合もある。一方、以上のような補間関数の比較とともに具体的な数値を示す図25、図26、図27を参照するとより理解しやすい。図15の例を参照し、もともとのエッジ部分である階調値「64」の画素 (P3) と、階調値「128」の画素 (P4) と、階調値「192」の画素 (P5) という三点に注目してみると、単純に直線的に連結する手法はバイリニア法に相当し、これに対してキュービック法では具体的なS字カーブが形成されているし、ハイブリッドバイキュービック法ではそのS字カーブがより急峻となっている。むろん、S字カーブの方向は画素の階調値変化を急峻とする方向であり、だからこそエッジが強調されている。また、このエッジ画素に隣接する領域 (P2 ~ P3、P5 ~ P6) ではいわゆるアンダーシュートとオーバーシュートが生じており、低い側に生じるアンダーシュートと高い側に生じるオーバーシュートにより、エッジ画素を挟む両側の高低差が大きくなる。従って、これらの二つの要因によってエッジが強調されることが理解できる。

30

40

【0084】

画像がシャープに見えるか否かはこのS字カーブにおける中央部分の傾斜角度が影響を与えることも容易に理解できる。また、エッジの両側のアンダーシュートとオーバーシュートによって生じる高低差も同様に影響を与えるものといえる。

各補間処理には以上のような特性の違いがあり、ステップST208ではステップST206にて求めた色数に基づいて非自然画であると判断する場合にはステップST210に

50

おけるニアリスト法による補間処理を実行するし、自然画であればキュービック法による補間処理を実行することになる。

【 0 0 8 5 】

補間処理自体は任意の倍率で実行可能であるが、プリンタドライバ 1 2 c における処理の高速化を図るため、整数倍の補間処理を受け付けるようにする。図 2 9 は水平方向と垂直方向に 2 倍に補間する処理例を示している。予め、補間後の画像データについての変数領域を確保すると、整数倍の補間処理であれば元画像の画像データは整数倍した座標値に対応する画素の画像データとなる。図に示す例で言えば、旧座標値 (0 , 0) は新座標値 (0 , 0) に対応し、旧座標値 (1 , 0) は新座標値 (2 , 0) に対応し、旧座標値 (0 , 1) は新座標値 (0 , 2) に対応し、旧座標値 (1 , 1) は新座標値 (2 , 2) に対応するということである。従って、残りの座標値についてのみ上述した補間処理に対応して画像データを生成していく。この場合、画像データの幅方向を主走査方向とし、長さ方向を副走査方向として順に走査していくことも可能であるし、画像データがある四つの格子点に囲まれた各ブロック毎に内部の座標値の補間処理をしていって埋めていくことも可能である。

10

【 0 0 8 6 】

そして、新たな座標値について全て補間処理したときにステップ S T 2 1 4 にて補間画像データを次段の処理へ引き渡す。ただし、補間倍率によっては補間画像データのデータ量が極めて多くなることもあるし、そもそもプリンタドライバ 1 2 c が利用可能なメモリ領域がさほど多くない場合もある。このような場合には一定のデータ量ごとに分けて出力

20

するようにしても構わない。
上述したようにステップ S T 1 1 6 , S T 1 1 8 は補間処理がブロック単位の領域毎に補間処理を変更する手法に対応している。これを上述したプリンタドライバ 1 2 c の具体的なソフトウェア処理に対応して説明すると、図 2 9 に示すように整数倍して格子点の画像データを新たな座標値に移行せしめておき、四つの格子点に囲まれた各ブロック毎に特徴量を取得して補間処理を選択することになる。上述した例では利用色数を集計するものであるのかかる特徴量を利用することはできないが、他の特徴量を取得して補間処理を変更していくことが可能である。

【 0 0 8 7 】

例えば、四つの格子点の差を求めると、自然画であれば当然にこの差が生じていることが多い。しかし、自然画に重ねて文字が組み込まれているような場合、文字が単一色であれば四つの格子点は一致し、差が生じない。むろん、差が生じない領域に内挿する画素は四つの格子点と同じデータとすればよいのであり、最も演算量の少ないニアリスト法で補間すればよい。

30

また、差が生じない場合に限る必要はない。空のように余り変化のない領域ではニアリストを使用しても画質の劣化は判断できないことが多いため、四つの格子点の差が小さいときにはニアリスト法で補間するようにしても良い。なお、以上の例では領域として最小の単位でブロックを形成しているが、より大きなブロックで補間処理を変更するようにしても構わない。

【 0 0 8 8 】

選択可能な補間処理はニアリスト法かキュービック法かのいずれか一方だけに限る必要もない。例えば、4 倍に補間するような場合に、最初にキュービック法にて 2 倍に補間処理し、続いてニアリスト法にて 2 倍に補間処理することも有意義である。補間処理によって画素数が増加する前に演算量の多い補間処理を実行しておき、その後で演算量の少ない補間処理を実行することになるからである。

40

【 0 0 8 9 】

すなわち、補間結果に関連するものとして、このような場合には補間倍率が影響しているといえる。従って、プリンタドライバ 1 2 c はオペレーティングシステム 1 2 a における基準解像度とカラープリンタ 1 7 b の解像度との比較において倍率が判定できたときに補間処理を選択することも可能である。

50

このように、画像入力デバイスとしてスキャナ 11 aなどを有するとともに画像出力デバイスとしてカラープリンタ 17 bなどを有するコンピュータシステム 10において、プリンタドライバ 12 cはステップ S T 202にて元画像データを取得し、ステップ S T 204, S T 206にて画像の利用色数を計数することにより、当該画像データの画像が自然画であるのか非自然画であるのかを判定するための特徴量を抽出し、ステップ S T 208にて同特徴量に基づいて自然画か非自然画かを決定することによって非自然画であればステップ S T 210のニアリスト法の補間処理を実行するし、自然画であればステップ S T 212のキュービック法の補間処理を実行するようにしたため、画像の特徴に応じた補間処理が実施され、最適な補間結果を極めて容易に得ることができる。

【0090】

そして、このようなプログラムを実行する前提として、コンピュータ 12には、CPU 12 eとRAM 12 fとROM 12 gとI/O 12 hなどが備えられている。同RAM 12 fは、画像をドットマトリクス状の画素で表現した画像データを記憶する画像メモリとして作用する。また、同RAM 12 fは、この画像メモリに記憶された画像データを対象として補間処理に関連する特徴量を取得するとともに、同特徴量に対応して複数の補間処理の中から最適な補間結果を得ることが可能な補間処理を選択し、同補間処理を上記CPU 12 aに実行させて上記画像メモリに書き込ませる処理プログラムを記憶する。さらに、I/O 12 hは上記画像データを入力および出力するインターフェイスとして作用する。

【0091】

むろん、CPU 12 eはRAM 12 fを一時的なワークエリアや設定記憶領域やプログラム領域として使用しながら、ROM 12 gに書き込まれた基本プログラムを適宜実行し、I/O 12 hを介して未処理の画像データを入力し、処理後の画像データを出力する。次に、上述した画像データの変化度合いを特徴量とした実施形態について説明する。

図31は、本画像データ補間装置を表すブロック図である。

本画像データ補間装置はこのような画素単位での拡大処理を実施するものであり、画像データ取得手段 D 1は、このような画像データを取得し、画素補間手段 D 2はこの画像データにおける構成画素数を増やす補間処理を行う。ここで、画素補間手段 D 2は補間処理として画素の変化度合いに応じた複数の補間処理を実行可能となっており、画素変化度合評価手段 D 3が上記画像データに基づいて画素ごとの変化度合いを評価する。すると、補間処理選択手段 D 4はそうにして評価された画素の変化度合いに対応して最適な補間結果を得ることが可能な補間処理を選択し、上記画素補間手段 D 2に実行させる。

【0092】

なお、本実施形態においては、ディスプレイドライバ 12 bやプリンタドライバ 12 cは上述した画素補間手段 D 2はもとより、以下に述べるように画素変化度合評価手段 D 3や補間処理選択手段 D 4を実行し、解像度変換において最もバランスの良い補間結果を得ることができるようにしている。

図32は、上述したプリンタドライバ 12 cが実行する解像度変換に関連するソフトウェアフローを示している。

【0093】

ステップ S T 302は元画像データを取得する。ステップ S T 304～S T 308は、読み込んだ画像データにおける画素の変化度合いを評価する処理である。

上述したような簡略化した演算で輝度を求めることとした上で、図33と図34は、輝度勾配を算出するためのエッジ検出フィルタを示している。画像データはドットマトリクス状の画素から構成されているので、注目画素を中心とする近隣の八画素との間で画像の変化度合いを評価すべきである。そういった意味では図34(a)に示すように、注目画素に8倍の重み付けを与えつつ周囲の画素を均等に評価してそれを合算することでフィルタを掛けることが好ましい。しかしながら、経験的には必ずしも周囲の八画素を評価しなくても図33(a)に示すように注目画素と周囲の四画素だけから評価可能である。むろん、四画素を利用するか八画素を利用するかでは演算量の差が大きく、このようにして評価対象を少なくすると処理時間を減らすことができる。

10

20

30

40

50

【 0 0 9 4 】

また、図 3 3 (b) と図 3 4 (b) には実際の画像データ (輝度) の例を示しており図 3 3 (c) と図 3 4 (c) には (a) に示すフィルタを (b) に示す画像データの配置に適用した場合の演算例を示している。画像データは概ね左斜め上方側に画像データ「 1 0 0 」のエリアがあり、右斜め下方側に画像データ「 7 0 」と「 6 0 」の領域があるような場合を示している。図 3 3 の例では、中心画素の上下左右の四画素 (画像データ「 1 0 0 」、「 1 0 0 」、「 7 0 」、「 7 0 」) についてそれぞれ「 - 1 」の重みが付加され、中心画素 (画像データ「 1 0 0 」) には「 4 」の重みが付加されている。そして、この五画素について重み付加算を行なう。この重み付加算結果は「 6 0 」であり、しきい値 (t h) の「 3 2 」を越えている。

10

【 0 0 9 5 】

一方、図 3 4 の例では、中心画素を取り囲む八画素についてそれぞれ「 - 1 」の重みが付加され、中心画素には「 8 」の重みが付加されている。この重み付加算結果は「 1 0 0 」であり、しきい値 (t h) の「 6 4 」を越えている。

図 3 3 や図 3 4 に示すエッジ検出フィルタを利用した結果を各画素のエッジ量 E と呼ぶと、その分布は図 3 5 に示すように正規分布的となることが予想され、画像の変化度合いが大きいエッジ部分であるか否かはしきい値 t h と比較することによって判定できる。図 3 3 と図 3 4 に示すエッジ検出フィルタはそれぞれしきい値として t h = 3 2 および t h = 6 4 というエッジ量のしきい値が妥当する。従って、エッジの画素か否かは次式から評価する。

20

$$(E < - t h) \quad \text{or} \quad (t h > E)$$

この評価をドットマトリクス状の画素の全てに実施するのがステップ S T 3 0 6 の処理であり、各画素単位でエッジの画素のように画像の変化度合いが大きい画素であるか否かを評価する。

【 0 0 9 6 】

ところで、各画素単位で画像の変化度合いが大きいか否かを判定するとしても、補間処理は一定の領域毎に画素を生成する処理であるから、その領域単位で画像の変化度合いが大きいか否かを判定する必要がある。各領域ごとにこの変化度合いを判定するのは煩雑であるから、ステップ S T 3 0 8 であらかじめエッジ画素であるか否かを判定してフラグを設定する。この場合、図 3 6 に示すように、エッジ画素を取り囲む全ての画素において画像の変化度合いが大きいものと判断する。より具体的には、各画素の変化度合いが図 3 7 (a) に示すようになっていくとすると、しきい値が「 3 2 」であれば、しきい値を越える画素は x y 座標で示すところの (0 , 0) (3 , 0) (4 , 0) (1 , 1) (2 , 1) であるとしても、エッジ画素の隣接画素にはフラグを設定することになる。すると、同図 (b) に示すように y = 0 , 1 の全画素と、 (4 , 2) を除く y = 2 の画素についてはフラグが設定されることになる。この結果、後の工程で各画素単位で注目ブロックを移動させていくときにフラグだけを参照して補間処理を適宜選択できるようになる。

30

【 0 0 9 7 】

むろん、本実施形態においては、これらのステップ S T 3 0 4 ~ S T 3 0 8 の処理が画素変化度合評価ステップに相当する。むろん、これらが C P U などのハードウェアと有機一体的に結合したものと考えると画素変化度合評価手段 D 3 を構成することになる。

