

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4515107号  
(P4515107)

(45) 発行日 平成22年7月28日 (2010. 7. 28)

(24) 登録日 平成22年5月21日 (2010. 5. 21)

(51) Int. Cl.		F I	
<b>G06T</b>	<b>1/00</b>	<b>(2006.01)</b>	G O 6 T 1/00 5 1 0
<b>G06T</b>	<b>5/20</b>	<b>(2006.01)</b>	G O 6 T 5/20 B
<b>G06T</b>	<b>5/40</b>	<b>(2006.01)</b>	G O 6 T 5/20 C
<b>H04N</b>	<b>1/46</b>	<b>(2006.01)</b>	G O 6 T 5/40
<b>H04N</b>	<b>1/60</b>	<b>(2006.01)</b>	H O 4 N 1/46 Z

請求項の数 3 外国語出願 (全 26 頁) 最終頁に続く

(21) 出願番号	特願2004-28661 (P2004-28661)	(73) 特許権者	596170170
(22) 出願日	平成16年1月5日 (2004. 1. 5)		ゼロックス コーポレイション
(65) 公開番号	特開2005-326904 (P2005-326904A)		XEROX CORPORATION
(43) 公開日	平成17年11月24日 (2005. 11. 24)		アメリカ合衆国、コネチカット州 068
審査請求日	平成18年12月28日 (2006. 12. 28)		56、ノーウォーク、ピーオーボックス
(31) 優先権主張番号	10/612, 234		4505、グローバー・アヴェニュー 4
(32) 優先日	平成15年7月1日 (2003. 7. 1)		5
(33) 優先権主張国	米国 (US)	(74) 代理人	100082005
(31) 優先権主張番号	10/612, 461		弁理士 熊倉 禎男
(32) 優先日	平成15年7月1日 (2003. 7. 1)	(74) 代理人	100067013
(33) 優先権主張国	米国 (US)		弁理士 大塚 文昭
(31) 優先権主張番号	10/612, 062	(74) 代理人	100074228
(32) 優先日	平成15年7月1日 (2003. 7. 1)		弁理士 今城 俊夫
(33) 優先権主張国	米国 (US)	(74) 代理人	100086771
			弁理士 西島 孝喜

最終頁に続く

(54) 【発明の名称】 走査された文書の上のページ背景検知および中立性

(57) 【特許請求の範囲】

【請求項 1】

画素輝度値または画素値の範囲の数をカウントして、入力画像の輝度ヒストグラム・アレイを生成し、

各画素輝度値に対して、第2のおよび第3のアレイに画素クロミナンス値またはその範囲を加えて、当該クロミナンス・チャンネルの各々に対して一つずつの、従属クロミナンス・ヒストグラム・アレイであって、2つの、前記入力画像の、従属クロミナンス・ヒストグラム・アレイを生成し、

前記第1、第2、および第3の輝度ヒストグラム・アレイおよび従属クロミナンス・ヒストグラム・アレイを処理して、背景抑圧情報を獲得し、更に、黒点、白点、および、コントラスト・パラメータを含む強調パラメータを抽出し、これらのパラメータの値を用いて、前記入力画像に対して背景除去および追加の画像強調を自動的に適用する、

ステップを含み、

前記背景除去は、圧縮解凍とマージの後にバックエンドで行われ、ここで色調再生 (TRC) ルックアップ・テーブルを通して画素値が修正されることを特徴とする、背景抑圧情報および更に画像強調情報を得るための方法。

【請求項 2】

入力画像の輝度ヒストグラム・アレイを生成するために画素輝度値または画素値の範囲の数をカウントするための手段、

各画素輝度値に対して、第2のおよび第3のアレイに、画素クロミナンス値またはその

範囲を加え、当該各クロミナンス・チャンネルに対して一つずつの、従属クロミナンス・ヒストグラム・アレイであって、2つの、入力画像の、従属クロミナンス・ヒストグラム・アレイを生成するための手段、

背景抑圧情報を獲得して、更に、黒点、白点、およびコントラスト・パラメータを含む強調パラメータを抽出し、これらのパラメータの値を用いて、前記入力画像に、背景除去および追加の画像強調を自動的に適用するために、前記第1、第2、および第3の輝度ヒストグラム・アレイ、および、従属の、クロミナンス・ヒストグラム・アレイを処理するための手段、

を備え、

前記背景除去は、圧縮解凍とマージの後にバックエンドで行われ、ここで色調再生( T R C ) ルックアップ・テーブルを通して画素値が修正されることを特徴とする、背景抑圧情報および画像強調情報を獲得するためのシステム。

【請求項3】

画素輝度値または画素値の範囲の数をカウントして、入力画像の輝度ヒストグラム・アレイを生成し、

各画素輝度値に対して、第2のおよび第3のアレイに、画素クロミナンス値またはその範囲を加えて、当該クロミナンス・チャンネルの各々に対して一つずつの従属のクロミナンス・ヒストグラム・アレイであって、2つの、前記入力画像の、従属のクロミナンス・ヒストグラム・アレイを生成し、そして、

第1、第2、および、第3の輝度ヒストグラム・アレイ、および、従属のクロミナンス・ヒストグラム・アレイを処理して、背景抑圧情報を獲得し、更に、黒点、白点、およびコントラスト・パラメータを含む強調パラメータを抽出し、これらのパラメータの値を用いて、前記入力画像のヒストグラム・アレイの解析に基づいて自動的に修正されたルックアップ・テーブル・コンテンツを生成するために、前記入力画像に、背景除去および追加の画像強調を自動的に適用する、

ステップを含み、前記背景除去は、圧縮解凍とマージの後にバックエンドで行われ、ここで色調再生( T R C ) ルックアップ・テーブルを通して画素値が修正されることを特徴とする、背景抑圧情報および、更に、画像強調情報を獲得するための方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、走査された文書の上のページ背景検知および中立性に関する。

【背景技術】

【0002】

高解像度で走査された文書は、非常に大きな量の記憶空間を必要とする。更に、大きな画像データの量は、ネットワークの上を動き回るか、移動させるために、実質的により多くの時間と帯域を必要とする。生の走査済のRGBフォーマットで記憶する替わりに、その容量を削減するために、データは一般的に、何らかの形式のデータ圧縮を受ける。そしてそれによって、それを記憶するための高コストを避ける。Lempel-Ziv(LZ)のような「ロスレス」圧縮法は、走査された(ノイジーな)画素データに対して、特に良く動作しない。JPEGのような「損失のある(不可逆的)」方法は、連続トーン画素マップの上でかなり良く作動する一方、それらは、テキストおよびライン・アートを含むページの部分の上で特に良く作動しない。画像データ圧縮を最適化するために、圧縮されているデータのタイプを認識可能な技術が必要とされる。

【0003】

異なったタイプのデータの圧縮のニーズを満足させるための一つのアプローチは、画像を記述するために、Mixed Raster Content(MRC)フォーマットを利用するエンコーダ・パイプラインを使用することであった。画像(カラーまたはグレイ・スケール情報と混ぜ合わされたテキストを有する複合画像)は、2つまたはそれ以上の平面(planes)(一般的に、上方および下方平面と呼ばれる)にセグメント化され、セクタ平面は、各画素に対して

、どの画像平面が、最終出力画像を再構築するために使用されるべき実際の画像データを含むかを指示するために生成される。このやり方で平面をセグメント化することによって、画像の圧縮を改善出来る。何故なら、データは、オリジナルの画像よりも、平面がより円滑に、より圧縮可能になるように整頓される(arranged)からである。セグメント化はまた、異なった平面への、異なった圧縮方法の適用を可能とする。よって、各平面内のデータのタイプに対する、殆どの圧縮技術が適用される。

【 0 0 0 4 】

画像コンテンツに関する主要統計の収集に関して責任を負う、統計(状態)モジュールが望ましい。それは、入って来る画素データ・ストリームをモニターし、画素値およびそれらの色情報の分布を蓄積することになる。状態モジュールは、L a bまたはY C Cのような、輝度・クロミナンス・色空間ドメインで作動することになる。この理由から、R G Bからのスキャナ・データを、L a b色空間に変換するスキャナ・カラー変換ユニットの後に、状態モジュールを配置することが望ましい。

10

【 発明の開示 】

【 課題を解決するための手段 】

【 0 0 0 5 】

ページ背景除去および、ページがグレイか又は色付きか否かを判断するための自動ニュートラル(neutral)検知のような種々の画像強調動作の適用の目的のための画像コンテンツに関する主要統計を収集するための、統計モジュール(Statistics Module)(S T S)が開示される。スキャナ・ノイズのいくらかを削除するために、統計モジュールは、D e - S c r e e nモジュールからの滲んだ信号B L Rを使用する。出力は、空間的3 Dカラー・ヒストグラム・アレイである。

20

【 発明を実施するための最良の形態 】

【 0 0 0 6 】

走査された及び/又は印刷されたカラー文書を自動的に処理して、オリジナルの文書コンテンツを正確に捕捉する、小さな、高度に圧縮されたファイルを生成するための新規のシステムが説明される。エンコーダまたはフォント・エンドとしても知られる本システムは、高圧縮および高画像品質を保証するために必要とされる全ての必要な処理要素を含む。出力ファイルは、新規に生まれてきたM R C (Mixed Raster Content)再製(現在、T I F FとP D Fの双方、およびPost Scriptにも含まれる)のスタンダードに従って生成される。本システムの中核は、オリジナルのスクリーンを除去することによって文書を事前に調整(preconditions)する、ディ・スクリーナ(descreener)および、それに引き続く、入力されてくるデータを、(3つ或いは4つの)平面(planes)(背景、前景、セレクトア、および、任意のヒント平面(Hint planes))にセグメント化するセグメンタ(segmentor)から構成される。

30

【 0 0 0 7 】

本発明とともに使用されるエンコーダのブロック図たる、図1を参照する。ここで、エンコーダ・パイプラインは、左上のスキャナ入力S R C 200からR G Bデータを受け取る。スキャナ色変換(S C C)モジュール202は、R G BからY C C色空間(これは、デ・スクリーニングおよびセグメンテーション化のためにより便利である)に、入力データを変換する。モジュールS C C 202は、1 DのT R C ルックアップ・テーブルおよびそれに続いて3 × 3マトリックス乗算ユニットを使用する。

40

【 0 0 0 8 】

スキャナ色変換202は、2つの別個のステップで実行される。第1に、スキャナから入力されるソース信号S R C (通常R G B色空間でのもの)が、中間線形x Y C C色空間に変換される。このような関係において、x Y C Cが線形である点を除いて、つまり、s Y C Cのようにはガンマ関数が適用されない点を除いて、x Y C Cは、標準s Y C Cと類似する色空間を表す。第2のステップにおいて、圧縮の直前に、x Y C C色空間は、I C C - L a bに変換される(これは以下に更に詳細に説明される)。エンコーダ・パイプラインは、C E F M R C色空間表現に対しての、以下の採り得る選択をサポートする。

50

## 【 0 0 0 9 】

- 1) 走査された文書(走査コンフィギュレーション)のために、I C C - L a b が用いられる。そして、  
 2) 合成(ノイズ・フリーの)データ(P D L コンフィギュレーション)のために、x Y C C が用いられる。