40

以上のように設定したフラグに基づき、ステップ S T 3 1 0 以下ではループ処理で補間画素を生成していく。図 3 8 は既存の画素に対して補間して生成される画素の配置を概略的に示している。既存の画素について仮に座標を (X , Y) として表示し、補間で生成される画素の座標を < X , Y > として表示している。同図の例では、約 2 . 5 × 2 . 5 倍の補間処理を行っている。

【 0 0 9 8 】

既存の四つの画素で囲まれる一つの領域をブロックと呼び、各ブロックごとに補間する画素の補間処理を選択する。ステップ S T 3 0 8 では各画素ごとに周囲の画素の変化度合も考慮してフラグを設定しているため、各ブロックでは四つの画素 (0 , 0) (1 , 0) (

50

0, 1) (1, 1) についていずれについてもフラグが設定されていれば変化度合の大きい場合の補間処理を選択することになり、どれか一つでもフラグが設定されていなければ変化度合の小さい場合の補間処理を選択することになる。ステップ S T 3 1 0 では、この条件に基づいて当該ブロック内部に適用する補間処理を判断し、変化度合いが小さい場合にはステップ S T 3 1 2 にてニアリスト法による補間処理で補間を実行するし、変化度合いが大きい場合にはステップ S T 3 1 4 にてキュービック法による補間処理で補間を実行する。また、一つのブロックを補間処理した後、ステップ S T 3 1 6 とステップ S T 3 1 8 にて処理対象となるブロックを移動させ、全てのブロックが終了すればステップ S T 3 2 0 にて補間された画像データを出力する。

【 0 0 9 9 】

なお、図中にはステップ S T 3 1 8 の終了後にステップ S T 3 1 0 に戻るような流れを実線で示しているが、破線で示すようにブロック毎にエッジ画素を集計する処理を繰り返すようにしても良い。

むろん、このような意味でステップ S T 3 1 0 の処理を中心としてステップ S T 3 1 6 , S T 3 1 8 の処理を含めて補間処理選択ステップに相当する。むろん、これらが C P U などのハードウェアと有機一体的に結合したものと考えると補間処理選択手段 D 4 を構成することになる。なお、プリンタドライバ 1 2 c の場合、解像度変換だけで印刷データが得られるわけではなく、色変換であるとか、ハーフトーン処理が必要になる。従って、ここで画像データを出力するというのは、次の段階へのデータの受け渡しを意味することになる。

【 0 1 0 0 】

本実施形態の場合は、四つの画素に囲まれる領域をブロックと呼んで補間処理を選択しているが、補間処理を変更する基準は演算能力や補間処理などに応じて適宜変更可能である。例えば、図 3 9 に示すように、注目画素を中心とする領域を基準として補間処理する場合もある。このような場合は、かかる注目画素を矢印のように走査させて移動させながら補間処理を適宜実施していけばよい。

ここで、注目画素を移動させていきながら補間処理を選択する手法について説明する。上述した例ではブロック毎に変化度合いが大きいか否かを判定するにあたり、当該領域に含まれる全てのフラグが「1」となっている場合にだけ画像の変化度合いが大きい領域と判断している。しかしながら、必ずしもこのように全てのフラグが「1」となっている必要もないともいえる。例えば、図 4 0 (a) に示すように 4 画素で囲まれる領域に補間処理で画素を生成するものとする。この場合、図 3 6 でエッジ画素の隣接画素にフラグを立てる関係上、本来であれば上述したように 4 画素全てにフラグが立っている場合だけが変化度合いの大きい領域と判定することになりえる。しかしながら、このように判断するとブロックを 1 画素分だ横に移動させた場合には縦の辺が共通する関係で縦の二画素について毎回判断が重複するし、縦方向に移動させれば横の辺が共通する関係で横の二画素について毎回判断が重複する。このような重複状況は演算処理において無駄である。一方、同図 (b) に示すように領域の隣接状況を考慮すれば、1 領域ごとに左上の 1 画素を代表させて関連づけることが可能であるし、少なくともエッジ画素に同視しうる画素の近辺で変化度合いが大きいと判断しても対して支障はないといえる。また、隣接画素同士に囲まれる領域というのは実際には極めて微少な領域であることを鑑みても十分であるといえる。そして、このようにして 1 画素に 1 領域を対応させれば、ブロックを移動させる際に注目画素を移動させていき、その注目画素のエッジ量だけで領域の変化度合いを判定することが可能となるし、判定に要する演算処理量も低減する。

【 0 1 0 1 】

また、補間する画素の側でブロックを形成するようにすることも可能である。図 4 1 はこの例を示しており、図中、 の格子点が補間する画素を示し、 の格子点が既存の画素を示している。いま、補間する画素について 5 × 5 のブロックを一つとし、その中に含まれる既存の画素のエッジ量に基づいて当該領域が画像の変化度合いの大きいものであるか否かを判断する。この場合、一つのブロックを決めて当該ブロックに含まれる既存の画素を

10

20

30

40

50

抽出し、そのエッジ量の積算値を求め、当該ブロック内では同一の補間処理で画素を生成すればよい。

【0102】

むろん、以上の場合においてより大きな領域毎にブロックを設定して補間処理を選択しても良く、例えば、 10×10 画素毎をブロックとすることも可能である。また、ブロックを設定せずに補間する画素毎にそれを取り囲む既存の画素についてのエッジ量を判断して補間処理を選択することも可能である。図41の例で言えば内側に配列される 3×3 の格子点は、いずれもで示す四つの既存の格子点の中に含まれ、それぞれの格子点を生成する際にこれを取り囲むで示す四つの既存の格子点についてのエッジ量に基づいて補間処理を選択するということである。むろん、演算処理上、このような処理の方が都合よい場合に実現すればよい。すなわち、先に補間処理するブロックを特定して補間処理を決めてからその内部に画素を補間するという手法であっても良いし、補間する画素毎にブロックの状況を判定して補間処理を選択しても良い。

10

【0103】

さらに、上述したフローではステップST308にて予めエッジ画素に隣接する画素にフラグを設定しておき、ブロック毎に同フラグを参照するようにしている。しかしながら、図32にて破線で示すようにブロックを移動させるフローとすることも可能であり、この場合には敢えてフラグを設定する必要もなく、当該ブロックの周囲の画素のエッジ量を判断して補間処理を選択するにすれば良い。

【0104】

上述したように別々の補間処理を備えているステップST312、ST314の処理は画素補間ステップに相当する。むろん、これらがCPUなどのハードウェアと有機一体的に結合したものと考えると画素補間手段D2を構成することになる。ここで、それぞれの補間処理について詳述する。

20

一方、このような関係は図29に示す補間関数において $t = 0 \sim 1$ の区間において傾斜が急となりつつ、 $t = 1 \sim 2$ の区間において増加した重み分を打ち消すように負の側へ引き寄せるカーブとなっている場合に生じる。

【0105】

従って、シャープさを調整しようとする場合には、1 補間関数においてシャープさの基準となる理想的な傾斜を決定し、2 $t = 0 \sim 1$ の区間において上記傾斜を発生させるカーブを決定し、3 $t = 1 \sim 2$ の区間においてこのカーブによって増える重み付けを相殺するように負の側に引き寄せつつ、オーバーシュートとアンダーシュートが生じやすいカーブを決定することによって実現できる。むろん、この後の作業では特定されるカーブとなるように多次演算関数のパラメータを決定するが、かかるパラメータの決定方法は極めて多様であるから、実質的な意味でS字カーブにおける中央部分の傾斜角度とアンダーシュート及びオーバーシュートを調整することに他ならない。

30

【0106】

各補間処理には以上のような特性の違いがあり、ステップST310にて画像の変化度合いが小さいと判断されたブロックでは、ステップST312にてニアリスト法の補間処理を実行するし、逆に変化度合いが大きいと判断されたブロックでは、キュービック法やハイブリッドバイキュービック法の補間処理を実行する。キュービック法で補間処理をする場合には演算時間が多大になってしまうものの、画像の変化度合いが小さいような部分ではニアリスト法に切り替えるため、全体としての処理時間は極めて低減する。特に、コンピュータグラフィックスのように同色で一定領域を塗りつぶしてあるような場合には一律にニアリスト法を実行しても全く問題ないので、処理時間は低減する。また、自然画であっても拡大したときにジャギーが目立ちやすい部分というのは面積比でいってもそれほど大きくないのが普通であるから、このように画像の変化度合いを逐次切り替えることによって画質を劣化させることなく処理量を低減させることができる。

40

【0107】

本実施形態においては、フラグによって二種類ある補間処理のいずれかを実行するように

50

しているが、画素の変化度合いに対して段階的に対応する複数の補間処理を実行するようにしても良い。また、図42に示すように、二つの補間処理を重ねて実行することとしてその拡大倍率を画像の変化度合いに対応させるようにしても良い。例えば、補間倍率が5倍であるとして画像の変化度合いが小さめであればニアリスト法で5倍に補間処理するし、画像の変化度合いが大きめであればキュービック法で5倍の補間処理する。これらの場合は上述した実施形態と同様であるが、画像の変化度合いが中間的な値である場合にはキュービック法で2倍に補間処理し、残りの2.5倍をニアリスト法で補間処理する。このようにして二つの補間処理でありながら実質的には画像の変化度合いに応じた複数の補間処理を選択できることになる。

【0108】

なお、前述のように、キュービック法のような補間処理の演算量が大きいものについては、補間倍率を整数倍とする。

このように、画像入力デバイスを有するとともに画像出力デバイスを有するコンピュータシステム10において、プリンタドライバ12cはステップST302にて元画像データを入力した後、ステップST304~108にて画像の変化度合いを検出してフラグを設定しておき、ステップST310にて同フラグを参照することにより、画像の変化度合いの小さいブロックではステップST312にてニアリスト法による補間処理を実行するし、画像の変化度合いの大きいブロックではステップST314にてキュービック法による補間処理を実行するようにしたため、画質を劣化させない範囲でできる限りニアリスト法を実行するように制御され、自動的に最適な補間処理を選択しつつ演算処理量を低減させる。

【0109】

以上説明したように、発明においては、画像をドットマトリクス状の画素で表現した画像データを取得する画像データ取得手段と、上記画像データに基づいて画素の変化度合いを評価する画素変化度合評価手段と、上記画像データにおける構成画素数を増やす補間処理を行うに複数の補間処理の中から選択して実行可能な画素補間手段と、上記画素変化度合評価手段によって評価された画素の変化度合いに基づいてその変化度合いに対応して最適な補間結果を得ることが可能な補間処理を選択して上記画素補間手段に実行させる補間処理選択手段とを具備する構成としてある。

【0110】

このように構成した本発明においては、画像をドットマトリクス状の画素で表現した画像データの構成画素数を増やす補間処理を行うにあたり、画素補間手段は複数の補間処理の中からいずれかを選択して実行可能となっており、画像データ取得手段が対象となる画像データを取得すると、上記画素変化度合評価手段は同画像データに基づいて画素の変化度合いを評価する。そして、上記補間処理選択手段はこの画素変化度合評価手段によって評価された画素の変化度合いに基づき、その変化度合いに対応して最適な補間結果を得ることが可能な補間処理を選択して上記画素補間手段に実行させる。

【0111】

すなわち、画素の変化度合は補間処理の具体的手法に密接に関連するので、同画素の変化度合を評価して積極的に補間処理を変更することにより、無駄のない補間処理を実現する。

以上説明したように本発明は、画像の変化度合いに応じて補間処理を変更することにより極めて簡易に最適な補間結果を得ることが可能な画像データ補間装置を提供することができる。

画素変化度合評価手段は、画素の変化度合を評価するものであり、評価の手法や結果は特に限定されるものではない。また、補間処理選択手段での同評価結果の利用態様に応じて相対的に変更可能なものである。例えば、具体的な変化度合いを数値として要するのであれば数値を出力すればよいし、単に変化度合いが大きいか否かといったものでよければ、それに合わせて出力すればよい。

【0112】

また、画素の変化自体をどのようにして把握するかも適宜変更可能である。その一例として、上記画素変化度合評価手段は、各画素の明るさのパラメータを求めるとともに周囲の画素のパラメータとの比較で上記変化度合いを算出する構成とすることもできる。

このように構成した場合には、画素の評価の基準として当該画素の明るさを利用するものとし、上記画素変化度合評価手段は各画素の明るさのパラメータを求め、当該画素とその周囲の画素とで同パラメータを比較し、比較結果を上記変化度合いとして算出する。