## 【 0 0 1 0 】

色空間の限定された選択の理由は、色空間は、多くの制約：

- ( a ) 第 1 に最も優先して、それは、広い範囲の製品によって共通に理解される、標準の装置独立(device-independent)のフォーマットでなければならぬ；  
 ( b ) P D F が、L A B に加えて R G B のような追加の選択をサポートするのと対照的に、フロントエンド・エンコーダが送出しうるものである T I F F 仕様が、現在 L a b フォーマットだけをサポートし、現在 s Y C C が追加されることが検討されている；  
 ( c ) 追加の圧縮のための人間の視覚システムの特性に影響力を与えるために、輝度・クロミナンス表現を持つ色空間を持つことが望ましい；そして、  
 ( d ) 色空間は、処理および操作のために便利でなければならない上に、ユーザに、色を操作及び/又は調整するための容易な方法を提供しなければならない；  
 を満足させねばならぬためである。

## 【 0 0 1 1 】

I C C - L a b 色空間の仕様は、複数の利点：

- ( 1 ) それは、殆ど全てのオペレーティング・システムおよびアプリケーションによってサポートされる標準色空間である；  
 ( 2 ) それは、装置独立である；  
 ( 3 ) それは、色データを、輝度とクロミナンス要素に分離する。これは、取り扱いと圧縮のために便利である。；そして、  
 ( 4 ) それは、人間の視覚システムに倣ってモデル化され、可視色空間内にあるどの一つのユニットのベクトル長もが、(経験的に測定されたときに)大体同じの色ウェイト(color weight)を持つように正規化された；  
 を持つ。

## 【 0 0 1 2 】

これらの理由から、エンコーダは、(他の色空間が使用され得るが)パイプラインに対して I C C - L a b の使用を採用した。フロント・エンド(エンコード)およびバック・エンド(デコード)パイプラインについて、内部色空間表現に関して以下のガイドラインが制定された：

- 1) 走査構成(configuration)：注記されない限り、( C E F L a b を含む)内部 x Y C C 表現は、クロマが、高速走査方向のみにおいて、2 x のファクター(factor of 2x)によってサブサンプルされることを想定する。更なる複雑さおよびストレージ・リクワイアメント(storage requirement)のために、パイプラインは、低速走査方向において、クロマをサブサンプルしない(サブサンプルによれば、若干、圧縮が増加するが)。もし、ソース画像が、走査線に沿って奇数の数の画素を持つならば、画素の数を偶数にするために、最後の画像列が複製される。  
 2) P D L 構成：内部 Y C C は、正確でノイズ・フリーの P D L データを表すので、いずれの方向にも、色空間のサブサンプリングは実行されない。

## 【 0 0 1 3 】

パイプラインは内部的に、Y C b C r または L A B のような、輝度クロミナンス色空間の色情報を維持する。L A B 色空間要素名は、S C C に関連するが、出力および走査パスの殆どは、実際には、L A B の代わりに Y C b C r 色空間を使用することに注意して欲しい。

## 【 0 0 1 4 】

別の意味に注記されない限り、画像データのクロミナンス要素は、高速走査方向(X C S S)に、2 のファクター(factor of 2)によってサブサンプルされることによって、X ク

10

20

30

40

50

ロマサブサンプリングと呼ばれる。色変換モジュール S C C 202は、X C S S サブサンプルされたデータを生成する、第 1 のモジュールである。サブサンプリング技術は、サブサンプリングの実行前に一緒に平均された 2 つの隣接画素からの (A , B) データを用いる。データが再構築されるべき時に、2 つの後続の画素に対して、平均された (A , B) 値が複製される。

【 0 0 1 5 】

走査色変換モジュール (Scan Color Conversion Module) S C C 202の詳細なブロック図が、図 2 に示される。走査色変換モジュール 202の構成は、3 つの 1 D T R C テーブル 302 に連なる (followed by) 3 × 3 のマトリックス乗算ユニット 304 を含む。走査色変換モジュール S C C 202は、入力として、スキャナまたは P D L ソースからのソース R G B 信号 200 を取る。入力されてくるスキャナ R G B データ 200 は、要素 (component) 当り、8 または 1 0 ビットの深さ (deep) で有り得る。スキャナデータは、束ねられる (packed) ことが、あまりあり得ないので、各 R G B 要素は、2 バイトを持つ。帯域幅に関しては、スキャナデータは、P D L データと比較して読込みに、2 倍多くの帯域を消費する。

10

【 0 0 1 6 】

走査色変換モジュール S C C 202 への入力は、完全な色の 8 ビットまたは 1 0 ビットの束ねられない R G B データ 200 である。8 ビット入力データの場合には、ソース信号は、R G B データの 3 バイトを正確に満たす。しかし、1 0 ビット入力データの場合には、3 つの 1 0 ビット R G B 量が、1 0 ビットの R G B 画素当り 4 バイトを持つ、3 2 ビットのダブルワードに束ねられる。

20

【 0 0 1 7 】

走査色変換モジュール S C C 202 からの出力 S C C 204 は、2 つの形式 (forms) の一つを想定する。通常の走査または P D L 動作では、サブサンプルされた交互の (Y, Cb, Y, Cr) のストリームは、各 Y C r の組が、画素当り一つの 1 6 ビット・ワード (2 バイト) の状態で生成される。第 2 の形式は、特別の高品質 P D L モードに対するものである。この場合には、クロマはサブサンプルされず、(Y, Cb, Cr) データは、画素当り 3 バイトを必要とする。いずれの場合でも、図 2 に示される 2 4 ビット表記は、出力 S C C 204 は、完全な (full) カラーの信号であることを示す。束ね及び/又はサブサンプリングのために、示されたビットの数は、必ずしも、ビットの実際の数を表さないことに留意して欲しい。

30

【 0 0 1 8 】

1 D の T R C テーブル・ユニット 302 は、3 つの独立の 1 D のルックアップ・テーブル (不図示) を含む。各テーブルは、1 0 ビットから 8 ビットへのマッピングを持ち、それゆえ、それぞれ、最大 1 0 2 4 の、1 バイトのエントリを持つ。T R C (Tone Reproduction Curve: トーン再製曲線) テーブルが生成され得、入力を任意に、出力にマップするために使用され得る。より有用なマッピングのいくつかには、ダイナミック・ストレッチ (dynamic stretch)、ガンマ曲線、および圧伸 (companding) の適用が含まれる。更に、スキャナ・カラーまたは暗さ / 明るさの調整、のようなユーザ制御を実施するために、T R C テーブルが利用される。

【 0 0 1 9 】

マトリックス乗算ユニット 304 は、以下の方程式によって 3 × 3 のマトリックス乗算を実行する。

40

【 数 1 】

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \frac{1}{M} \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} Y\_Bias \\ Cb\_Bias \\ Cr\_Bias \end{bmatrix} \quad (1)$$

ここで、入力 R G B 値は、符号無しの 8 ビットで、マトリックス係数 { A m n } は、乗算されるべき 1 2 ビットの符号付きの量である。そして、M は、正規化ファクタで、M = 2

50

^ S C C \_ M a t \_ S h i f tである。マトリックス乗算は、最小として、少なくとも20ビットの中間蓄積された精度 (intermediate accumulated precision) を必要とする。

【0020】

マトリックス乗算および右回りのシフトの後に、結果としての符号付きの ( Y , C b , C r ) 量は、0と255の間の全ての3値をクランプすることによって、通常の符号無しYCbCrコーディングに変換される。一般のマトリックス変換において、入力データは、乗算前に、エンコードされた符号無し量から、符号付きの量に調整される必要があり得る、ことに留意して欲しい。例えば、YCCからRGBへのマトリックス変換において、乗算前に、( 0 , - 1 2 8 , - 1 2 8 ) が、( Y , C , C ) 量に付加されねばならない。

【0021】

図3は、高速走査方向XCSSでの2×サブ・サンプリングの適用を示す。オペレーションは、2つの画素上で同時に実行される。スキャナ色変換が、入力RGBデータ200に適用された後に、ラスタ走査上のシーケンスとして、各々の変換された画素に対して ( L , A , B ) トリプレットが生成される。高速走査方向の、2×によるクロマのサブサンプリングは、後続の画素のクロマ値 ( A n , A n + 1 ) および ( B n , B n + 1 ) の組を独立に平均化することによって実現される。

【0022】

図3での最上部の帯310は、色変換ユニットから変換された画素の「ストリーム」を表す。そのパターンが異なった色を表すように充填された各画素は、それぞれ8ビットとして表現された ( L , A , B ) 値に対する出力値のトリプレットを含む。画素の間の境界は、サブサンプリングに対する後続の画素の組 (paring) を一緒にしるす (mark)。

【0023】

図3での底部の帯 (strip) 350は、クロマ・サブ・サンプリング後の出力画素の「ストリーム」を表す。「ストリーム」は、( L , A ) 値および ( L , B ) 値の一続きの交互の組から構成される。ここで、Lは、{ L } 値のオリジナルのシーケンスに対応し、交互の (alternating) ( A , B ) は、画素の組の平均化された量である。画素の組の「A」値は常に、「B」値に先行する。オリジナルの ( L , A , B ) トリプレットから、( L , A , L , B ) フォーマットへのマッピングは、図3の中央部分に示される。

【0024】

内部サブ・サンプリング・スキームの主要な利点は、帯域および記憶メモリの3から2への削減である。内部 ( L , A , B ) データに対して8ビット量を想定すると、オリジナルのデータは、画素当たり3バイトを必要とする一方、サブサンプルされた出力は、画素当たり2バイトだけを使用する。更に、サブサンプリング法は、それが、大きな品質の損失 (quality loss) 無しにクロマチャンネル ( A , B ) のサブサンプルを許容する点で、人間の視覚システムの特性に影響を与える (leverages)。低速走査方向でのクロマのサブサンプリングによっても更なるゲインが、得られ得るが、このオペレーションは、以前の画素のラインの記憶を必要とするため、実行のためにはよりコスト高である。それゆえ、内部LAB表示に対して低速走査サブサンプリングを使用しないことが決定された。

【0025】

同様に図4は、サブサンプルされたデータからの、オリジナルの (サブサンプルされない) 画素の再構築の方法を示す。本方法の実行は、まっすぐ (straightforward) である。画素の組からの平均化された ( A , B ) クロマ値は、複製されて、再構築された画素の各々とともに使用される。「L」値は、変更されない。

【0026】

再度図1を参照する。ここで、SCC出力204は、スクリーン評価モジュール (Screen Estimate Module : SEM) 216およびデ・スクリーン・モジュール (DSC) 206に送られる。更に、エンコード・パイプラインは、印刷中に生成されるPDL画像のような合成 (ノイズ・フリーの) 入力画像を取り扱える。この、特別のPDLモード236において、デ・スクリーン・ユニットは、バイパスされ、SCC出力204は、PDLセグメント・モジュール (SEG) 236 (これは、特別セグメンテーション・モードで作動する) に直接