【0113】

むろん、これ以外にも画素の変化度合いを把握することは可能であるが、多要素のパラメータで表される画素を一律に把握するにあたって明るさのパラメータは比較的容易である。

10

このようにすれば、画像の変化度合いを明るさのパラメータに基づいて判断するため、比較的容易に同変化度合いを求めることができる。

一方、補間処理の処理内容にも画素の変化度合いが影響を及ぼす範囲が異なってくると言える。例えば、補間処理を実行するにあたって必要とする画素の数が一つであるものもあれば、複数の画素に基づいて補間処理するものもある。特に、後者の例であれば、一つでも変化度合いが大きい画素があると補間処理を変えるのか、あるいは一つでも変化度合いが小さい画素があると補間処理を変えるのかといったことが問題となる。

【0114】

このような状況に対する一例として、上記画素変化度合評価手段は、各画素ごとに求めた上記変化度合いを周囲の画素の変化度合の評価についても利用する構成とすることもできる。

20

補間処理で要する画素が複数であり、そのうちの一つでも変化度合いが大きいときには、二つの態様が考えられる。すなわち、補間処理で対象とする範囲の残り画素については評価する必要がなくなるし、逆に既に評価した画素について変化度合いが小さかったとしても当該評価は不要となる。

【0115】

このような二方向の意味で、各画素ごとに求めた上記変化度合いを周囲の画素の変化度合の評価についても利用することになる。

このようにすれば、一の画素の変化度合いを周囲の画素においても利用することにより、演算量を低減できるし、補間処理の影響を受ける適当な範囲で共有することにより、最適な補間結果を得ることができる。

30

画素補間手段では、画素の変化度合いに関連する複数の補間処理を実行可能であればよく、補間処理自体としては各種の処理が可能である。その一例として、上記画素補間手段は、変化度合の小さい領域で適用して好適な補間処理として、補間処理前の最近隣画素の画像データを新たな構成画素の画像データに利用する補間処理を実行可能な構成とすることもできる。

【0116】

このように構成した場合には、一つの補間処理として補間処理前の最近隣画素の画像データを新たな構成画素の画像データに利用するが、同じ画素のデータが増えるとしても変化度合の小さい領域であれば何ら問題なく、処理量が少ない点で好適である。

40

このようにすれば、変化度合の小さい領域では画質に影響することなく処理量を減らすことができる。

また、他の一例として、上記画素補間手段は、変化度合の大きい領域で適用して好適な補間処理として補間する画素の画像データがなだらかに変化するように周囲の画素の画像データから演算処理で補間画素の画像データを算出する補間処理を実行可能な構成とすることもできる。

【0117】

このように構成した場合には、周囲の画素の画像データを利用して演算処理することにより、補間する画素の画像データはなだらかに変化する。このように、なだらかに変化させると、変化度合いの大きい画素の並びがあったとして、この間を補間したとしても段差が

50

目立たない。従って、変化度合いの大きい画素の並びについてこの間を補間したとしても段差が目立たず画質の劣化を防止することができる。

【0118】

補間する画素の画像データがなだらかに変化する演算手法は各種のものを採用可能であるが、その変化態様は画質に影響を与える。このため、ある意味では演算手法を変えることによって画質を調整可能となるともいえる。画質を調整可能な一例として、上記画素補間手段は、変化度合いの大きい画素間で補間画素の画像データを算出するにあたり、画像データの変化態様を略S字型としつつその傾斜を調整するとともに、両端部位では低い側にアンダーシュートを発生させつつ高い側にオーバーシュートを発生させて高低差を形成してその高低差を調整することにより、画像の変化度合いを最適なものとするように調整する構成とすることもできる。

10

【0119】

このように構成した場合には、補間する画素の画像データをなだらかに変化させるにあたり、変化度合いの大きい画素間で画像データの変化態様を略S字型とする。従って、なだらかに変化するもののその変化態様は単に直線的に結ぶ勾配よりは急峻とさせることができ、その傾斜を調整して画像の変化度合いを最適なものとするのが可能となる。また、両端部位で低い側にアンダーシュートを発生させつつ高い側にオーバーシュートを発生させると高低差は大きくなり、かつ、その高低差を調整することによっても見かけ上の画像の変化度合いを最適なものとするのが可能となる。このような演算処理の一例としては、多次演算処理の3次たまたみ込み内挿法などを使用可能であるし、かかる調整を可能とする演算処理はこれに限られず、他の演算手法を採用することもできる。

20

【0120】

このようにすれば、S字カーブの傾斜と、アンダーシュートとオーバーシュートによる高低差とにより、画質の調整を比較的容易に実現できる。

画素の変化度合いが画像全体にわたって一定であることはないため、補間処理選択手段は、適宜、補間処理を選択して切り換えなければならない。そして、かかる切り換えの頻度も特に限定されるものでなく、各種の手法を採用可能である。その一例として、上記補間処理選択手段は、上記画素変化度合評価手段によって評価された画素の変化度合に基づいて画素単位で上記補間処理を選択して実行させる構成とすることもできる。

30

【0121】

このように構成した場合には、上記画素変化度合評価手段によって評価された画素の変化度合に基づき、上記補間処理選択手段が画素単位で上記補間処理を選択して実行させる。すなわち、変化度合いが画素単位で評価される以上、これに対応して補間処理も変更する。

従って、画素単位で補間処理を選択するのできめ細かに補間結果を向上させることができる。

また、他の一例として、上記補間処理選択手段は、上記画素変化度合評価手段によって評価された画素の変化度合に基づいて複数画素からなる所定の小領域毎に上記補間処理を選択して実行させる構成とすることもできる。このようにすれば、小領域毎に補間処理を選択するので処理を簡易化することができる。

40

【0122】

次に、補間処理を前提としつつシャープさを併せて修正する実施形態を説明する。

図43は、このような画像データ補間装置を表すブロック図である。

元の画像が自然画であるとする、画像によってはシャープさに欠けることがある。例えば、ピントの甘いような写真などが該当する。また、装置間の解像度を一致させるための拡大にとどまらず、画像自体を拡大して出力したいような場合には、シャープさの欠ける画像はさらにピントが甘くなりかねない。

本画像データ補間装置は画像データについて画素単位での拡大処理を実施する際にシャープさを調整するものであり、画像データ取得手段E1は、同画像データを取得し、画素補間手段E2はこの画像データにおける構成画素数を増やす補間処理を行う。ここで、画素

50

補間手段 E 2 は補間処理に付随して画像のシャープさを変化させることが可能となっており、シャープ度合評価手段 E 3 が上記画像データに基づいて画像のシャープさを評価する。すると、補間処理制御手段 E 4 はそのようにして評価されたシャープさが低ければこれを高めるような補間処理を実行するように上記画素補間手段 E 2 を制御する。

【 0 1 2 3 】

上述したように、オペレーティングシステム 1 2 a で管理する解像度とカラープリンタ 1 7 b の解像度とが一致しない場合にプリンタドライバ 1 2 c は解像度を一致させる処理を実行する。通常、カラープリンタ 1 7 b の解像度はオペレーティングシステム 1 2 a が管理する解像度よりも細かいので、解像度を一致させるためには画素を増やすための補間処理が行われる。このようにアプリケーション 1 2 d によって拡大処理する場合と、プリンタドライバ 1 2 c によって解像度を一致させる場合に補間処理が行われるが、これらの補間処理で画像のシャープさに影響を与えることができる。

10

【 0 1 2 4 】

画像のシャープさはそれぞれの隣接画素間での変化度合いの総合評価と言える。シャープさに欠ける画像というのは本来のエッジ部分でなだらかに画素が変化していることを意味し、シャープな画像では本来のエッジ部分で隣接画素間の変化度合いが急峻である。補間処理は既存の画素と画素の間に新たな画素を生成することになるので、新たな画素をどのような値とするかで画像のシャープさが変化するからである。

【 0 1 2 5 】

この意味で、本発明の画像データ補間装置では、アプリケーション 1 2 d やプリンタドライバ 1 2 c が上述した画素補間手段 E 2 はもとより、以下に述べるようにシャープ度合評価手段 E 3 や補間処理制御手段 E 4 を構成する。

20

図 4 4 は、補間処理を実行する一例としてのプリンタドライバ 1 2 c が実行する解像度変換に関連するソフトウェアフローを示している。

ステップ S T 4 0 2 は元画像データを取得する。ステップ S T 4 0 4 ~ S T 4 0 8 は、読み込んだ画像データにおける各画素の変化度合いから画像のシャープさを評価する処理である。

【 0 1 2 6 】

図 3 3 や図 3 4 に示すエッジ検出フィルタを利用した結果を各画素のエッジ量 E と呼ぶと、その分布は図 3 5 に示すように正規分布的となることが予想される。このようにしてドットマトリクス状の画素の全てにおいて算出するのがステップ S T 4 0 6 の処理であり、エッジ量の絶対値をステップ S T 4 0 8 にて集計する。集計は単純な平均値であっても良いが、背景部分の面積比の影響を受けやすいとも言える。例えば、図 4 5 では被写体たる人物像が大きく映って背景部分が少ないが、図 4 6 では被写体たる人物像が小さく映って背景部分が多くなる。背景部分では画素の変化度合いが小さくなりがちであるから、背景部分の面積割合が大きい図 4 6 に示すものでは図 4 5 に示すものと比較して平均値が低くなりがちである。この意味で、或る一定のしきい値を設けておき、そのしきい値以上のものだけの平均を算出するようにしても良い。

30

【 0 1 2 7 】

一方、この集計の段階では画像のシャープさを求めるのが主目的であるが、そもそも自然画のようなシャープさを要求される画像であるのか否かをこの集計結果から判断することも可能である。自然画の場合は単なる背景のような部分であっても色の明暗であるとか背景としての実物の形状に応じて同一の画素が並んでいるわけではないことから、エッジ量の絶対値の集計結果は図 4 7 に示すようになり、エッジ量は大きめになりがちである。これに対してビジネスグラフのような画像では同色で一定領域を塗りつぶすことが多いので、エッジ量の絶対値の集計結果は図 4 8 に示すようになり、エッジ量が低めになる。

40

【 0 1 2 8 】

従って、集計結果の平均値 (a v) があるしきい値 T h より低い場合にはシャープさを増すような処理が必要でない画像と言え、シャープさに影響を与えない補間処理を実行させるようにすればよい。

50

以上のようにしてステップST404～ST408において画像を構成する各画素の変化度合いを集計し、当該画像がシャープな画像と言えるか否かの評価を実施したことになるため、これらのステップST404～ST408の処理がシャープ度合評価手段E3を構成することになる。

【0129】

この評価結果に基づき、ステップST410では画像のシャープさの高低に応じた補間処理を選択する。本実施形態においては、シャープな画像に対してキュービック法による補間処理を実行し、シャープさに欠ける画像に対してハイブリッドバイキュービック法による補間処理を実行する。従って、この意味で当該ステップST410は補間処理制御手段E4を構成するし、別々の補間処理を備えているステップST412, ST414の処理は画素補間手段E2を構成することになる。ここで、それぞれの補間処理について詳述する。

10

【0130】

各補間処理には以上のような特性の違いがあり、ステップST410にて画像がシャープであると判断されればステップST412にてキュービック法の補間処理を実行するし、逆にシャープではないと判断されるとハイブリッドバイキュービック法の補間処理を実行する。ハイブリッドバイキュービック法で補間処理をする場合には補間するカーブが急峻となってシャープさを増すことができ、かかる補間処理が選択されるのは対象とする画像のシャープさを評価し、その評価結果に基づくものである。このため、操作者は特段の判断をしなくてもシャープでない画像をシャープにすることができる。

20

【0131】

本実施形態においては、二種類ある補間処理のいずれかを実行するようにしているが、画素の変化度合いに対して段階的に対応する複数の補間処理を実行するようにしても良い。図49はシャープさの評価を4段階に分けて3次内挿法のパラメータを変化させた四つのキュービック法を実施する例を示している。図中「0」は通常のシャープさの画像に適用されるキュービック法のカーブを示しており、「+1」のカーブはわずかにシャープさに欠ける画像に適用されるキュービック法を示しており、「+2」のカーブはかなりシャープさが欠ける画像に適用されるキュービック法を示している。また、シャープすぎる画像については「-1」のカーブのキュービック法を適用する。むろん、これらはいずれもハイブリッドバイキュービック法の場合と同様にS字カーブにおける中央部分の傾斜角度とアンダーシュート及びオーバーシュートを調整して実現している。