10

20

30

40

50

進まされる。

【 0 0 2 7 】

スクリーン評価モジュール ( S E M ) 216は、ハーフトーン・スクリーン周波数および強度 (これらは後に、デ・スクリーン ( D S C ) 206およびセグメント ( S E G ) 234モジュールによって用いられる) を評価する。 S E Mモジュール216は、 S C C 202モジュールでの R G B から Y C C への変換の後に得られた入力スキャナデータの輝度チャンネル Y 上で作動する。 S E M 216モジュールは、異なった感度 (例えば、最大、高感度、低感度) を持つ、一つあるいはそれ以上の独立のチャンネルを使用する。最も高感度のチャンネルは、周波数評価 ( frequency estimate ) を駆動し、 2 つの他のより低感度のチャンネルは、スクリーン強度 ( screen magnitude ) を生成するために結合される。スクリーン評価モジュール216は、 2 つの (一つの構成要素の) 出力を提供する。それらは、スクリーン周波数評価 S c f 218と、スクリーン強度評価 S c m 220である。これらは、デ・スクリーン ( D S C ) 206およびセグメント ( S E G ) 234モジュールによって用いられる。スクリーン強度は、関心事である画素の近傍のローカル・エリアがハーフトーン・スクリーンであるという、確信レベル ( confidence level ) として解釈され得る。スクリーン評価モジュール216は、合成 ( ノイズフリーの ) 入力に対してバイパスされる。

10

【 0 0 2 8 】

デ・スクリーン ( D S C ) 206モジュールの目的は、走査された入力信号からハーフトーン・スクリーンを選択的に評価する一方、テキストまたはライン・アート・オブジェクト ( line art object ) のシャープなエッジ情報を維持または強調することである。デ・スクリーン・モジュール206は、走査色変換 ( S C C ) 202モジュールによって生成された変換された Y C C 画像上で作動する。このモジュールは、スクリーン評価モジュール216 ( S E M ) からの、評価されたスクリーン周波数 S c f 218および強度 ( magnitude ) S c m 220信号を使用する。デ・スクリーニングの方法は、動的に ( dynamically ) 制御されたフィルタ・バンク ( filter bank ) に基づき、複数のフィルターを通された出力の中での可変の混合 ( blending ) を提供する。本モジュールは、可変非シャープ・マスキングメカニズムを用いた、ポスト滲みシャープニング ( post-blur sharpening ) 、の独立の連続的制御をも含み、出力における画素のニュートラリティ ( neutrality ) を調整する能力を提供する。デ・スクリーン・モジュール206は、デ・スクリーンされた ( de-screened ) 出力信号 D S C 222を生成するとともに、更に、本モジュールは、統計モジュール ( S T S ) 210によって用いられる、入力信号の滲んだバージョン ( blurred version ) B L R 208をも生成する。

20

30

【 0 0 2 9 】

スクリーン・モジュール ( S C L ) 224は、いずれかの次元に ( dimension ) 、任意の総量だけ、入って来るデータをスケールリングする ( scales ) 。スケールリングの方法は、 2 D の 2 重線形補間 ( bi-linear interpolation ) である。いかなる、 (例えば削減のための) 事前フィルタリングを実行するに際しても、モジュール224は、デ・スクリーン・モジュール206に依存する。エンコード・パイプラインは、スケール・モジュール S C L 224の、 2 つの場合 ( instances ) を用いる。一つの場合では、デ・スクリーナ出力 D S C 206は、スケールリングされて、出力 D S S 226を生成する。第 2 の場合では、スクリーン評価モジュール216からの S c m 220信号はスケールリングされて、出力 S M S 230を生成する。スケールリングされたデ・スクリーナ出力 D S S 226は、全域強調モジュール ( Gamut Enhance Module : G M E ) 228によって強調される。結果としての、 G M E 232出力および S M S 230は、セグメント・モジュール ( S E G ) 234に届けられる。そして、 S M S 230がモノクロ (一つのコンポーネント) だけである一方、 D S S 226は、フル・カラー ( 3 - コンポーネント ) 信号であることに注意して欲しい。

40

【 0 0 3 0 】

図 5 は、スケール・モジュールの詳細なブロック図を示す。スケール・モジュール224は、一つあるいはそれ以上の入力カラー画像をスケールリングして、出力でスケールされた色画像を生成する責任を負う。各画像は、独立にスケールされる。スケール・モ

50

ジュール224は、必要な事前フィルタリングの総量を提供するに際して、先行するモジュールに依存するので、スケーリングの方法は、事前フィルタリング無しの、2重線形 (bilinear) 補間である。スケール・モジュール224は、高速および低速走査方向での異なる総量によって、ソース画像を独立にスケーリング出来る (アナモルフィックのスケーリング)。一般的に、帯域への考慮によって、各方向でのスケーリングは、1%増加 (increments) 程度において、4×削減 (25%) から4×拡大 (400%) の範囲に限定される。

#### 【0031】

フロントエンド画像パスにおいて、スケール・モジュール224の2つの場合が、独立に使用される。第1のスケール・モジュール402は、セグメンテーション化前に、デ・スクリーンされた色画像 D S C 222 をスケーリングするために用いられる。その結果、スケーリングされた色画像出力 D S S 226 が得られる。更に、第2のスケール・モジュール404が、スケーリングされたバージョン S M S 230 (これは、セグメンテーション・オペレーションのために必要とされる) を生成するために、モノクロのスクリーン強度 (magnitude) 信号 S C M 220 をスケーリングするために用いられる。スケール・モジュール402、404の双方が、それぞれ、同じ総量のスケーリングを、それらの対応するソース画像に加える (apply) ことに留意して欲しい。

#### 【0032】

フロントエンド・パイプラインにおいて、削減 (reduction) のために、2つのスケーリング・ユニットが主に用いられることが期待される。その理由は、前景カラーおよび背景データをスケーリング・ダウンすることによって、C E F のサイズが最小化されるためである。同様に、拡大の場合には、背景端部 (back-end) においてデータを拡大することによって、圧縮された C E F サイズを再度最小化することは、道理に適ったことになる。それゆえ、何等かのフロント・エンドスケーリング・ユニットが実際に、拡大のために使用される、ことは期待されない。しかし、画像をフロントエンドで拡大することが望まれる場合に、(例えば、走査してエクスポートすることに対して) いくつかの状況が存在し得る。もしそれが必要であれば、その目的のために、2つのスケーリング・モジュールもまた、利用され得る。

#### 【0033】

第1のスケール・モジュール402への入力、クロマ (A, B) 信号が一般的に、(輝度チャンネルに対して) 高速走査方向においてのみ、2×のファクターによってサブサンプルされた状態での、フル・カラー (L, A, B) ソース信号 D S C 222 である。第1のスケール・モジュール D S S 226 の出力は、入力ソース信号 D S C 222 の、(より小さく、またはより大きく) スケーリングされたバージョンである。出力信号のフォーマットもまた、クロマ (A, B) が、高速走査方向においてのみ2×のファクターでサブサンプルされた、フル・カラー (L, A, B) 信号であるという点で、出力信号のフォーマットは、入力に類似する。

#### 【0034】

第2のスケール・モジュール404インスタンス (instance) は、入力と出力信号 S C M 220 および S M S 230 がモノクロ (L\*チャンネルのみ) である点を除いて、動作上、第1のスケール・モジュール402に類似する。それゆえ、スケール・モジュール・インスタンス S M S 230 の記述が、クロマ部分が除外された、スケール・モジュール・インスタンス D S S 226 の輝度部分から引き続く (follow) ことになる。

#### 【0035】

高速および低速走査方向でのスケーリングの総量は、図6に示されるように、固定整数 410、およびフラクション表記 (fraction notation) 412 を用いて表現される。定数 ScIsrc\_Sft412 は、フラクション部分 (現在13ビットに設定されている) に割り当てられたビットの数を規定する。16ビット表現を用いて、要求される4×の削減範囲をカバーするために、これは、整数部分410に対して、少なくとも3ビットを与える。更に、13ビット・フラクションは、トータルの蓄積されたポジショニング・エラーが、レターサイズの

10

20

30

40

50

ページの幅に亘って約 1 画素より小さいことを保証する。

【 0 0 3 6 】

上述の表現を用いて、低速および高速走査スケーリングの総量はそれぞれ、16ビットのレジスタScale\_XおよびScale\_Yを用いて、事前に規定される（そして、ページに亘って固定される）。各方向に、同じ量だけ入力画像をスケーリングするために、Scale\_XおよびScale\_Yの双方が、同じ適切な数で充填（loaded）されねばならないことに留意して欲しい。図6は、有効なScale\_XおよびScale\_Y値に対する複数の例、420、430および440を含む。

【 0 0 3 7 】

アルゴリズムの初期化フェーズ中に、図7のテーブル450に示されるように、16ビット変数が初期化される。これらの変数は、スケール・モジュールのオペレーションを通じて、使用するために便利である。別異に注記されない限り、全ての変数は、16ビット・ワードを使用することに留意して欲しい。変数のいくつかは、図6に示されるように、固定整数/フラクシオン注記（fixed integer/fraction notation）を用いる。

【 0 0 3 8 】

スケール・モジュール224のオペレーションが、図8に示される。（図8の黒三角506で示されるように、）第1に、次の出力画素の位置を計算するために、（X，Y）ソース・ステップ・ファクターStpSrc\_XおよびStpSrc\_Yが用いられる。この位置は次に、ターゲット目標画素506を取り囲む、4つの最も近いソース画素502を識別する（locate）ために用いられる。2Dの二重線形補間アルゴリズムが次に、4つの隣接した値に基づいて、目標画素508の値を計算（補間）するために用いられる。この工程は、一般的に、4つの新規のソース画素を用いて、全ての目標画素が出力されるまで、各目標画素について反復される。

【 0 0 3 9 】

50%削減または拡大より大きい場合には、時々、2つまたは4つの目標画素は、同じ4つのソース画素が1回以上使用されるように、同じ影付きエリア内に配置され（located）得ることに留意して欲しい。

【 0 0 4 0 】

各目標画素の位置は、2つの32ビット・レジスタ（yPosおよびxPos）内に維持される。各レジスタは、19ビットの整数部分と13ビットの（Src\_Sft）フラクシオン部分を持つ。レジスタ値は、原点がページの左上隅にあるような、ソース画素ユニット内にある。yPosとxPosの整数部分は、図8および9の影付きエリアの左上隅における、ソース画素604の行と列インデックスである。yPosとxPosのフラクシオン部分は、目標画素の、左上隅の参照画素からの下および右へのオフセットを表す。これらのフラクシオンは、補間重み（interpolation weight）として用いられる。

【 0 0 4 1 】

各目標画素に対して、影付きエリアに亘っての2Dの二重線形補間が、3つの1D線形補間ステップのシーケンスを用いて実行される。一般性の損失無しに、垂直（低速走査）補間が最初に起こることを想定する。3つの補間ステップは次に、2つの垂直（低速走査）の1Dのステップ、および後続の一つの1Dの水平（高速走査）ステップ、から構成されることになる。図9を参照する。ここで、第1の垂直ステップにおいて、2つの左画素604は、垂直に補間され、目標ラインに沿った値を計算する。第2のステップにおいて、右の画素の組606は、垂直に補間され、目標ラインに沿って後続の値を計算する。第3および最終のステップにおいて、2つの前の結果608が次に、水平に補間され、目標画素自身での値を計算する。3つの1Dの補間ステップの詳細は、更に以下に説明される。

【 0 0 4 2 】

スケーリング・オペレーションは、ソース画像の左上の画素（これは、現在のソース画像位置レジスタ（xPos，yPos）が（0，0）に初期化された状態で、スケーリングの原点として参照される）から開始する。

【 0 0 4 3 】

10

20

30

40

50

ここを開始点として、出力画素は、4つの隣接ソース画素を用いて、各ステップにおいて生成される。Xでのステッピング (stepping) は、StpSrc\_XをxPosに追加することによって実現される。Yでのステッピングは、yPosにStepSrc\_Yを追加することによって実現される。ラスタ走査順序 (order) に対して、yPosが増加された時に、xPosは、0 にリセットされる。スケーリング・ファクターおよびページ上での位置に応じて、補間エリアを取り囲む4つのソース画素は、以前のステップとは完全に異なり得るか、または、完全に同じであり得る。

【0044】

各レジスタの整数ポートは、ScIsrc\_Sftオペレーションの右シフトとともに抽出される。各レジスタのフラクショナル部分は、 $((1 << \text{ScIsrc\_Sft}) - 1)$  オペレーションのマスクとともに抽出される。フルの13ビット・フラクショナルを含む、大きなページ・ディメンションを十分に実現するために、(以下に説明される) yPosとxPosの双方が、32ビットのダブル・ワードである必要があることに留意して欲しい。図9は、ソース・グリッド(その上で補間が行われる)の上に(図8の影付きエリア)の2Dの矩形を形成する4つのソース画素を示す。

【0045】

図9に示されるように、2Dの2重線形の補間自身は、3つの1Dの線形補間ステップから構成される。このステップにおいて、左画素の組は、垂直フラクショナルを用いて、垂直方向で補間される。この計算は、以下の式を用いる。

$$\text{Left} = \text{pix}_{\text{TL}} + \{ [(\text{pix}_{\text{BL}} - \text{pix}_{\text{TL}}) * \text{fraction\_Y}] \gg \text{StpSrc\_Sft} \}; \quad (2)$$

ここで、 $(\text{pix}_{\text{BL}} - \text{pix}_{\text{TL}})$  はそれぞれ、ベースおよび後続のソース・ライン上での、最左上および最左下の画素である。同様に第2のステップにおいて、右の画素の組が、以下の式に従って、垂直フラクショナルを用いて、垂直方向で、次に補間される。

$$\text{Right} = \text{pix}_{\text{TR}} + \{ [(\text{pix}_{\text{BR}} - \text{pix}_{\text{TR}}) * \text{fraction\_Y}] \gg \text{StpSrc\_Sft} \}; \quad (3)$$

ここで、 $(\text{pix}_{\text{BR}} - \text{pix}_{\text{TR}})$  はそれぞれ、ベースおよび後続のソース・ラインの上の、最右上および最右下の画素である。式(2)および(3)で、出力ラインに沿った、補間された左および右の値を計算した後に、2つの結果は、高速走査方向で補間され、最終結果が生成され得る。第3の補間フェーズは、以下の式による。

$$\text{Value} = \text{Left} + \{ [(\text{Right} - \text{Left}) * \text{fraction\_X}] \gg \text{StpSrc\_Sft} \}; \quad (4)$$

【0046】

輝度チャンネルの場合には、式(4)の結果は、出力に直接送られる最終結果である。輝度値は、各目標画素に対して出力される。

$$\text{L\_out} = \text{Value}; \quad (5)$$

【0047】

論理的には、クロマ・コンポーネントは、ただ、Lとして計算され、後に、各出力の組が平均され、Xクロマ・サブ・サンプルされた形式に戻る。

【0048】

クロマ画素計算は、クロマ・コンポーネントが、高速走査方向において2xのファクターによってサブサンプルされた、という事実に影響を与える (leveraging) ことによって削減され得る。クロマX補間が必要とされないこととなる可能性は50/50である。このことを利用するために、x補間は、最初に為される必要がある。

【0049】

再度図1に戻る。ここで、全域強調モジュール(GME)232は、入力画像のカラー・コンポーネントの各々に、3つの独立のトーン再製曲線を適用する。この実行は、3つの独

10

20

30

40

50

立で完全にプログラマブルの1Dのルックアップ・テーブルを介して為される。全域強調モジュール232への入力、スケーリングされ、デ・スクリーンされた、ソース画像のバージョン、を表す、スケーリング・モジュール(SCL)224からの出力DSS226である。出力は、全域マッピング(gamut-mapped)された信号GME232である。

【0050】

全域強調モジュールGME228のブロック図が、図10に示される。入力信号DSS226は、3つの独立の8ビットの1Dのルックアップ・テーブル(このそれぞれは、別個の色成分(color components)で作動する)を通過させられる。次のブロック290は、各画素の色成分のニュートライジング(neutralizing)/維持、のオプションを持つ。各画素の輝度値が0または255と等しい、当該各画素の色距離(color distance)が、最初に計算される。もし画素の色距離値が、所定の閾値総量(threshold amount)より大きいならば、その輝度値は、255より小さいプリセットされた値(例示の目的のみでいえば、例えば254)に設定される。もし画素の色距離値が、所定の閾値総量(threshold amount)以下ならば、画素の色成分は、ニュートラルの値に設定される(a=b=128)。これらの調整の目的は、非常に色が強い画素は、純粋な白ではあり得ないこと、および、白画素はニュートラル(クロマ無し)でなければならないことを保証することである。

【0051】

エンコーダ・パイプライン全域強調モジュールGME228は、入力として、スケーリング・モジュールSCL224からの、フル・カラーの、デ・スクリーンされ、スケーリングされた、信号DSS226を取り、フル・カラーの全域強調された信号GME232を生成する。入力信号SCLと出力信号GMEの双方は、一般的に、高速走査方向のみに、2xのファクターでサブサンプルされたフル・カラーのLab信号である。(a,b)クロマ・チャンネルの束ね(packaging)およびサブ・サンプリングのために、SCLまたはGMEの下に示されるビット数(24)は、必ずしも、実際のビット数を表さないことに留意して欲しい。実際のLabデータは、画素当る16ビットで、交互の(L,a,L,b)の画素の組として束ねられる

【0052】

再度図1を参照する。ここで、セグメント・モジュール234の目的は、入って来る画像を、3つのMRC層、つまり、前景240、セクタ248、および、背景238平面、にセグメント化することである。合成(ノイズ・フリーの)画像のために、別個のセグメント化のモードもまた、提供される。モジュール234は、全域強調モジュール228からの全域強調された色画像GME232の上で作動する。このモジュールは、スクリーン評価モジュール(SEM)216からの、スクリーン強度評価された(magnitude estimated)信号SCM220をも使用する。SEGモジュール234は、前景および背景平面に対応する、2つのフルカラーの画像FGD240およびBGD238、および、セクタ平面のための一つのバイナリの出力SEL248画像を出力する。更に、スクリーン評価モジュール234は、合成(ノイズ・フリーの)画像の特別の場合に対して、もし利用可能であれば、入力して来る、PDLヒント(hints)250を維持出来る。入力して来るRhtヒント(hints)260は、CEFヒント(hints)平面Hntにマッピングされる。

【0053】

ルックアップ・テーブル(LUT)モジュール242は、圧縮前に、カラーの前景および背景データを、YCCから、装置独立(device-independent)のLab色空間に変換する。変換の方法は、変化する2のべき乗の節点スペーシングを持つ、3Dの四面体の補間を使用している。エンコード・パイプラインは、2つの別個の、LUTモジュール242のインスタンス(instances)を用いる。一つの場合では、セグメント・モジュール(SEG)234によって生成された前景入力FGD240は、YCCから、Labに変換される。第2の場合では、セグメント・モジュール234からの背景入力BGD238、YCCからLabに変換される。第3のLUTインスタンスもまた、デコード・パイプライン(不図示)で使用されて、Labからのマージされたデータを、装置CMYKに変換する(3D→4D変換)。

【0054】

10

20

30

40

50

より詳細には、LUTカラー・ルックアップ・テーブル・モジュール242は、内部YCbCr色空間から、最終CEFLab表現に変換する。この変換は、3Dのルックアップ・テーブルを用いて実現される。しかし、それは、セグメンテーションの後に適用され、データは既に、3つのMRC平面に分離されているので、各色平面は、独立に変換されなければならない。バイナリのセレクト平面は全く、色情報を運ばないので、変換を必要としない。しかし、前景および背景平面は、独立に、変換されなければならない。変換されるべき2つの平面が存在するが、それらは一般的に、オリジナルの走査に対して各方向で、少なくとも2×によってサブサンプルされるので、処理の総量は、依然として、分割する(splitting)前に、オリジナルの走査を変換するために必要とされるであろうものより少ない。

10

## 【0055】

カラー・ルックアップ・テーブル・モジュールLUT242は、入力として、ソースYCbCr色信号238を取り(takes)、与えられた3DのLUTを用いて、それを、LABに変換する。カラー・ルックアップ・テーブル・モジュールの2つの場合が用いられる。一つは、ソース前景SEG\_Fgdを変換して、Fgdを生成することであり、第2は、背景SEG\_Bgdを、Bgdに変換することである。双方の場合で、同じLUTテーブルが使用されることに留意して欲しい。

## 【0056】

3DのLUTは、可変(2のべき乗)のスペーシングで、最大17の格子状の節点(cubed nodes)をサポートする。可変節点スペーシングの特徴についての更なる情報は、本出願で、後に説明される。テーブルの節点は、ダウンロード可能である。3Dのテーブルだけが、いかなる与えられた時刻においても常駐する。前景または背景のいずれに対しても、同じテーブル構造が使用される。

20

## 【0057】

LUTモジュール242は、単純で効率的な、一つあるいはそれ以上の次元での、内挿または外挿の作動の実施の方法を記述する。現在の技術では、節点値のためのテーブルを用いた後に内挿段階があり、一般的に節点が、均一にスペーシングされることを必要とする。この出願では、ルックアップ・テーブルでの選択された節点位置が、任意の2のべき乗の節点スペーシングでプログラム可能であることを可能とする、新規の方法が説明される。この新規の方法は、現存する技術についての非常に少ない追加の計算しか必要としない。そして、それは、同じ数の節点に対する、非常に優れた内挿補間品質を提供出来る。本新規の方法は、シンプルで、フレキシブルで、ハードウェア・アーキテクチャに関わらず不変である。本新規の方法は、ソフトウェアで実施されており、デモンストレートされ、(従来の固定線形節点スペーシング・アーキテクチャに較べて)色空間変換の応用に対するカラー忠実度における大きな改善を示した。SCC(第2の色空間変換)の応用において、実験は、提案された方法が、従来の方法に対して、殆ど2倍の(factor of almost two)、色補間の正確さを改善することを示した(メートル法によるカラー・エラー(color error metric) Eによる測定)。

30

## 【0058】

多次元補間に対する必要性が、信号及び画像処理の多くの分野で生じているが、そのような分野では、[多次元]空間内の点の値を、一組の他の点(ノード)の既知の値に基づいて求めることが必要である。多くのアプリケーションでは、ノードは、便宜上、固定された、均等に離間したグリッド位置で測定される。非線形空間の補間精度が、[同数の]ノードが、必ずしも規則的なグリッド位置上に制限されていない任意の位置に対して最適化される場合に著しく上がることは周知である。しかしながら、任意のノード位置を許容するには計算上の複雑性が増すことから、このようなスキームを実際のアプリケーションで使用することは、通常は禁止されている。