30

【0132】

次に、この場合の手続のフローを図50に示す。

画像のシャープさに基づいてステップST510ではこれらのパラメータを設定し、かかるパラメータを使用したキュービック法をステップST514にて実行する。また、このフローでは、画像のシャープさが部分的に異なることを考慮し、画像を小領域であるブロックに分割して各ブロック毎に最適な補間処理を実行する。すなわち、各ブロック毎にシャープさを評価して補間処理を選択するため、ステップST508にてブロック毎のエッジ量を集計し、ステップST516, ST518にてブロックを順次移動させながら補間処理を実行するようにしている。

40

【0133】

全ての画像データについて補間処理を終了したら、ステップST420やステップST520にて補間された画像データを出力する。なお、プリンタドライバ12cの場合、解像度変換だけで印刷データが得られるわけではなく、色変換であるとか、ハーフトーン処理が必要になる。従って、ここで画像データを出力するというのは、次の段階へのデータの受け渡しを意味することになる。

このように、画像入力デバイスを有するとともに画像出力デバイスを有するコンピュータシステム10において、プリンタドライバ12cはステップST402にて元画像データを入力した後、ステップST404～108にて画像のシャープさを評価して集計するとともに、ステップST410にて同集計結果に基づいてシャープさに欠ける画像であれば

50

ステップST414にてシャープさを増す補間処理を実行するし、シャープな画像であればステップST412にて通常の補間処理を実行するようにしたため、操作者が別段にシャープさを増す画像処理を選択しなくても補間処理を経るだけでシャープな画像とすることができる。

【0134】

以上説明したように、本実施形態においては、画像をドットマトリクス状の画素で表現した画像データを取得する画像データ取得手段と、上記画像データに基づいて画像のシャープさを評価するシャープ度合評価手段と、上記画像データにおける構成画素数を増やす補間処理を行うにあたり画像のシャープさを変化させる補間処理を実行可能な画素補間手段と、上記シャープ度合評価手段によって評価された画像のシャープさが適当でなければシャープさを変化させて適当となるように上記画素補間手段に補間処理を実行させる補間処理制御手段とを具備する構成としてある。

10

【0135】

このように構成した場合には、画像をドットマトリクス状の画素で表現した画像データの構成画素数を増やす補間処理を行うにあたり、画素補間手段は画像のシャープさを変化させる補間処理を実行可能となっており、画像データ取得手段が対象となる画像データを取得すると、上記シャープ度合評価手段は同画像データに基づいて画像のシャープさを評価する。そして、上記補間処理制御手段はこのシャープ度合評価手段によって評価されたシャープさに基づき、画像のシャープさが適当でなければシャープさを変化させて適当となるように上記画素補間手段に補間処理を実行させる。

20

【0136】

すなわち、画像のシャープさが低い場合には補間処理によってシャープさを増し、また、シャープすぎる場合にはシャープさを低減させる。このようにすれば、画像のシャープさに応じて補間処理でシャープさを調整しているため、操作を煩雑にすることなく簡易に画質を向上させることが可能な画像データ補間装置を提供することができる。補間を実行しつつ画像のシャープさを変化させる手法として画像データの変化態様を略S字型としつつその傾斜を調整するとともに、両端部位では低い側にアンダーシュートを発生させつつ高い側にオーバーシュートを発生させて高低差を形成してその高低差を調整することにより、画像の変化度合いを最適なものとするように調整して画像のシャープさを変化させることを示した。

30

【0137】

このようなS字カーブをとる一例として、上記画素補間手段は、3次たたまみ込み内挿法におけるパラメータを調整して画像のシャープさを変化させる構成とすることもできる。このように構成した場合には、補間処理として利用される3次たたまみ込み内挿法のパラメータを調整することにより、元の画像での隣接する画素の間に補間される画素が3次関数を採用することによってS字を描き、なだらかでありながら急峻さも併せ持つことになる。そして、このS字の曲がり具合をパラメータで調整することによって急峻さが変化し、画像のシャープさが変化する。

40

【0138】

このようにすれば、多次演算処理として3次たたまみ込み内挿法を利用することにより、S字カーブを調整して比較的容易にシャープさを調整することができる。シャープさを変化させるにあたり、必ずしも一つの演算法だけを採用する必要はなく、シャープさに影響を与える複数の補間処理を実行することも可能である。そのような一例として、上記画素補間手段は、画像のシャープさの変化度合いの異なる複数の補間処理を実行可能であるとともに、それぞれの補間倍率の割合を変化させて画像のシャープさを調整する構成とすることもできる。

【0139】

このように構成した場合には、複数の補間処理のそれぞれで画像のシャープさの変化度合いが異なり、必要な補間倍率を得るために複数の補間処理を実行する。従って、その補間

50

倍率の分担割合を互いに变化させることにより、シャープさを調整可能となる。例えば、シャープさの変化度合いの低い補間処理とシャープさの変化度合いの高い補間処理とがある場合に両者の分担割合を变化させれば二つの変化度合いの中間を選択可能となる。

【0140】

このようにすれば、複数の補間処理で分担する補間倍率を変えるだけであるので、パラメータの設定が簡易になる。

画像のシャープさは、必ずしも高ければよいわけではない。従って、シャープさを増す必要がない場合もある。この場合、操作者が判断することも可能であるが、かかる判断を同時に実現する構成とすることもできる。その一例として、上記補間処理制御手段は、上記画像のシャープさが所定のしきい値を越えていると評価されたときに上記画素補間手段にて画像のシャープさを变化させるように制御する構成とすることもできる。

10

【0141】

このように構成した場合には、上記補間処理制御手段が上記画像のシャープさと所定のしきい値とを比較し、画像のシャープさがこれを越えていると評価されたときに上記画素補間手段にて画像のシャープさを变化させるように制御する。しかしながら、画像のシャープさがしきい値を越えていないようであればあえて画像のシャープさを变化させるようには制御しない。例えば、自然画であるときの画像のシャープさと非自然画であるときの画像のシャープさとを比較すれば、前者のものの方が一般的にはシャープさが高いと言えるからである。むろん、画像のシャープさは自然画か非自然画かといった分類だけで決まるものでもないため、他の判断要素を加えることも可能である。例えば、画像の分類を取得し、その分類にたった上で上記しきい値を变化させればより柔軟な対応が可能となる。

20

【0142】

このようにすれば、ある一定の範囲まではシャープさを变化させないようにするため、シャープさを变化させることが不適当な画像まで自動的にシャープさを調整してしまうといった不便さがなくなる。

画面のシャープさは個々の画素を基準とすると、画像全体にわたって一定であるわけではないので、必ずしも一定の補間処理に限られるものでもない。このため、画素単位で画像のシャープさを評価するとともに、評価された画像のシャープさに基づいて画素単位で画像のシャープさを变化させるように制御する構成とすることもできる。また、所定の小領域毎に画像のシャープさを評価するとともに、評価された画像のシャープさに基づいて所定の小領域毎に画像のシャープさを变化させるように制御する構成としてある。

30

【0143】

次に、画像処理を選択できるようにした実施形態を説明する。

図51は、このような画像データ補間装置を表すブロック図である。

本画像データ補間装置は画像データについて画素単位の画像処理を実施する際に拡大処理とシャープさの変更処理を補間処理として同時に行うものであり、画像データ取得手段F1にて同画像データを取得するとともに、画像処理選択手段F2で実施する画像処理を選択する。同時処理判断手段F3は実行すべき画像処理が拡大処理とシャープさの変更処理とを同時に行うものであるか否かを判断するものであり、同時に行う必要がある場合には画素補間手段F4が当該画像データにおける構成画素数を増やす補間処理を行なうのに付随して画像のシャープさを变化させる。そして、拡大処理後の画像データは画像データ出力手段F5が出力する。

40

【0144】

かかるコンピュータシステム10では、画像入力デバイスであるスキャナ11aなどで画像データを取得し、アプリケーション12dで所定の画像処理を実行する。この画像処理には各種のものがあり、拡大縮小、シャープさの強弱、コントラストの強弱、色合いの修正といったものがあげられる。なお、上述したように、画像出力デバイスとしてのディスプレイ17aやカラープリンタ17bに表示出力するためには解像度の一致が必要であり、特にプリンタの解像度に合わせる際には拡大処理としても実行する補間処理が行われる。そして、この補間処理はプリンタドライバ12cで実行しても良いし、アプリケーショ

50

ン 1 2 d で実行することもできる。

【 0 1 4 5 】

ここで、アプリケーション 1 2 d では、画像処理として拡大処理とシャープさの変更処理を選択することが可能であり、また、プリンタの解像度に合わせる際に拡大処理するのに伴ってシャープさを変更することも可能である。従って、これらの場合に拡大処理とシャープさの調整処理を実行することになる。

この意味で、本発明の画像データ補間装置は、上述したコンピュータシステム 1 0 におけるアプリケーション 1 2 d として実現されることになる。そして、アプリケーション 1 2 d は上述した画素補間手段 F 4 はもとより、以下に述べるように画像処理選択手段 F 2 や同時処理判断手段 F 3 を構成する。また、データの入出力が伴うので、ファイル入力やファイル出力あるいは印刷データの出力という意味で画像データ取得手段 F 1 や画像データ出力手段 F 5 を構成する。

【 0 1 4 6 】

図 5 2 は、アプリケーション 1 2 d のソフトウェアフローの概略を示している。アプリケーション 1 2 d は図 5 3 に示すようなメニュー選択によって各種の画像処理を実行可能であり、必ずしも図 5 2 に示すようなソフトウェアフローに限定されるものではないが、理解の便宜のために簡略化して表示している。

ステップ S T 6 0 2 では元画像データを取得する。アプリケーション 1 2 d におけるファイルメニューなどでスキャナ 1 1 a から画像を読み込む処理などが該当する。本実施形態においては後述する画像処理に使用する画像データを生成すればよいので、新規ファイル作成などを選択して画像ファイルを生成するような処理であっても同様に元画像データの取得といえる。オペレーティングシステム 1 2 a やハードウェアの構成を除いた元画像データの取得処理が画像データ取得に相当する。むろん、これらが C P U などのハードウェアと有機一体的に結合したものと考えると画像データ取得手段 F 1 に該当する。

【 0 1 4 7 】

ステップ S T 6 0 4 では、画像処理を選択し、ステップ S T 6 0 6 , S T 6 0 8 では、選択された画像処理を判断する。画像処理を選択するには、図 5 3 に示すように画面上にてウィンドウ枠に表示されるメニューバーの中から、マウス 1 5 b によって「画像」の文字部分をクリックすると、図 5 4 に示すように実行可能な各種の画像処理が表示される。同図に示す例では、「拡大」処理と、「シャープネス」調整処理と、「コントラスト」調整処理と、「明るさ」調整処理とが実行可能となっている。それぞれの画像処理には「ボタン」スイッチを用意しており、複数の画像処理を同時に選択可能となっている。また、「ボタン」スイッチで選択されるまでは各処理はグレイ表示されており、非表示状態が一別できるようになっている。

【 0 1 4 8 】

所望の画像処理のボタンスイッチで選択状態とするとともに、それぞれのパラメータをセットして「OK」ボタンをクリックすると、選択を入力した処理として扱われ、ステップ S T 6 0 6 , S T 6 0 8 にて選択された処理を判断する。この例では、ステップ S T 6 0 6 にて拡大処理が選択されたか否かが判断され、拡大処理が選択されていなければステップ S T 6 1 0 にてそれぞれに対応する画像処理を実施する。一方、拡大処理が選択されている場合には、ステップ S T 6 0 8 にてシャープネス調整処理が選択されているか否かを判断する。この結果、拡大処理とシャープネス調整処理が共に選択されているならば、ステップ S T 6 1 2 へと進み、拡大処理は選択されているもののシャープネス調整処理は選択されていないのであればステップ S T 6 1 4 へと進む。なお、ステップ 1 1 2 , S T 6 1 4 の処理については後述する。

【 0 1 4 9 】

このソフトウェアフローでは、理解しやすくステップ S T 6 0 6 , S T 6 0 8 の二つの分岐処理で行っているが、実際にはケース処理で多数の分岐を選択するようにしてもよい。本実施形態においては、ステップ S T 6 0 4 にて画像処理を選択するので画像処理選択ステップに相当するし、ステップ S T 6 0 6 , S T 6 0 8 にて拡大処理とシャープネス調整