40

## 【0059】

従って、ノード間の整数[非正規化]距離が2の二進乗数となるように制限することにより、任意のノード位置に付帯する付加的な複雑性を低減する1つの効果的な方法が提案

50

されている。この新しい方法は、単純で実施が容易であり、必要なのは小さなルックアップ・テーブルを1つ追加するだけである。

#### 【0060】

この新しい方法は、とりわけ色空間変換に対する多次元補間の領域に適用可能である。現代のデジタル情報の時代では、カラー文書は、通常、RGBで走査されCMYKで印刷される。従って、データは、印刷処理の一環として一方の色空間から他方の色空間に（即ち、RGBからCMYKに）変換せねばならない。大抵のCMYK印刷処置はマーキング装置の属性並びに複雑な多色トナー又はインクの相互作用により非線形性が高いことは、広く知られている。本実施形態、及び別の事例においても、YCCデータをLab色空間に変換する際にはLUTモジュール242が使用される。しかしながら、ここで、この基本的な技術は、他の多くの信号及び画像処理分野にも共通して適用可能であり、色空間変換の具体的な事例は、ここでは本方法を説明することを目的として使用しているに過ぎない点を強調しておきたい。

10

#### 【0061】

色空間変換の問題を取り扱う1つの方法は、所与の入力RGB又はYCC値を最良に再生するために、特定のマーキング・エンジンに適用すべき好適なCMYK又はLab値をテーブル内に記憶することである。テーブルのコンテンツは、試行錯誤的実験により確定されることになる。そのようなテーブルを構築できれば、CMYK又はLab値をリアルタイムで探索することは簡単なことになる。しかしながら、このようなテーブルは、可能なRGB配色が何百万と存在するのでテーブル全体が大きくなりすぎることから、殆ど実現性がない。例えば、RGBおよびCMYKの表現は大抵は8ビット/色であるが、これではテーブル毎に $2^{24}$ バイトを保有するテーブルが4つ必要になり（CMYK各色につき1テーブル）、この数は、10ビットのRGB入力に対して $2^{30}$ にまで増加する。

20

#### 【0062】

サイズの問題を解決するための一般的なアプローチは、ノードに対しては粗い3-Dテーブルを使用し、その後細かい3-D補間段階を続けるというものである。ノードの数（品質により異なるが、通常、9、17、又は33）は、可能な色数よりもずっと少ないので、テーブルのサイズは管理可能な大きさに留まる。テーブルは、現下のRGB又はYCC点に最も近いノードに対して所望のCMYK又はLab値を探索するために使用される。次いで、空間は隣接するノード間では区分的線形に近似しているとの仮定の下で、補間段階を用いて、最も近いノードのCMYK又はLab値に基づき、（推定される）出力CMYK又はLab値を計算する。近似の程度は、無論、ノードの個数、その相対位置、及び色変換の非線形性により決まる。YCC対Labの事例も同様のやり方に従う。わかり易くするため、以下ではRGB対CMYKの事例だけについて論じる。

30

#### 【0063】

なお、文献では、数種類の補間スキームが提案されている。自然な選択は、周知の多線形補間スキームであるが、より効果的に実施される多面体補間スキームのような他の技術もある。違いを説明するために、上記の3-Dの場合を考察する。トリ-リニアの場合、先ず最も近いノードを8個識別し（3D空間内にある対象の所与の点を包含する立方体の角部とするのが最も分かり易い）、それらのCMYK値をテーブルで探索する。CMYK値を、線形に（直接又は対のシーケンスの何れかで）補間して、所与の点から各ノードまでの比距離に基づいて推定値を求める。代わりに、四面体補間スキームでは、立方体は更に多数の四面体（非ゼロ体積を持つ最小ノード数オブジェクト）に分割される。第1の段階は、上記四面体のうちのどの四面体に所与の点が存在するかを求めることである。特定の四面体の角部に対応する4つのノードのCMYK値しか、テーブルで探索する必要はない（トリ-リニアの場合に対して、テーブル探索の帯幅/個数は半分となる）。最終的に、CMYK値は（直接又は対のシーケンスの何れかで）線形に補間され、所与の点から四面体各角部までの比距離に基づく推定値が求められる。なお、最終的な補間フェーズを実施するのに必要な乗算器/加算器の個数も、トリ-リニアの場合に必要な個数よりも少なくなる。従って、四面体方法は遙かに効率的に実施することができる。なお、本書で

40

50

提示する方法は、トリ - リニア又は四面体を含め多くのタイプの補間スキームに適用可能であるという点が重要である。

【 0 0 6 4 】

何次元であっても、区分的線形近似による多次元補間段階は、カスケードシーケンスの何回かの1次元補間に減少させることができる。例えば、トリ - リニアの場合、補間は、図 1 2 に示すように、先ず次元の1つ702に沿って対で補間を進める(4回の1 - D線形補間を要する)ことができる。次いで、求められた4つの値を対で、他の1つの次元706に使用する(更に2回の1 - D線形補間を要する)。最後に、求められた2つの値を、第3の次元に沿って補間(更に1回1 - D補間)して、最終的な結果を出す。

【 0 0 6 5 】

従って、トリ - リニア補間は、合計7回の1 - D線形補間を必要とするが、これを3段階シーケンスで実施している。多次元補間がコ - リニアである限り、この結果は真であり、最初に使用する次元の順序は結果に影響しない。

【 0 0 6 6 】

上記の一般的方法(テーブル + 補間)は、色空間非線形性は複雑すぎる場合がしばしばで、単純なモデルで容易に記述できないため、実際には、広く実用されている。従来のアプローチは、テーブルのノードを、固定された均一のグリッド上で測定することに限定したものであった。ノード間の距離が等しいと仮定すれば、実施は単純化するが、実際に正当化するのには難しい。視覚的により重要な空間の領域に、又は非線形性の度合いが高い領域に、より多くのノードを割り付けることが望ましく、そうしなければ区分的線形近似が失敗することは明らかである。固定ノードスキームの1つの利点は、ベースノードまでの比距離を計算するのが非常に容易であり、立方体又は特定の四面体の体積が空間のどこにおいても一定であるということである。

【 0 0 6 7 】

他方、従来のスキームでは、ノードを全く任意の様式で位置決めできるようにすれば、複雑性が著しく増す。ノード間の距離が変化することによって複雑性が付け加わり、立方体又は四面体のサイズが異なることを考慮に入れるために、多次元正規化オペレーション(即ち、除算)が必要となる。このような除算は実施に費用の掛かることは周知である。

【 0 0 6 8 】

ノード間の整数(正規化されていない)距離は2の二進乗数であるという条件の下で、ノード間の距離を任意にできるようにすることが提案されている。この思想の動機は、2の整数乗数による除算は単純な桁操作として実施できるということにある。2の二進乗数システムは、小さな段階と大きな段階を同時に記述する場合に便利な対数スケールであることから、この付加的制約はあまり拘束的ではない。更に、この方法は、所望の再生忠実に合致するようにノード位置を選定するに際し完全な柔軟性を維持している。つまり、均一グリッドの場合に比較して、通常、必要なノードの数は少ない。加えて、実施が簡単なので、付帯費用(例えば、ASIC設計のシリコンゲート、及び計算の複雑性)を大幅に節約できる。

【 0 0 6 9 】

上記ルックアップ・テーブルは、入力ドメインから出力ドメインへの非線形マッピングを含んでいる。色空間変換の例では、入力ドメインは(アプリケーションによっては、YCbCr又はCIELABのようなデバイス非依存性の強い中間空間を使用の方が望ましいこともあるが)RGBであり、出力ドメインはデバイス(マーキング・エンジン)CMYKである。非線形マッピング関数は、一般的に知られていないが、又は直截的なやり方では実施が複雑すぎるので、LUTのコンテンツは、通常はキャリブレーション処理の一部として実験により求められる。一般的には、LUTが与えられると、基本的1 - D補間段階は、(図 1 3)のように表現することができる。

$$V_0 = V[i] + \frac{1}{2} \cdot (V[i+1] - V[i]) \quad (6)$$

10

20

30

40

50

$$d_i = x - \text{node}[i] \quad (7)$$

$$d_i - \text{node}[i+1] - \text{node}[i] \quad (8)$$

ここに、 $x$  は入力点、 $V[i+1]$  及び  $V[i]$  は、それぞれノード  $\text{node}[i+1]$  と  $\text{node}[i]$  における値である。 $i$  および  $i+1$  はノード・インデクスである。 $d_i$  は、点  $x$  からベースノード  $i$  までの距離であり、 $d_i - \text{node}[i+1] - \text{node}[i]$  はノード  $i$  と  $i+1$  の間の距離であり、 $V_0$  は、入力点  $x$  における補間値である。

【0070】

数式(6)の補間は以下の3つの主要な演算を含んでいる。

10

1) ベースノード検索：この演算は、 $[\text{node}[i] < x < \text{node}[i+1]]$  となるように、ベースノード  $\text{node}[i]$  を検索するための比較論理を含んでいる。

2) 分数計算：この演算は、数式(6)の正規化された距離分数  $d_i / (d_i - \text{node}[i+1] - \text{node}[i])$  を計算するが、ハードウェアで実施するためには費用のかかる全体除算を必要とする。

3) 補間：数式(6)の最終的な補間値  $V_0$  を計算するのに、乗算1回、加算(減算)2回が必要である。

【0071】

従来の単純な固定線形ノード・スペーシングでは、何れの2つのノード間のスペースも一定であり、即ち、

20

$$\Delta_i = \Delta = \text{Const.} \quad \forall_i$$

である。一定のスペーシングとすれば、因数  $d_i / (d_i - \text{node}[i+1] - \text{node}[i])$  は既知なので、正規化因数として予め計算しておくことができる。しかしながら、等しいスペーシングという制約は、上に述べたように、高品質を達成するために色空間内のノードの分布を変更するための柔軟性を提供することができない。他方、完全プログラム可能ノード・スペーシングは、最大限の柔軟性を維持するが、上記のように、任意除算を計算する必要があるため、実施には費用がかかり複雑である。

先の技術の柔軟性(品質)と単純性(費用)に取り組むために、本書は、下記2の乗数ノード・スペーシングを使ったプログラム可能ノード・アーキテクチャを提案している。

30

【数2】

$$\Delta_i = \text{node}[i+1] - \text{node}[i] = 2^{k_i}, \quad k_i \in N, \quad (9)$$

ここで、 $N$  は自然数のセットであり、 $k$  は各ノード対に対する異なる整数定数である。2の乗数を用いると、一般的な除算は、桁操作で行うことができ、実施が簡単である。つまり、

$$d_i / (d_i - \text{node}[i+1] - \text{node}[i]) = \dots > k_i \quad (10)$$

【0072】

40

図14は、カラー・ルックアップ・テーブルの処理の実施形態730のフローチャートである。既知ノードのデータベースが形成される。既知ノードは、何れの隣接する既知ノード間の距離も2の整数乗数となるように選択される(ブロック732)。所与の入力ノード  $x$  について、入力ノード  $x$  がノード1と隣接するノード2の間になるようにベースノードのノード1を求めてデータベースが検索される(ブロック734)。ノード2とノード1との差は  $2^k$  である。ノード1とノード2の関数値は、それぞれ  $V(\text{ノード1})$  と  $V(\text{ノード2})$  である(データベースに記憶されている)。ノード1は、補間が関数値  $V(\text{ノード1})$  から始まることになるのでベースノードと呼ばれる。入力ノード  $x$  とノード1との差が計算される(ブロック735)。値  $d_i / (d_i - \text{node}[i+1] - \text{node}[i])$  は、論理的に  $k$  位置だけ右方向にシフトされるが、 $k$  は、ベースノードのノード1と第2ノードのノード2の間の距離の2を底とする対数

50

である。(ブロック736)。関数値  $V(x)$  は、シフトされた に、 $V(\text{ノード}2)$  と  $V(\text{ノード}1)$  の間の差を掛けることにより得られた積と、 $V(\text{ノード}1)$  を組み合わせることにより計算される(ブロック738)。

【0073】

ベースノードの選択は、一般的には、入力色空間の起点で決まる。入力色空間が起点0を有するアプリケーション(例えば、R、G、Bの範囲が0から255までのRGB色空間)では、ベースノードは、通常、入力ノードよりも小さい正の値になるように選択される。この場合、 $V(\text{ノード}2) - V(\text{ノード}1)$  は正の値である。入力色空間の起点が0ではなく中間範囲のどこかであるアプリケーション(例えば、 $(a, b)$  又は  $(c_b, c_r)$  の範囲が-128から+127までのCIE-Lab空間又はYCbCr色空間)(例えば、図18参照)では、ベースノードは、通常、入力ノード  $x$  が起点よりも大きい場合にはその入力ノード  $x$  より小さい正の値になるように、入力ノード  $x$  が起点よりも小さい場合にはその入力ノード  $x$  より大きい正の値になるように選択される。この場合、 $V(\text{ノード}2) - V(\text{ノード}1)$  は、起点よりも大きい入力ノード  $x$  に対しては正であり、起点よりも小さい入力ノード  $x$  に対しては負となる。

10

【0074】

殆どのアプリケーションにおいて、ノード間指数とも呼ばれる  $k$  値は、予め計算してデータベースに記憶されている。アプリケーションによっては、必要に応じ  $k$  値をリアルタイムで計算するものもある。

【0075】

20

以下の例は、RGB対CMYK色空間変換アプリケーションで、データベースがどのように形成されるかを示している。まず、任意のRGBノードとそれに対応する好適なCMYK値とのテーブルが生成される。好適なCMYK値は、特定のマーキング・エンジンに適用した場合に、対応するRGB値を最適に再生する値である。テーブルのコンテンツは、マーキング・エンジンの出力上の或る点の実際の色値を測定する高精度の器具を使用して、試行錯誤的実験法で求めることができる。テーブルが生成された後、テーブルのコンテンツに(上記)四面体法などの補間スキームを使用して、選択されたRGB点のセットに対するCMYK値を得る。RGB点は、何れの2つの隣接する選択されたRGB点の間のスペーシングも、2の整数乗数になるように選択される。次いで、これら選択されたRGBノードとそれに対応するCMYK値だけを使って、データベースが形成される。

30

【0076】

図15は、カラー・ルックアップ・テーブル用のシステムの実施形態740のブロック図である。システム740は、データベース742と、検索モジュール744と、計算モジュール750を備えている。計算モジュール750は、結合モジュール752と、シフト・モジュール754と、乗算モジュール756を備えている。

【0077】

データベース742は、既知ノードと、対応する既知ノードの関数値と、ノード間指数のリストを記憶している。既知ノードは、何れの2つの隣接する既知ノードの間の距離も2の整数乗数になるように配置されている。リストの境界に位置していない既知ノードは、それぞれ、対応する既知ノードとそれぞれの隣する既知ノードの間のそれぞれの距離の、2を底とする対数を表す2つのノード間指数と関係付けられている。

40

【0078】

検索モジュール744は、入力ノードがベースノードとベースノードに隣接する第2ノードとの間に位置するようなベースノードを求めて、データベースを検索する。検索方向は、入力ノードの上方であれ下方であれ、起点の位置に基づいてプログラム可能である。

【0079】

結合モジュール752は、入力ノードとベースノードの間の差 と、第2ノードの関数値とベースノードの関数値の間の差  $V$  を計算する。シフト・モジュール754は、 を  $k$  位置だけ右方向にシフトするが、 $k$  は、第2ノードに関しベースノードに関係付けられたノード間指数である。乗算モジュール756は、シフトされた に差  $V$  を掛けて、積の値を

50

出力する。結合モジュール752は、ベースノード関数値と積の値を結合して、入力ノードの関数値  $V(x)$  を形成する。

【 0 0 8 0 】

計算モジュール750は、図 1 5 に示すようにデータベースと通信する。これは、検索モジュール744が、計算モジュール750に、検索で得たデータを全ては送らない場合である。例えば、検索モジュール744は、ベースノードのノード・インデクスしか送らないこともある。計算モジュールは、このノード・インデクスを使って、データベース742内の他のデータを探索する。検索モジュールが必要な全データを計算モジュール750に送る別の実施形態では、計算モジュール750は、データベース742にアクセスする必要はない。

第 1 の実施形態では、データベース742は、既知のノードに対応するノード・インデクスと、既知のノードと、対応するノード関数値と、ノード間指数とを含む 1 つのルックアップ・テーブルとを備えている。

10

第 2 の実施形態では、ノード間指数は、ルックアップ・テーブルに記憶されておらず、必要に応じて計算される。

第 3 の実施形態では、データベース742は、主ルックアップ・テーブルと、予備のルックアップ・テーブルを含んでいる。主ルックアップ・テーブルは、既知のノードに対応するノード指数と、対応するノード関数値を含んでいる。予備のルックアップ・テーブルは、既知のノードに対応するノード指数と、既知のノードと、ノード間指数を含んでいる。

【 0 0 8 1 】

新しいアーキテクチャを使えば、ノード・インデクスと、ノード値と、ノード間指数と、出力値を保有する L U T を構築することができる。指数は、任意の隣接する 2 つのノード間の距離の、2 を底とする対数である。即ち、

20

$$\text{exponent}[i] = \log_2(\text{node}[i+1] - \text{node}[i]) \quad (11)$$

一例を図 1 6 に示す。

【 0 0 8 2 】

所与の入力  $x$  に対して、補間出力  $V_0$  を計算することができる。なお、二等分アルゴリズムは、ベースノードを見つける際に検索エンジンとして使用することのできる多くの可能な検索アルゴリズムの中の 1 つに過ぎない。理論的には、提案しているアーキテクチャは、何れのビット精度入力対何れのビット精度出力をもマップすることができる。入力と出力は、L U T 内で作表されるので、どの様なビット精度の表現も表内に収容することができる。実行時には、ビット精度は、信号(色)を表現するために使用されるビットの数と実施費用とに基づいて選定される。提案のアーキテクチャは、ビット精度の選定に如何なる制限も設けていない。図 1 7 は、図 1 6 の 8 ビット入力に替わり、10 ビット入力を採用した L U T の例を示している。

30

【 0 0 8 3 】

図 1 6 は、起点近くに高密度のサンプルが必要な色空間変換の例を示す。ノードスペーシングは、起点から遠いノードより起点に近いノードの方が密になっている点に留意されたい。(実行時、平方根規則を使用することの多い) R G B 入力色空間で作業する場合はこの種のノード分布が望ましいことが分かっている。場合によっては、システムは、異なる入力色空間、例えば、C I E L\* a\* b\*、F a x L\* a\* b\*などをサポートする必要がある。C I E L\* a\* b\*の場合、a\*とb\*チャンネルは、起点が128である。他方、F a x L\* a\* b\*は、a\*チャンネルは起点が128で、b\*チャンネルは起点が96であると仮定している。提案のアーキテクチャは、非常に柔軟性があり、インプリメンテーションを変えずに、各種の起点差及び必要性を容易にサポートすることができる。従って、本アーキテクチャは、インプリメンテーション不変式で包括性がある。異なる要件が必要な場合は、異なる L U T を使用することで対処できる。図 1 8 と図 1 9 は、起点が128と96のそれぞれの例を示しており、L U T 表を再ローディングすることにより、C I E L\* a\* b\* と F a x L\* a\* b\*用のb\*チャンネルを収容する方法を示している。

40

50

## 【 0 0 8 4 】

提案の方法を、スキャナRGB入力をCIE L\*a\*b\*色出力に変換する、スキャナ色変換アプリケーションでテストした。キャリブレーション目標として標準ITU-8チャートを使用している。変換精度を、固定線形ノード・スペーシング、2の乗数ノード・スペーシング、及びノード数が2倍の非線形ノード・スペーシングを使った基準インプリメンテーション、により得られたCIE Lab値の間の色差信号 E によって測定した。始めの2つの方法、即ち、固定線形技法と、提案の2の乗数技法では、ノード数を同数として、17×17×17 LUTと四面体補間を使用した。ベンチマークとして機能する遙かに高密度の補間として、非線形立方根ノード・スペーシングで32×32×32 LUTを使用した。最初の2つの方法のノード位置を下に示す。

10

## 【 0 0 8 5 】

線形 :	0	1 6	3 2	4 8	6 4	8 0	9 6	1 1 2	1 2 8	1 4 4	1 6 0
	1 7 6	1 9 2	2 0 8	2 2 4	2 4 0	2 5 5					
2の乗数 :	0	4	8	1 6	3 2	4 8	6 4	8 0	9 6	1 1 2	1 2 8
	4 4	1 6 0	1 7 6	1 9 2	2 2 4	2 5 5					

## 【 0 0 8 6 】

線形ノード・スペーシングで得られた平均 E と、2の乗数ノード・スペーシングで得られた平均 E は、それぞれ2.67と1.53であった。この実験から分かるように、新しい2の乗数アーキテクチャは、固定線形ノード・スペーシング・アーキテクチャより遙かに正確な変換を提供しており、余分な計算量も少なく済んでいる。

20

結論として、提案している2の乗数ノード・スペーシング・アーキテクチャは、柔軟性と複雑性の間に良好なトレードオフを提供する。固定線形ノード・スペーシング・アーキテクチャとは異なり、画像品質要件により良く合致する非線形ノード・スペーシングを選択する柔軟性を保持していると同時に、他方では、完全プログラム可能ノード・アーキテクチャとは異なり、実施が簡単で費用効率が高い。このアプリケーションの一部として使用されるSCC(スキャナ色空間変換)のアプリケーションでは、提案している方法は、従来の方法に比べて、カラー忠実度を略2倍向上させることが実証されている。

## 【 0 0 8 7 】

図1に戻るが、統計モジュール(STS)210は、ページ背景除去や、ページが白黒か有色かを判定する自動ニュートラル検知などの、各種画像強調オペレーションを適用する目的で、画像コンテンツについて基本的な統計を収集することに責任を持つ。この統計モジュール210は、スキャナノイズの一部を除去するために、ディ・スクリーン・モジュール206からのぼやけた信号BLR208を使用する。出力は、特別な3-Dカラー・ヒストグラム・アレイである。