10

20

30

40

50

処理が共に選択されているか否かを判断することになるので同時処理判断ステップに相当する。むろん、これらがCPUなどのハードウェアと有機一体的に結合したものと考えると画像処理選択手段F2や同時処理判断手段F3に該当する。

【0150】

ここで、画像処理選択手段F2や同時処理判断手段F3の変形例について説明する。図55～図57はその一例を示しており、アプリケーション12dが図54に示すように明示的なシャープネス調整処理を備えていないものとし、拡大処理は選択可能となっているとする。この例では、ステップST704にて画像処理の選択として図56に示すような拡大処理のパラメータ入力ウィンドウを表示する。ここでは拡大率を%で選択入力可能となっており、所望の倍率にセットして「OK」ボタンをクリックすると、次の段階へと進行する。通常であれば、拡大率を入力することによってその倍率に応じた補間処理を実行すればよいが、この例ではステップST706にて拡大処理が選択されていると判断すると、ステップST708にて図57に示すようなシャープネス調整の問い合わせ用ウィンドウを表示する。このウィンドウには「NO」、「Low」、「High」の三つの選択肢が用意され、それぞれに択一的に選択可能なボタンが割り当てられている。デフォルトは「NO」であり、操作者は必要に応じて「Low」や「High」を選択できるようになっている。これらはシャープネス強調についての問い合わせであり、「NO」は強調せず、「Low」はやや強調し、「High」は強調するという意味である。

10

【0151】

そして、選択された結果に基づいて、補間処理であるステップST712, ST714, ST716のいずれかを実行する。拡大処理の際には、後述するように補間処理を選択すればシャープネスを変化させることが可能であるので、拡大処理を選択した場合に自動的に問合せようにしている。この例では、ステップST704, ST708が画像処理選択ステップに相当するし、ステップST710が同時処理判断ステップに相当する。

20

【0152】

また、図58～図60には、直接には画像処理を選択しているようには見えないが、実際には内部的に拡大処理するような場合を示しており、より具体的には印刷処理の場合である。図示しないファイルメニューの中から印刷を選択すると、ステップST804にて図59に示すような印刷メニューを表示する。この印刷メニューの中でも各種のパラメータを設定可能であるが、その一つとして「印刷解像度」の選択ボックスがある。アプリケーション12dが内部的に扱っている解像度とは関係なく、印刷時にどの解像度で印刷実行するかによって解像度の一致作業が必要となる。カラープリンタ17b2の解像度が720dpiであるとして、印刷解像度を720dpiとして印刷する場合には画像データの1ドットが印刷時の1ドットと対応するので解像度変換は不要である。しかしながら、300dpiで印刷する際には印刷データとの対応を一致させなければならず、この意味での解像度変換が必要となってくる。

30

【0153】

このため、ステップST808にて印刷解像度ボックスのパラメータとオペレーティングシステム12aが管理しているカラープリンタ17b2の解像度とを比較して解像度変換を実施する必要があるか否かを判断し、解像度変換が必要な場合にはステップST810にて図60に示すようなシャープネスの問い合わせ用ウィンドウを表示する。この例では図54の場合と同様にシャープネスの調整程度を%で選択して入力するようにしており、ステップST812では入力されたパラメータに従って処理を選択する。すなわち、調整処理が要求されない場合にはステップST814へと進み、調整処理が要求される場合にはステップST816へ進むようにしている。この後、解像度が一致した状態でステップST818にて印刷処理を実行する。

40

【0154】

この例では、ステップST804, ST810が画像処理選択ステップに相当するし、ステップST812が同時処理判断手段ステップに相当する。むろん、これらの画像処理選択手段F2と同時処理判断手段F3についてはこれら以外の手法で実現することが可能で

50

あることはいうまでもない。

以上のような判断を経て補間処理が実行されることになる。具体的には、拡大処理だけが選択されてシャープさの強調処理が選択されない場合にはニアリスト法による補間処理を実行するし（ステップST614，ST712，ST814）、拡大処理とシャープさの強調処理が選択されたときにはハイブリッドバイキュービック法による補間処理（ステップST612，ST816）とキュービック法による補間処理（ステップST714）を実行する。従って、後者のステップST612，ST816，ST714が画素補間手段F4を構成することになる。

【0155】

本実施形態のように、拡大処理の中でシャープさを調整できるようにすることは、単に拡大処理とシャープさ調整処理とを別個に行うよりも優れている点がある。例えば、シャープさを強調してから拡大する処理を行うとすると、拡大処理がニアリスト法のようなものであればジャギーが目立ってしまい、シャープな感じを維持できないことがある。また、ニアリスト法で拡大してからシャープさを強調するとした場合は、ジャギーが目立った状態でシャープにすることになるので、画質が向上するとも言えない。これに対して、一体の拡大処理の中でシャープさをも合わせて行うようにすれば、このような弊害は生じにくい。

【0156】

画像データについて補間処理や他の画像処理を終了したら、ステップST66にて画像データを出力する。ここでいう画像データを出力するというのは広義の意味を含んでおり、カラープリンタ17b2に出力するとか、ハードディスク13bに書き込むといった処理に限られず、データとしては保持しておきながらディスプレイ17b1に表示させ、次なる画像処理に備えるというものであっても構わない。むしろ、本実施形態においては、このステップST616が画像データ出力手段F5を構成する。

【0157】

このように、画像入力デバイスや画像出力デバイスなどを有するコンピュータシステム10において、アプリケーション12dは各種の画像処理を実行可能となっており、ステップST604にて実行すべき画像処理を選択させたとき、拡大処理とシャープさの変更処理とが同時に指定された場合には、ステップST608の判断を経てステップST612にてシャープさを増す補間処理を実行し、拡大処理だけが選択された場合にはステップST614にてシャープさに影響を与えない通常の補間処理を実行するようにしたため、拡大処理とシャープさの変更処理とを個別に実行するための余分な時間がかからないし、両者が一体的に実行されるので、確実にシャープさを調整することができる。

【0158】

以上説明したように、発明においては、画像をドットマトリクス状の画素で表現した画像データを取得する画像データ取得手段と、上記画像データに対して個々の画素における画像データを変更することによって各種の画像処理を実行するべく実行可能な画像処理を表示して選択を入力する画像処理選択手段と、この画像処理選択手段によって画像の拡大処理と画像のシャープさの変更処理とが共に選択されたか否かを判断する同時処理判断手段と、この同時処理判断手段によって画像の拡大処理と画像のシャープさの変更処理とが共に選択されたとき、上記画像データにおける構成画素数を増やして画像を拡大するにあたり、この補間する画像データの変化度合いを調整することにより選択された画像のシャープさとなるように補間処理を実行可能な画素補間手段と、生成された画像データを出力する画像データ出力手段とを具備する構成としてある。

【0159】

このような構成とした本発明においては、画像データ取得手段にて画像をドットマトリクス状の画素で表現した画像データを取得したら、この画像データに対して個々の画素における画像データを変更することによって各種の画像処理を実行するべく画像処理選択手段にて実行可能な画像処理を表示して選択を入力させる。ここで、同時処理判断手段はこの画像処理選択手段によって画像の拡大処理と画像のシャープさの変更処理とが共に選択さ

10

20

30

40

50

れたか否かを判断し、両処理が共に選択されたと判断されれば、画素補間手段は上記画像データにおける構成画素数を増やして画像を拡大するに際し、この補間する画像データの変化度合いを調整することにより選択された画像のシャープさとなるように補間処理を実行する。そして、画像データ出力手段は生成された画像データを出力する。

【0160】

すなわち、画像の拡大とシャープさを変更する処理を同時に実行させる必要が生じれば、補間処理で生成する画像データを調整することによって拡大しつつシャープさを変化させる。

以上説明したように本発明は、拡大処理で必要となる補間処理によって画像のシャープさを変更するようにしたため、拡大処理とシャープさの変更処理とを個別に行う必要が無く、処理時間を短くすることが可能な画像データ補間装置を提供することができる。

10

【0161】

また、同時に行われるので、拡大処理によってはその後のシャープさを変更する際に良好な結果を得られなくなったりすることもないし、むしろ、シャープさを変更してから拡大処理することによってシャープさの変更処理が無駄になってしまうこともない。

画像処理選択手段は実行可能な画像処理を表示して選択を入力するものであり、同時処理判断手段は画像の拡大処理と画像のシャープさの変更処理とが共に選択されたか否かを判断するものである。これらは、結果的に両処理を同時に実行する必要があるか否かを判断するものであればよく、選択の態様などは適宜変更可能である。

【0162】

20

その一例として、上記画像処理選択手段は、拡大処理の選択とシャープさ変更の選択とを個別に選択可能であり、上記同時処理判断手段は、この画像処理選択手段にて拡大処理の選択とシャープさ変更の選択とが同時に選択されたか否かを判断する構成とすることもできる。

このように構成した場合には、拡大処理の選択とシャープさ変更の選択とを個別に選択可能であるので、拡大処理だけであるとか、シャープさ変更だけが選択されることもある。そして、同時処理判断手段はそのような状況を前提として拡大処理の選択とシャープさ変更の選択とが同時に選択されたか否かを判断する。

【0163】

このようにすれば、拡大処理とシャープさの変更処理が個別に選択できる場合に好適である。

30

また、他の一例として、上記画像処理選択手段は、拡大処理を選択可能であるとともに、上記同時処理判断手段は、上記画像処理選択手段にて拡大処理が選択されたときにシャープさの変更度合を選択させる構成とすることもできる。

このように構成した場合には、上記画像処理選択手段で拡大処理だけが選択可能となっており、シャープさを変更する選択までは入力しない。しかしながら、この画像処理選択手段にて拡大処理が選択されたときには、上記同時処理判断手段が独自に判断してシャープさの変更度合を選択させる。むしろ、シャープさを変更しないことを選択することも可能であって、その場合には拡大処理だけを実行することになるし、逆にシャープさを変化させることを選択した場合には拡大処理とシャープさを変更する処理とを同時に選択されたものと判断する。

40

【0164】

従って、表面上は拡大処理しか選択できないような場合においても合わせてシャープさを変更することが可能となる。

さらに、上述した例では拡大処理を明示的に選択するようになっているが、拡大処理自体は明示的なものに限られる必要もない。その一例として、上記同時処理判断手段は、上記画像処理選択手段が画像処理に伴って解像度の変換処理を行うときに、シャープさの変更度合を選択させる構成とすることもできる。

【0165】

このように構成した場合には、上記画像処理選択手段において選択された画像処理に伴っ

50

て付随的に解像度の変換処理を行う必要性が生じることがあり、この場合に上記同時処理判断手段は、シャープさの変更度合を選択させる。そして、シャープさを変化させることを選択した場合には拡大処理とシャープさを変更する処理とを同時に選択されたものと判断する。

このようにすれば、付随的に拡大処理するような場合にもシャープさを合わせて変更することができる。

【0166】

なお、これらの場合における表示と入力の操作はGUIのもとでの画面表示やマウス操作などで実現しても良いし、ハードウェア的なスイッチで実施することも可能であるなど、適宜変更可能である。

【図面の簡単な説明】

【図1】本発明の一実施形態にかかる画像データ補間装置のブロック図である。

【図2】同画像データ補間装置の具体的なハードウェアのブロック図である。

【図3】本発明の画像データ補間装置の他の適用例を示す概略図である。

【図4】本発明の画像データ補間装置の他の適用例を示す概略図である。

【図5】本発明の画像データ補間装置の他の適用例を示す概略図である。

【図6】本発明の画像データ補間装置の他の適用例を示す概略図である。

【図7】本発明の画像データ補間装置における汎用的なフローチャートである。

【図8】本発明の画像データ補間装置におけるより具体的なフローチャートである。

【図9】元画像の大きさを示す図である。

【図10】サンプリング周期を示す図である。

【図11】サンプリング画素数を示す図である。

【図12】元画像とサンプリングされる画素の関係を示す図である。

【図13】非自然画についての輝度のヒストグラムを示す図である。

【図14】自然画についての輝度のヒストグラムを示す図である。

【図15】画像の変化度合いを直交座標の各成分値で表す場合の説明図である。

【図16】画像の変化度合いを縦軸方向と横軸方向の隣接画素における差分値で求める場合の説明図である。

【図17】隣接する全画素間で画像の変化度合いを求める場合の説明図である。

【図18】しきい値を変化させる領域を示す図である。

【図19】プリンタドライバからオペレーティングシステムに問い合わせを行う状況を示す図である。

【図20】ニアリスト法の概念図である。

【図21】ニアリスト法で各格子点のデータが移行される状況を示す図である。

【図22】ニアリスト法の補間前の状況を示す概略図である。

【図23】ニアリスト法の補間後の状況を示す概略図である。

【図24】キュービック法の概念図である。

【図25】キュービック法の具体的適用時におけるデータの変化状況を示す図である。

【図26】キュービック法の具体的適用例を示す図である。

【図27】ハイブリッドバイキュービック法の具体的適用例を示す図である。

【図28】バイリニア法の概念図である。

【図29】補間関数の変化状況を示す図である。

【図30】整数倍の補間処理を示す概略図である。

【図31】本発明の一実施形態にかかる画像データ補間装置の概略ブロック図である。

【図32】本発明の画像データ補間装置におけるフローチャートである。

【図33】エッジ検出フィルタの一例を示す図である。

【図34】エッジ検出フィルタの他の一例を示す図である。

【図35】エッジ量の分布としきい値との関係を示す図である。

【図36】注目画素とフラグの設定判断で対象とする画素の関係を示す図である。

【図37】エッジ量とフラグの設定状況を示す図である。

10

20

30

40

50

【図 3 8】既存の画素で形成されるブロックと補間する画素との関係を示す図である。
 【図 3 9】注目画素を基準としてブロックを形成する場合を示す図である。
 【図 4 0】フラグの状況と領域の対応を示す図である。
 【図 4 1】補間する画素でブロックを形成する場合を示す図である。
 【図 4 2】画像の変化度合いに応じて複数の補間処理の補間倍率を分配する関係を示す図である。