30

## 【 0 0 8 8 】

統計モジュール(STS)210は、画像クリーンアップ及び更に強調という、より大きなチェーンの一部である。チェーン全体を図18に示す。統計モジュール(STS)210は、画素値と色についての情報を収集することにより、チェーンの第1段階を形成する。画像強調チェーンの第2の段階では、統計モジュール(STS)210により収集されたデータが、キー画像強調パラメータを求めるために解析される。最終的に、これらパラメータの値は、画像強調装置に送られ、実際の画像強調が行なわれる。

40

## 【 0 0 8 9 】

図20の3つの機能の区分けは、各部分の異なる役割を説明する上で役に立つ。統計収集フェーズ802は、通常、画素および色値が入手可能な画像捕捉時(フロントエンド)に行なわれる。入力画像は、セグメントに分けられエンコード・パイプラインで圧縮された後は、各種面を圧縮解凍してマージする[という時間を消費するオペレーション]なしには、直接入手できなくなる。画像解析806フェーズは統計データに頼って、強調パラメータのセット808を抽出する。

## 【 0 0 9 0 】

50

画像強調チェーンのオペレーションの例は、背景除去である。単純な白黒テキスト文書は、黒字を、そうではない白色背景上に置いたものとして再生するよう期待されている。しかしながら、このようなページの走査画像が、完全にクリーンな（即ち、輝度値100の）背景を有することは殆どありえない。一般には、背景領域の画素値は、ノイズで、実際の輝度値は用紙媒体の反射率に依存している。しかし、低品質の新聞紙から高反射率の写真紙まで、広範囲に及ぶ媒体種に対してクリーンな背景を生成することが望まれている。

#### 【0091】

背景除去に対する標準的なアプローチは、ページの「白点」値を求め、この点を使って、圧縮された画像のヘッダファイル内のタグを変更することである。実際の背景除去は、圧縮解凍とマージの後のバックエンドで行われ、ここで色調再生（TRC）ルックアップ・テーブルを通して画素値が修正される。

10

#### 【0092】

なお、上記背景除去例では、統計データは、白点として識別された1つのパラメータを抽出するために使用される。より高精度の背景除去アルゴリズムは、黒点、スキャナノイズ量、及び背景色に関する付加的情報も抽出する。ここで重要なことは、統計データは、再構築時に強調の程度を制御する数個の強調パラメータを抽出するために、解析フェーズで使用されるということである。

#### 【0093】

（AIEの様な）自動的リアルタイム画像強調又は領域セグメント化などのような追加的画像強調オペレーションを適用して、CEF画像を更に強調することができる。これらには、テキスト又は画像データの自動検知と強調が含まれる。統計モジュール（STS）210により収集された情報は、このようなオペレーションに使用することができる。

20

#### 【0094】

統計モジュール（STS）210は、ノイズを低減するために、画像のぼやけたバージョン208についての統計を蓄積する。統計モジュール（STS）210は、輝度/クロミナンス空間（Lab、YCCなど）内で、入力としてフルカラー信号を取る。入力信号は、普通は、スキャン色変換（SCC）モジュール内の高速走査方向に2倍サブサンプルされるものと想定されている。しかしながら、統計モジュール（STS）210アルゴリズムは、汎用性があり、サブサンプルされたデータもサブサンプルされていないデータも等しく良好に処理することができる。

30

#### 【0095】

走査、印刷、コピーなど一般的なアプリケーションでは、完全に結合された3次元ヒストグラムは不要であることが分かっている。それに代えて、従属クロミナンス合計値を伴う輝度チャンネルの1-Dヒストグラムは適切な結果をもたらすと判断された。従って、統計モジュール（STS）210の出力は、以下のデータ・アレイを含んでいる。

- 1) 輝度ヒストグラム（各32ビット、256個の符号のないエントリのアレイ）
- 2) 2つの従属クロマ合計（各32ビット、256個の符号付のエントリのアレイ）
- 3) 4つの32ビット輝度および色強度レジスタ（以下に説明）

40

#### 【0096】

ヒストグラム・ピンの数を減らすためには、メモリの制限されたインプリメンテーションが求められる。128ピンのインプリメンテーションでは、画像の黒点と白点の両方が判定できるように、中央の128ピンを除去して、64ピンをゼロ付近に残し、64ピンを255付近に残す。ピンの数は、アプリケーションと精度の必要性に合わせてプログラム可能である。

#### 【0097】

画像が殆ど不変である場合は、スキャナ分解能と最大媒体サイズにより最大合計サイズが決まる。

50

Max-Chroma-Sum=255\*width\*height\*resolution<sup>2</sup>

【0098】

32ビット幅カウンタ/レジスタを超えないことが望ましい。600dpiで11×17インチの画像サイズの場合、最大クロマ合計は、34ビット幅になるので、最大合計<232を維持するためある程度のサブサンプリングが許容される。最大クロマ合計が32ビットに合うように、xとy方向に、十分に均等な2の乗数のサブサンプリングを行なうことになる。

【0099】

統計モジュール(STS)210のオペレーションは以下の通りである。8ビットの符号なしの着信輝度入力信号Lを、データ・アレイ内の特定の位置を指すアドレスとして使用する。同じアドレスを全3アレイに対して使用する。

10

【0100】

輝度Lのヒストグラム要素については、現下の輝度値でのアレイのコンテンツが1だけ増やされる。このヒストグラムは、同じ輝度値を有する画素のカウントにより与えられた輝度値の分布により形成される。

【0101】

STSユニット210は、2つのクロマ・チャンネル(A、B)を備えており、実際のクロマ値が、所与の輝度アドレスの従前のコンテンツに加えらる。色手段は、後に、輝度カウントにより加えられた値を正規化することにより計算される。しかしながら、クロマ要素は符号付きの数なので、クロマ・アレイは、8ビットの符号付き範囲を収容するため、より大きなワードサイズを使用せねばならない。

20

【0102】

これは、ページのニュートラリティを判定するために使用される2つの色強度測定と、ページの二進検知のための2つの輝度強度測定とを含んでいる。各着信画素毎に、先ず、ニュートラル軸(A=B=0)からのマンハッタン距離を以下のようにして計算する。

色距離 = abs(A-128)+abs(B-128)

【0103】

ここで、128のバイアスは、クロマ要素の符号バイアスによるものである。次に、色距離は、2つの16ビット閾値、即ち弱い色の閾値WeakClrClrThr及び強い色の閾値StrongClrClrThr、と比較される。色距離が弱い色の閾値以上である画素の数と、色距離が強い色の閾値以上である画素の数を、別々に数える。同様のアプローチを二進ページ検知について行なう。11×17ページの場合、この2つのカウントは、2つの32ビットレジスタとして提供される。

30

【0104】

アレイ・サイズを縮小するために、着信クロマ・チャンネルを、或いは輝度チャンネルさえも、更にサブサブサンプルするというオプションも在る。少量のサブサンプリングは多分許容可能だが、アレイ・サイズを大幅に縮小することにはならない。サブサンプリング量を増やすには実行可能性の検討が必要で、小さな強い有色形状が拡大サブサンプリングのせいで失われるかもしれないページニュートラリティについては、特にそうである。

40

【0105】

圧縮モジュール(CMP)252は、独立的に4つのCEF平面上で作動する。このモジュールは、損失の大きいJPEG圧縮法を使ってカラーの前景256と背景254面を圧縮し、二進セクタ258とヒント262平面では損失のないITUグループ4圧縮を使用する。

最後に、圧縮モジュール(CMP)252からの圧縮されたMRC平面と、統計モジュール(STS)212からの収集されたヒストグラムデータは、DXWラップ・モジュール214内で一緒にラップされて、TIFF又はPDF MRC CEF出力270を形成する。

【図面の簡単な説明】

【0106】

【図1】混合ラスタ・コンテンツ画像(Mixed Raster Content images)を用いるためのエ

50

ンコード・パイプラインのブロック図を示す。

【図2】走査色変換モジュール(Scan Color Conversion Module) S C Cの詳細なブロック図を示す。

【図3】高速走査方向(fast scan direction)での2 x サブサンプリングの適用を示す。

【図4】サブサンプルされたデータからの、オリジナルの(サブサンプルされていない)画素の再構築の方法を示す。

【図5】スケール・モジュール(Scale Module)の詳細なブロック図を示す。

【図6】固定整数および分数表記(fixed integer and fraction notation)を用いて表現された高速および低速走査方向でのスケージングの総量を示す。

【図7】スケール・モジュールで用いられる16ビット変数を示す表。

【図8】スケール・モジュールの動作を示す。

【図9】スケール・モジュールでの3つの1D補間ステップを示す。

【図10】全域強調モジュール(Gamut Enhancement Module) G M Eのブロック図である。

【図11】カラー・ルックアップ・テーブルL U Tのブロック図である。

【図12】トリ-リニア補間(tri-linear interpolation)の3つのステップを示す。

【図13】1 - Dでの補間ステップを示す。

【図14】カラー・ルックアップ・テーブルL U Tを計算するためのフローチャートである。

【図15】図14に示されるフローチャートとともに使用されるブロック図である。

【図16】2ノード・スペーシング技術で使用される値のための表を示す。

【図17】8ビット入力の替わりの10ビットの値のための表を示す。

【図18】L U Tに対する128での例示の値を持つ表を示す。

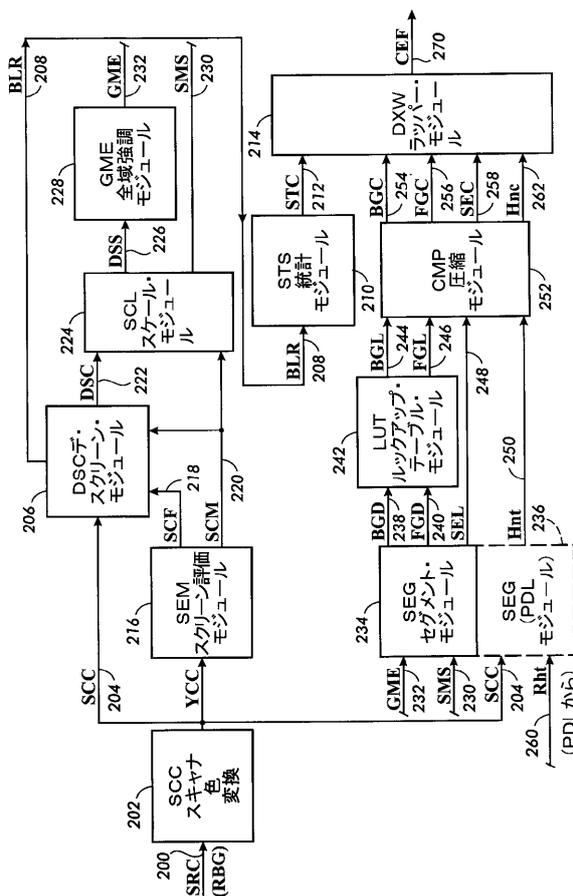
【図19】L U Tに対する128での例示の値を持つ表を示す。

【図20】統計モジュールのブロック図である。

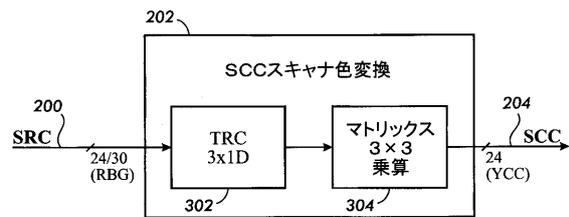
10

20

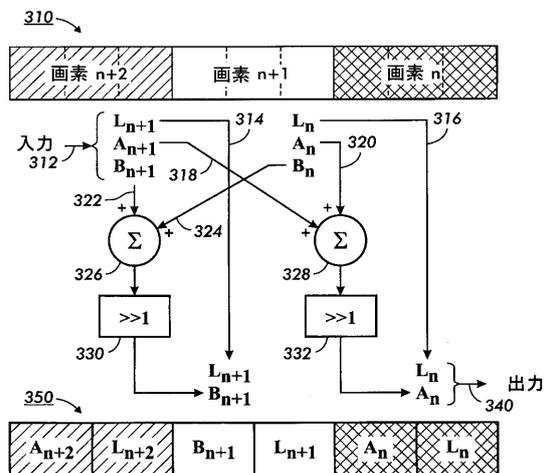
【図1】



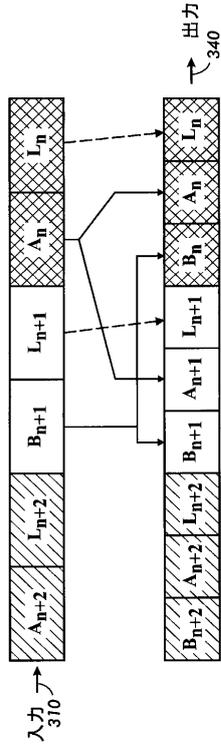
【図2】



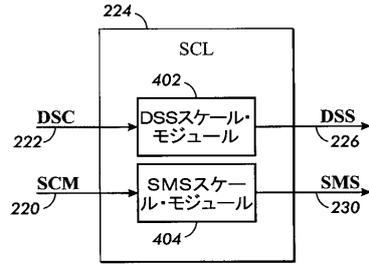
【図3】



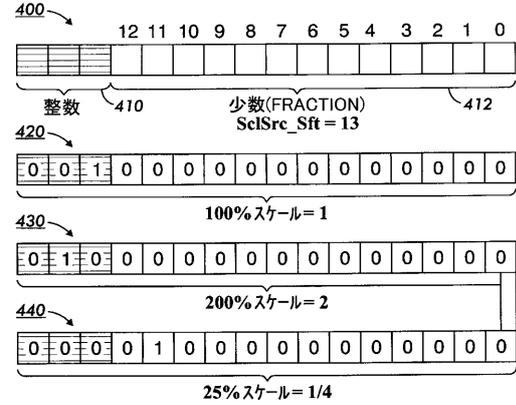
【図4】



【図5】



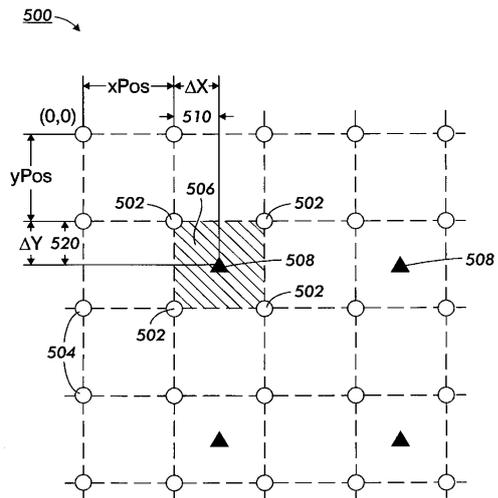
【図6】



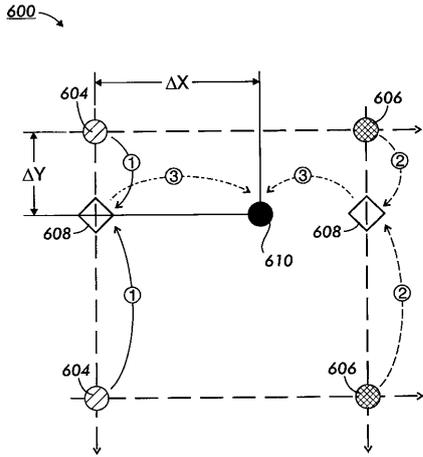
【図7】

変数	初期化	意味	使用
ScISrc_Sft	13	固定 (3.13)	一定
StpSrc_X	$(1 < < \text{ScISrc\_Sft}) / \text{scale\_X}$	$\Delta X$	$X\_Src\_step$
StpSrc_Y	$(1 < < \text{ScISrc\_Sft}) / \text{scale\_Y}$	$\Delta Y$	$Y\_Src\_step$
マスク	$(1 < < \text{ScISrc\_Sft}) - 1$	$1.0 - \epsilon$ (1 LSB)	分数(fraction)を得るために加算
半分	$(1 < < \text{ScISrc\_Sft}) >> 1$	0.5	丸めのために加算され得る

【図8】



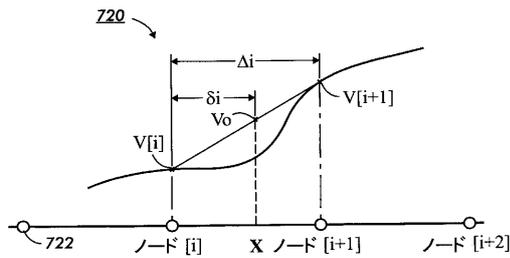
【図9】



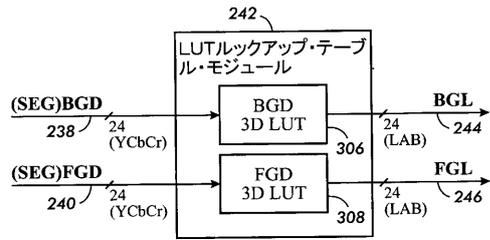
【図10】



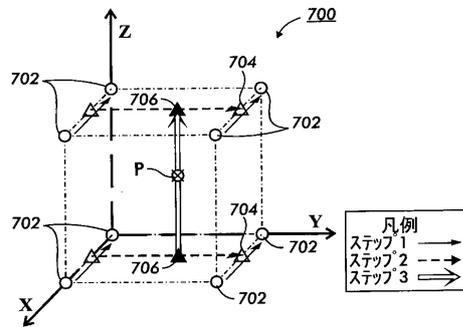
【図13】



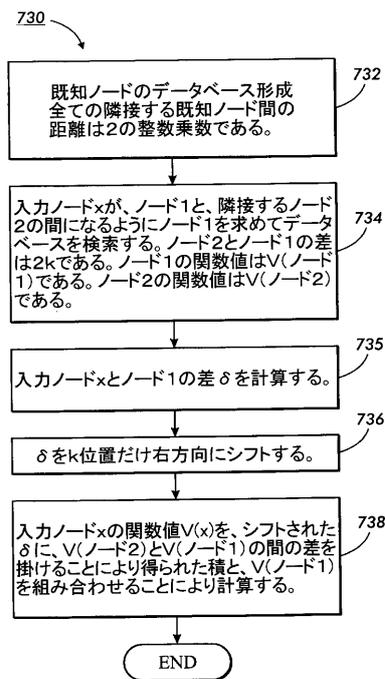
【図11】



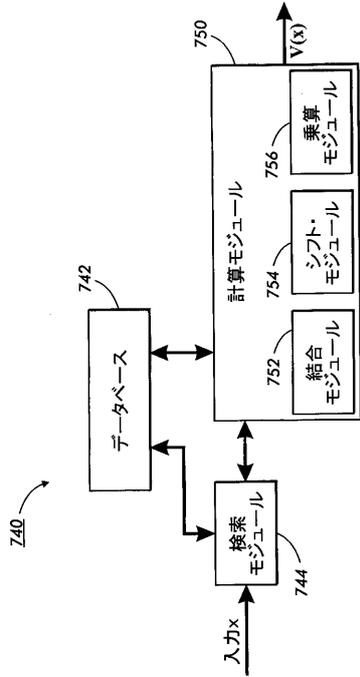
【図12】



【図14】



【図15】



【図16】

ノード・インデックス	ノード値	指数
0	0*	2
1	4	2
2	8	3
3	16	4
4	32	4
5	48	4
6	64	4
7	80	4
8	96	4
9	112	4
10	128	4
11	144	4
12	160	4
13	176	4
14	192	5
15	224	5
16	255	5

【図17】

ノード・インデックス	ノード値	指数
0	0*	4
1	16	4
2	32	5
3	64	6
4	128	6
5	192	6
6	256	6
7	320	6
8	384	6
9	448	6
10	512	6
11	576	6
12	640	6
13	704	6
14	768	7
15	896	7
16	1023	7

【図18】

ノード・インデックス	ノード値	指数
0	0	5
1	32	5
2	64	4
3	80	4
4	96	4
5	112	3
6	120	2
7	124	2
8	128*	2
9	132	2
10	136	3
11	144	4
12	160	4
13	176	4
14	192	5
15	224	5
16	255	5

(a\*およびb\*に対して起点は128)

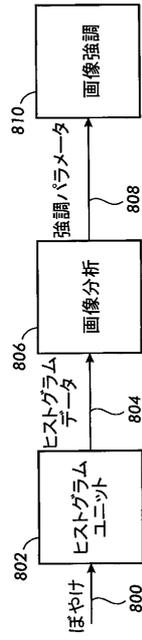
【図 19】

790

ノード・インデックス	ノード値	指数
0	0	5
1	32	4
2	48	4
3	64	4
4	80	3
5	88	2
6	92	2
7	96*	2
8	100	2
9	104	3
10	112	4
11	128	4
12	144	5
13	176	5
14	192	5
15	224	5
16	255	5

(Fax b'に対して起点は128)

【図 20】



## フロントページの続き

(51)Int.Cl. F I  
**H 0 4 N 1/40 (2006.01)** H 0 4 N 1/40 D  
 H 0 4 N 1/40 1 0 1 Z

(31)優先権主張番号 10/612,261

(32)優先日 平成15年7月1日(2003.7.1)

(33)優先権主張国 米国(US)

(72)発明者 ドナルド ジェイ カリー

アメリカ合衆国 カリフォルニア州 9 4 0 2 5 メンロ パーク リーランド アヴェニュー  
 3 3 3

(72)発明者 ドロン クレッター

アメリカ合衆国 カリフォルニア州 9 4 4 0 2 サン マテオ タートル ベイ プレイス 2  
 0

(72)発明者 アスガー ナファリー

アメリカ合衆国 カリフォルニア州 9 4 0 2 5 メンロ パーク モリー ドライヴ 5 2 5

審査官 松永 隆志

(56)参考文献 特開平09-093445(JP,A)

特開2002-077616(JP,A)

特開平07-184072(JP,A)

(58)調査した分野(Int.Cl., DB名)

G 0 6 T 1 / 0 0

G 0 6 T 5 / 2 0

G 0 6 T 5 / 4 0

H 0 4 N 1 / 4 0

H 0 4 N 1 / 4 6

H 0 4 N 1 / 6 0