【図 4 3】本発明の一実施形態にかかる画像データ補間装置の概略ブロック図である。

【図 4 4】本発明の画像データ補間装置におけるフローチャートである。

【図 4 5】背景部分の小さい画像を示す図である。

【図 4 6】背景部分の大きい画像を示す図である。

10

【図 4 7】自然画のエッジ量の集計結果を示す図である。

【図 4 8】ビジネスグラフのエッジ量の集計結果を示す図である。

【図 4 9】補間関数の変化状況を示す図である。

【図 5 0】補間関数を選択するフローを示す図である。

【図 5 1】本発明の一実施形態にかかる画像データ補間装置の概略ブロック図である。

【図 5 2】本発明の画像データ補間装置におけるフローチャートである。

【図 5 3】画像処理を実行するためのメニューを表示を示す図である。

【図 5 4】画像処理を選択する画面を示す図である。

【図 5 5】画像データ補間装置の変形例を示すフローチャートである。

【図 5 6】拡大処理を指示する画面を示す図である。

20

【図 5 7】シャープネスを指示する画面を示す図である。

【図 5 8】印刷処理のフローチャートである。

【図 5 9】印刷時のパラメータを指示する画面を示す図である。

【図 6 0】シャープネスを指示する画面を示す図である。

【符号の説明】

1 0 ... コンピュータシステム

1 1 a ... スキャナ

1 1 a 2 ... スキャナ

1 1 b ... デジタルスチルカメラ

1 1 b 1 ... デジタルスチルカメラ

30

1 1 b 2 ... デジタルスチルカメラ

1 1 c ... ビデオカメラ

1 2 ... コンピュータ本体

1 2 a ... オペレーティングシステム

1 2 b ... ディスプレイドライバ

1 2 b ... ドライバ

1 2 c ... プリンタドライバ

1 2 d ... アプリケーション

1 3 a ... フロッピーディスクドライブ

1 3 b ... ハードディスク

40

1 3 c ... C D - R O M ドライブ

1 4 a ... モデム

1 4 a 2 ... モデム

1 5 a ... キーボード

1 5 b ... マウス

1 7 a ... ディスプレイ

1 7 a 1 ... ディスプレイ

1 7 b ... カラープリンタ

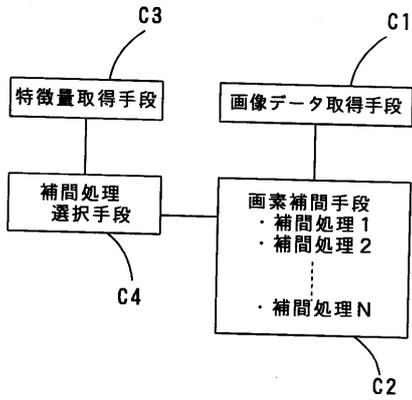
1 7 b 1 ... カラープリンタ

1 7 b 2 ... カラープリンタ

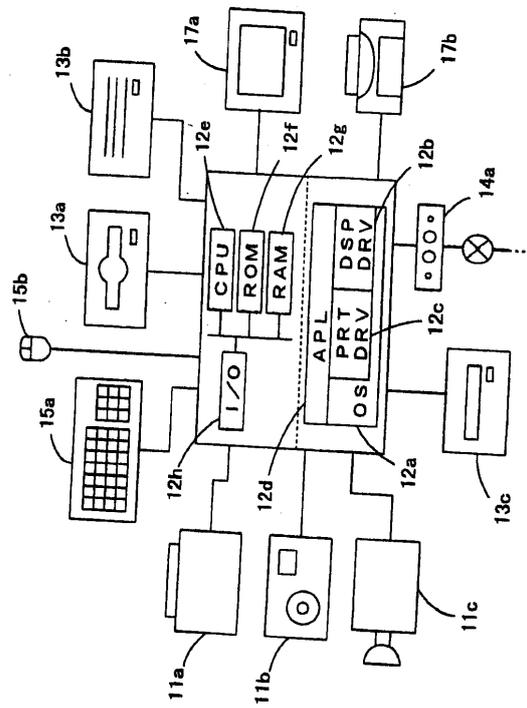
50

- 18 a ... カラーファクシミリ装置
- 18 b ... カラーコピー装置

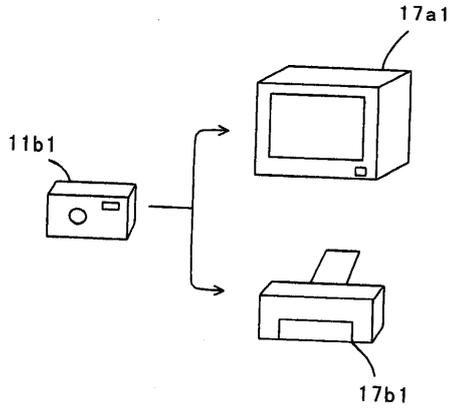
【図1】



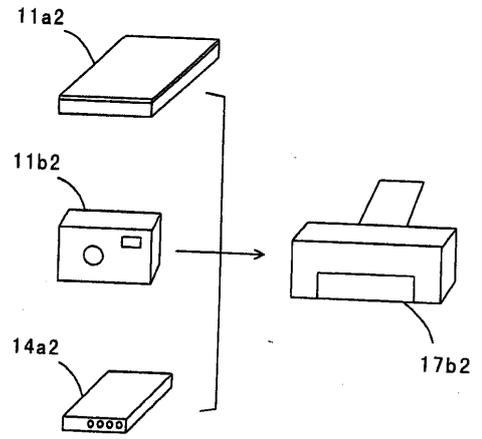
【図2】



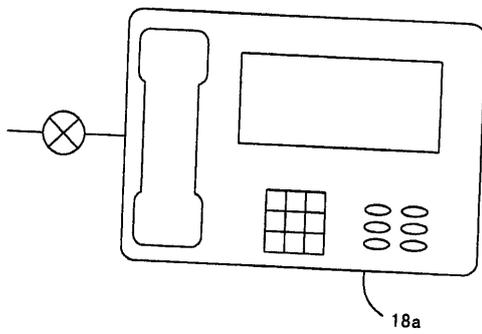
【図3】



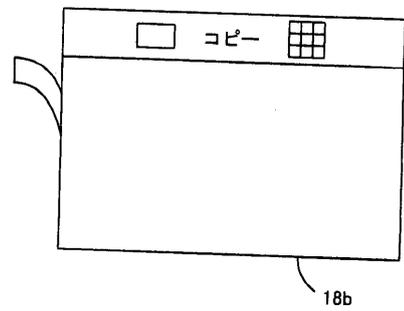
【図4】



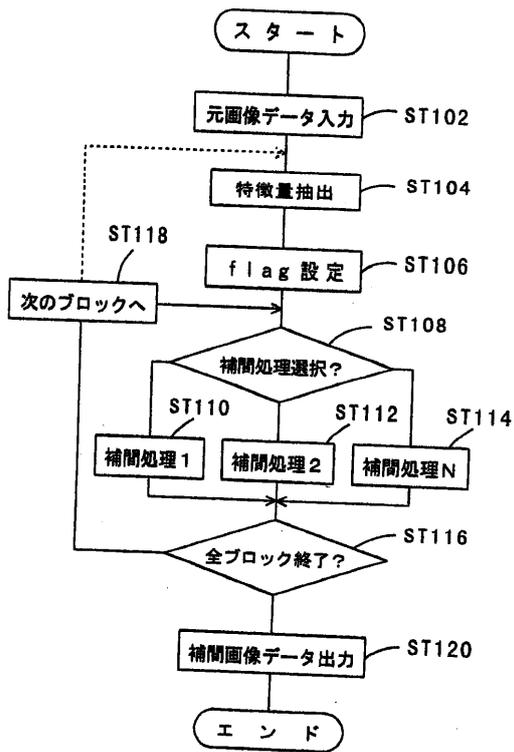
【図5】



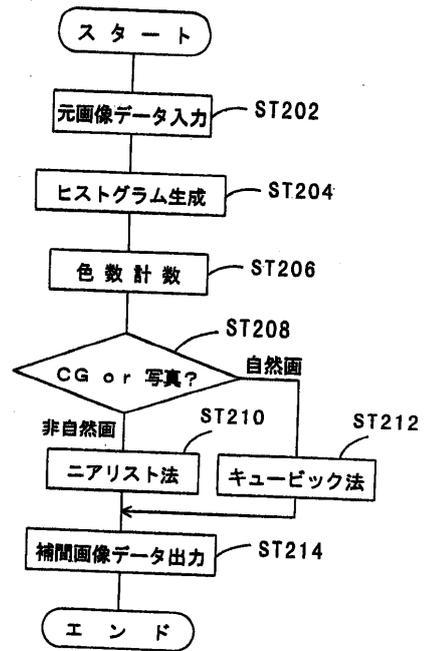
【図6】



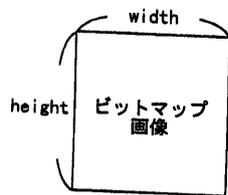
【図7】



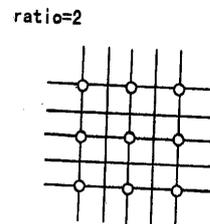
【図8】



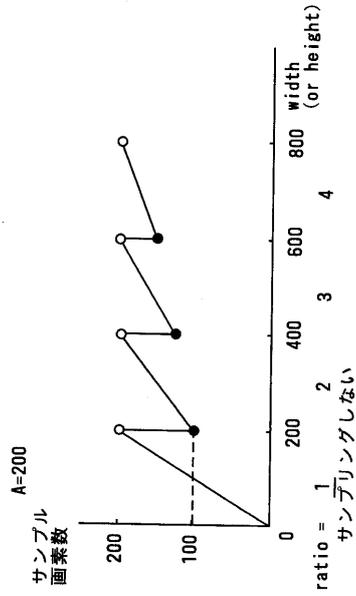
【図9】



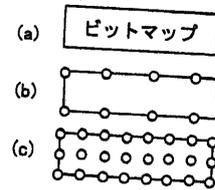
【図10】



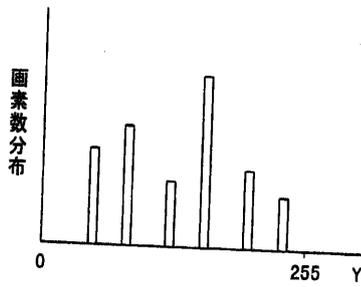
【図11】



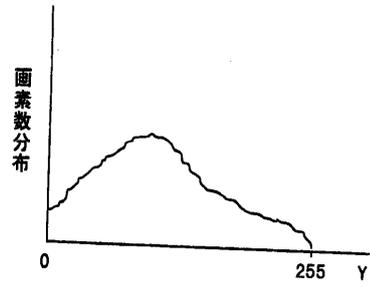
【図12】



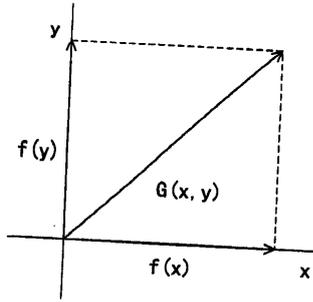
【図13】



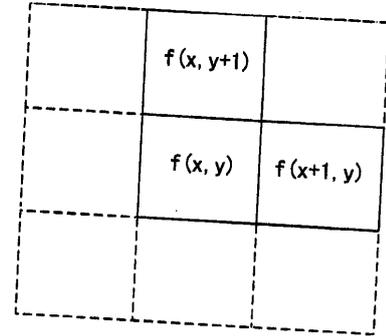
【図14】



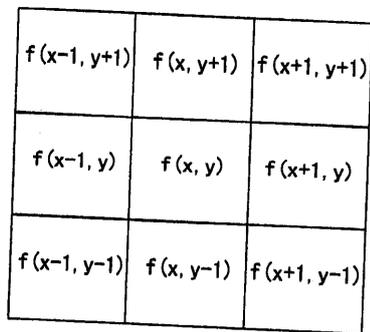
【図 15】



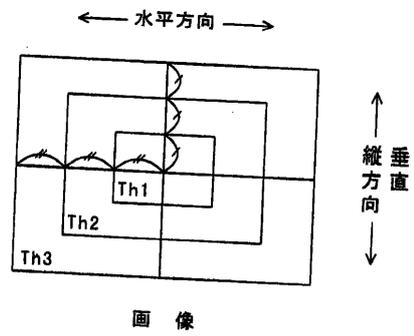
【図 16】



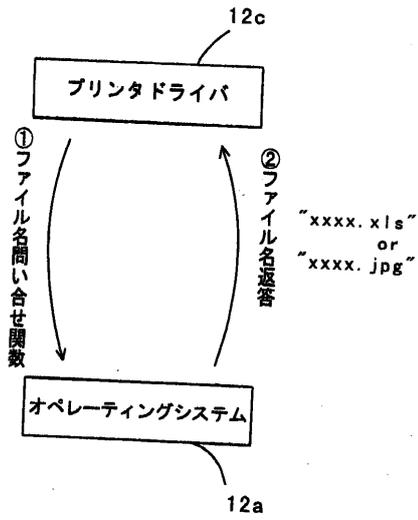
【図 17】



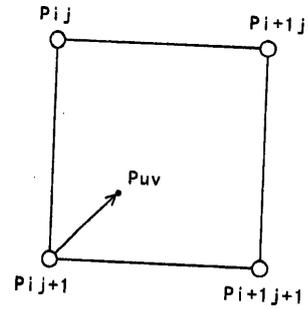
【図 18】



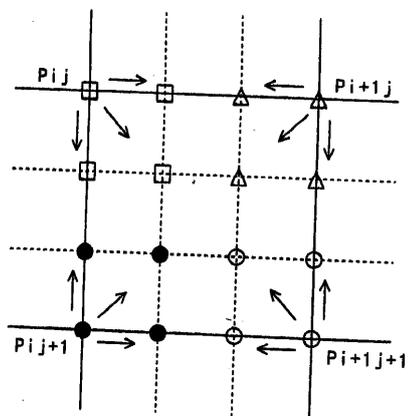
【図19】



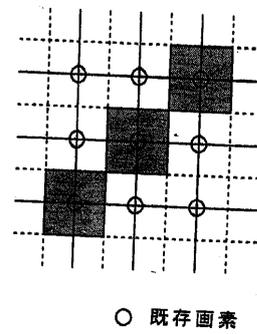
【図20】



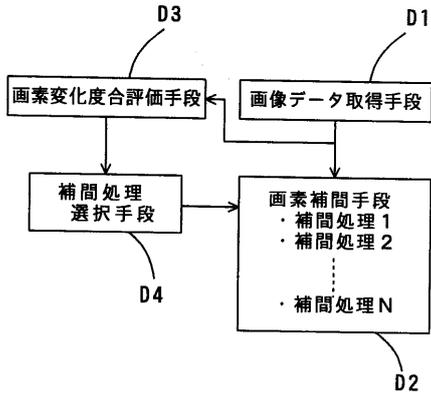
【図21】



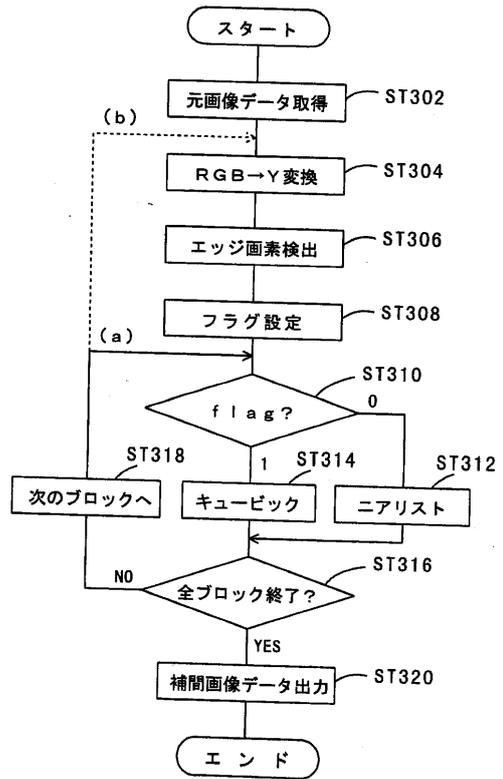
【図22】



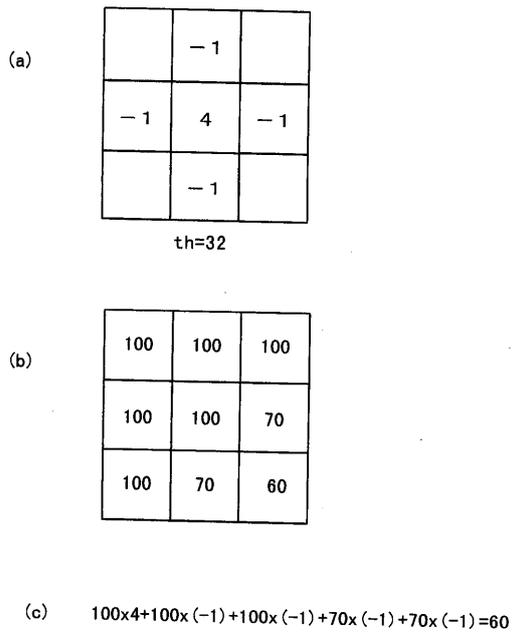
【図31】



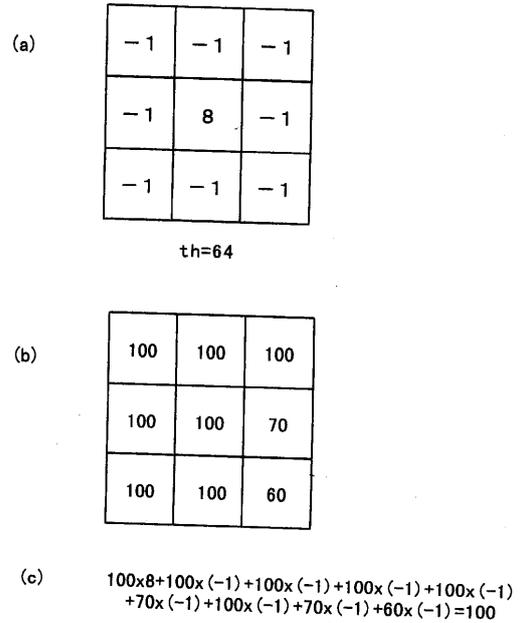
【図32】



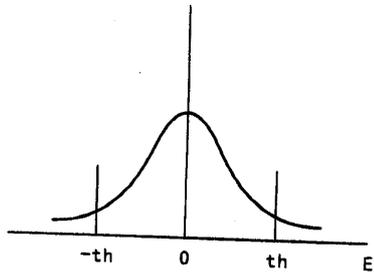
【図33】



【図34】

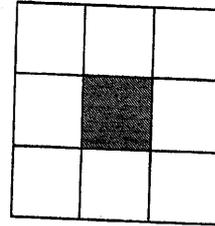


【図 35】



if $E < -th$ or $E > th$ then エッジ画素

【図 36】



注目画素（斜線部）を中心とする 3×3 画素に、
1つでもエッジ画素が存在すれば、flag=1
そうでなければ、flag=0

【図 37】

エッジ量E

th=32

(a)

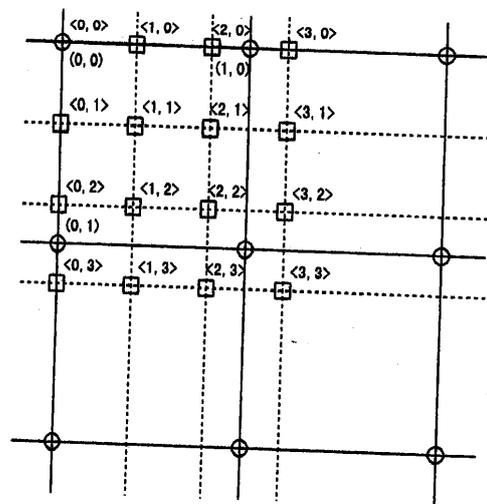
	x=0	1	2	3	4
y=0	40	20	20	18	38
1	12	38	39	8	10
2	20	16	8	4	2
3	0	10	10	14	14

flag

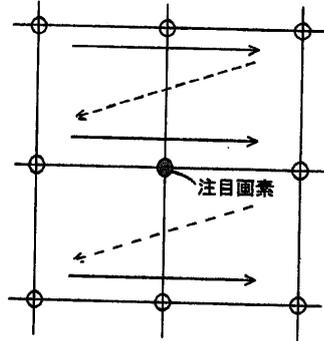
(b)

	x=0	1	2	3	4
y=0	1	1	1	1	1
1	1	1	1	1	1
2	1	1	1	1	0
3	0	0	0	0	0

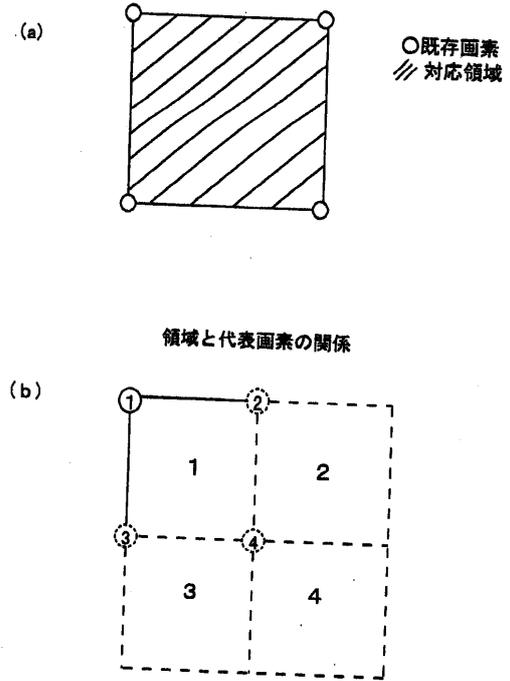
【図 38】



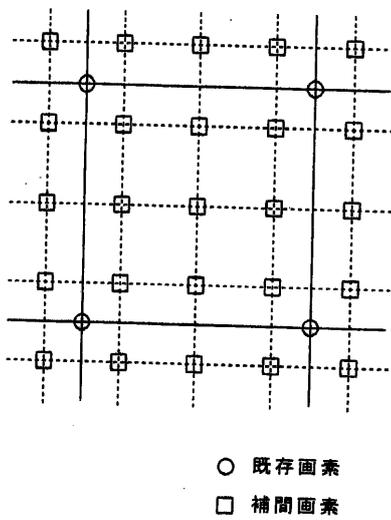
【図39】



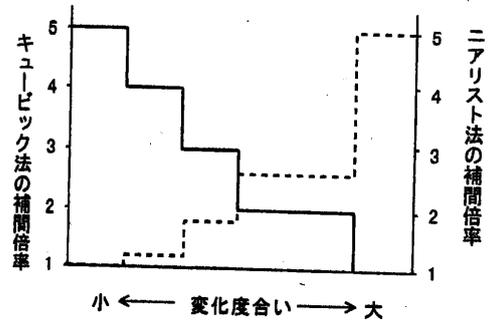
【図40】



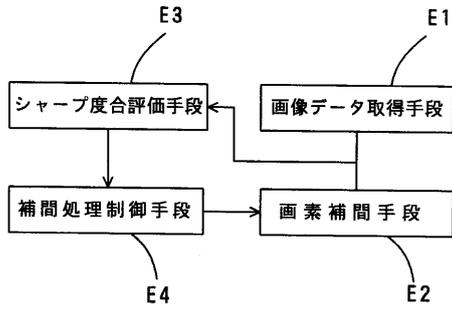
【図41】



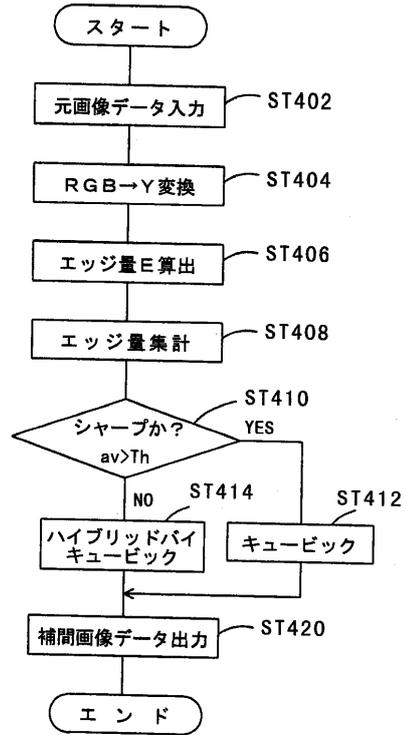
【図42】



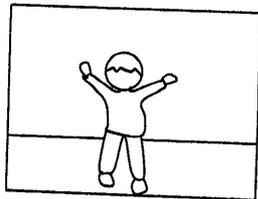
【図43】



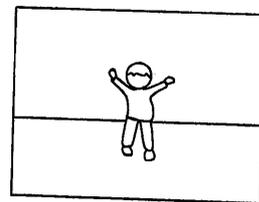
【図44】



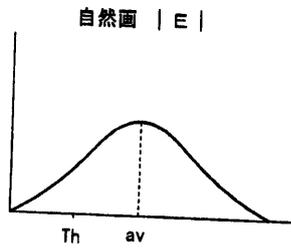
【図45】



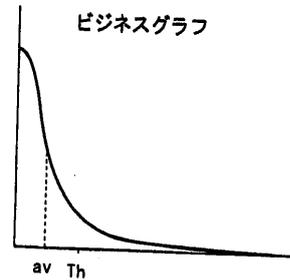
【図46】



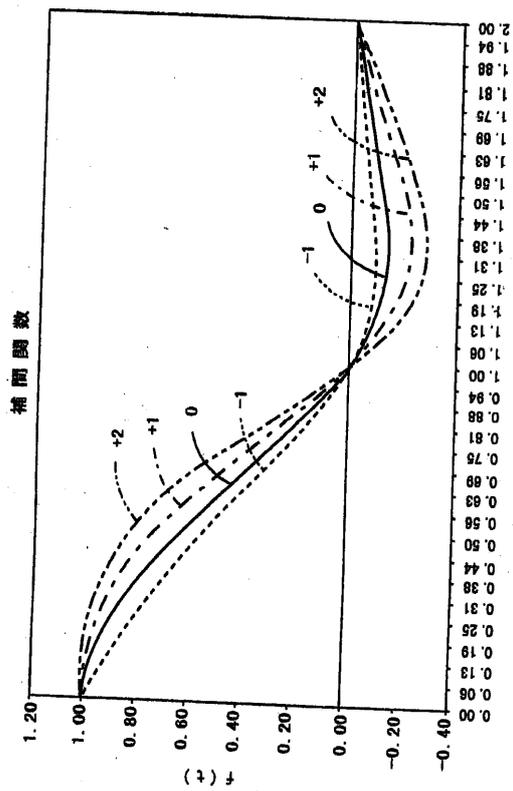
【図47】



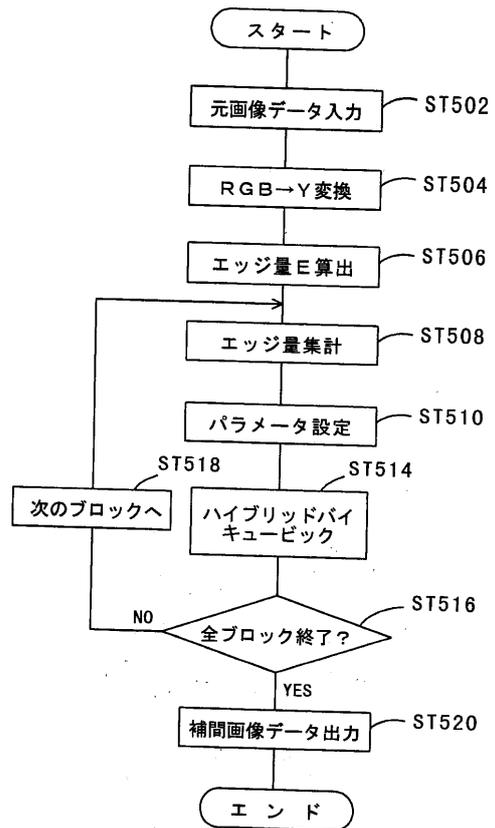
【図48】



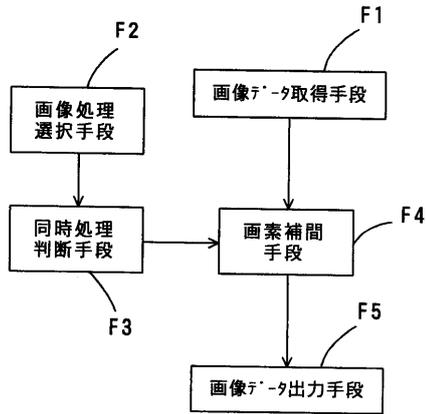
【図49】



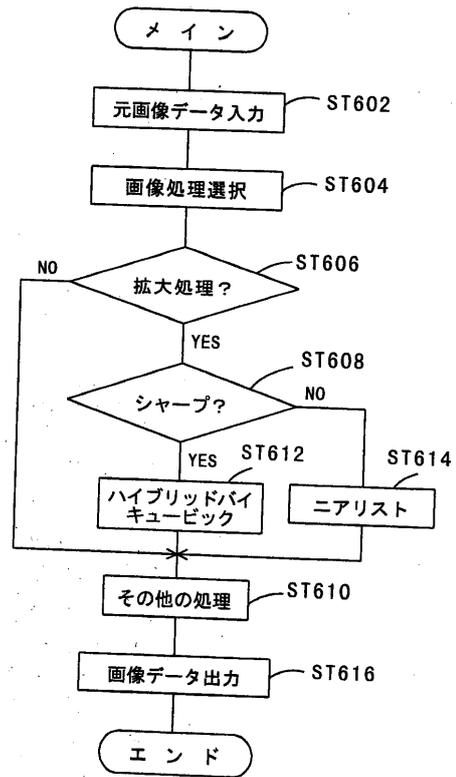
【図50】



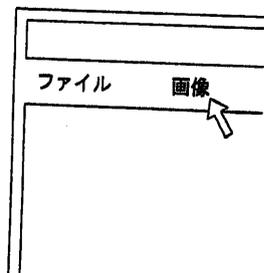
【図51】



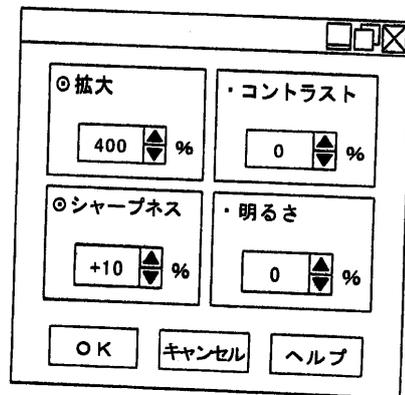
【図52】



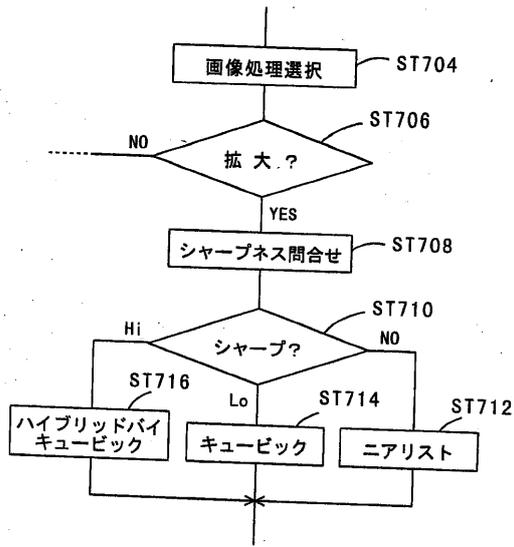
【図53】



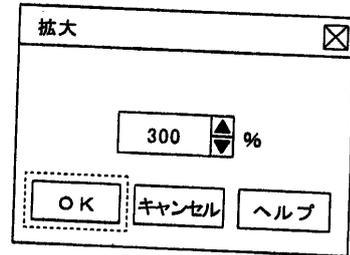
【図54】



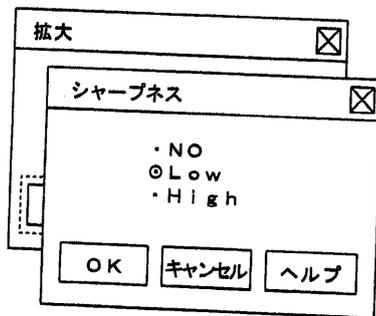
【図55】



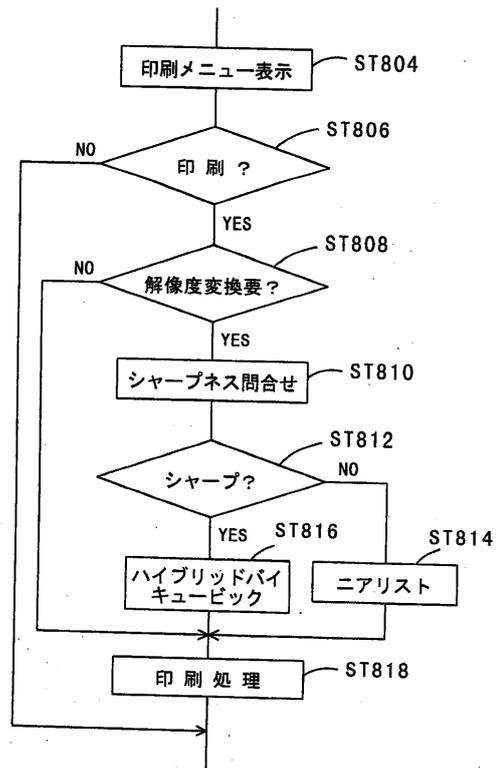
【図56】



【図57】

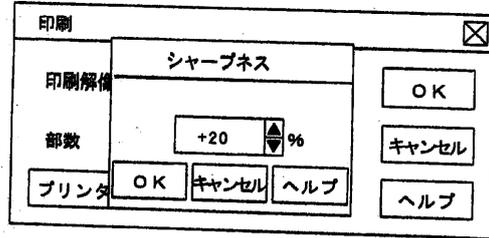
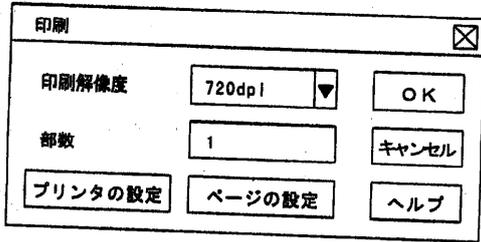


【図58】



【図59】

【図60】



フロントページの続き

- (56)参考文献 特開平 1 1 - 1 1 2 8 2 0 (J P , A)
特開平 0 6 - 2 3 7 3 6 9 (J P , A)
特開平 1 0 - 1 4 3 6 4 6 (J P , A)
特開平 0 5 - 1 5 8 5 3 5 (J P , A)
特開平 1 0 - 2 5 7 3 2 2 (J P , A)
特開平 0 8 - 3 2 0 9 9 8 (J P , A)
特開平 1 0 - 0 5 1 6 3 1 (J P , A)
特開平 0 8 - 3 2 9 2 5 4 (J P , A)

(58)調査した分野(Int.Cl. , D B 名)

H04N 1/387
G06F 3/12
G06T 3/40