

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2014-200014

(P2014-200014A)

(43) 公開日 平成26年10月23日(2014.10.23)

(51) Int.Cl.			F I			テーマコード (参考)	
HO4N	1/46	(2006.01)	HO4N	1/46	Z	5B057	
HO4N	1/60	(2006.01)	HO4N	1/40	D	5C077	
G06T	1/00	(2006.01)	G06T	1/00	510	5C079	

審査請求 未請求 請求項の数 10 O L (全 23 頁)

(21) 出願番号 特願2013-74858 (P2013-74858)
 (22) 出願日 平成25年3月29日 (2013.3.29)

(71) 出願人 000001007
 キヤノン株式会社
 東京都大田区下丸子3丁目30番2号
 (74) 代理人 100076428
 弁理士 大塚 康德
 (74) 代理人 100112508
 弁理士 高柳 司郎
 (74) 代理人 100115071
 弁理士 大塚 康弘
 (74) 代理人 100116894
 弁理士 木村 秀二
 (74) 代理人 100130409
 弁理士 下山 治
 (74) 代理人 100134175
 弁理士 永川 行光

最終頁に続く

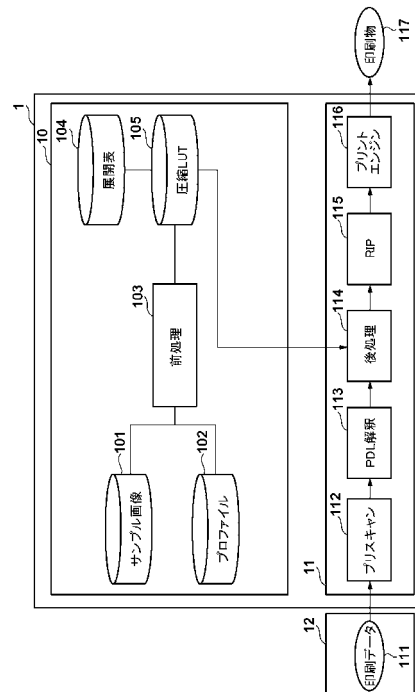
(54) 【発明の名称】 画像処理装置および画像処理方法

(57) 【要約】

【課題】色変換用LUTの必要部分のみをメモリ展開して参照することで、メモリ使用量を削減しつつ、高速な色変換を行うことを可能とする。

【解決手段】前処理部103で、色変換LUTを部分ごとに圧縮した圧縮LUTを作成する。すなわち、サンプル画像記憶部101に保持されたサンプル画像の色分布に応じて、色変換LUTの入力値を示す各格子点を、高頻度で参照される高参照部とその他の低参照部に区分するための参照表(展開表)を作成し、これを展開表記憶部104に保持する。また、プロファイル記憶部102に保持された色変換条件に基づいて各格子点に対する出力値を算出する。そして各格子点の出力値を、参照表が示す高参照部ごと、および低参照部ごとにデータ圧縮した圧縮LUTを作成し、圧縮LUT記憶部105に保持する。後処理部114では圧縮LUTおよび参照表を用いて、色変換LUTの高参照部のみを伸張した部分LUTを作成し、色変換を行う。

【選択図】 図1



【特許請求の範囲】**【請求項1】**

第1のデバイスに依存する第1の色空間から、第2のデバイスに依存する第2の色空間への変換用の色変換テーブルを作成する画像処理装置であって、

サンプル画像の色分布に応じて、前記色変換テーブルの入力値を示す各格子点が、色変換時に高頻度で参照される高参照部とその他の低参照部のいずれに含まれるかを示すための参照表を作成する参照表作成手段と、

前記第1および第2のデバイスに応じた色変換条件を入力する変換条件入力手段と、

前記色変換条件に基づいて前記色変換テーブルの各格子点に対する出力値を算出する算出手段と、

前記色変換テーブルにおける各格子点の出力値を、前記参照表が示す高参照部ごとにデータ圧縮し、該参照表が示す低参照部ごとにデータ圧縮した圧縮テーブルを作成する圧縮手段と、

前記参照表と前記圧縮テーブルを対応付けて保持する保持手段と、

を有することを特徴とする画像処理装置。

10

【請求項2】

前記参照表作成手段は、前記サンプル画像から、前記色変換テーブルの各格子点に対応する値を有する画素の出現確率を算出し、該出現確率が所定値以上である格子点を高参照部に区分し、該出現確率が該所定値未満である格子点を低参照部に区分することで、前記参照表を作成することを特徴とする請求項1に記載の画像処理装置。

20

【請求項3】

さらに、色変換対象となる画像データを入力する画像入力手段と、

前記保持手段に保持された前記参照表および前記圧縮テーブルを取得するテーブル取得手段と、

前記圧縮テーブルから、前記参照表によって高参照部に区分された格子点の圧縮データを伸張した部分テーブルを作成する部分伸張手段と、

前記部分テーブルを用いて、前記画像データの画素ごとに前記第1の色空間から前記第2の色空間への変換を行う色変換手段と、

を有することを特徴とする請求項1または2に記載の画像処理装置。

30

【請求項4】

前記色変換手段は、

前記画像データの画素ごとに、前記圧縮テーブルにおいて当該画素の値が対応する注目格子点が、前記参照表において高参照部に区分されているか否かを判定する判定手段と

、

前記判定手段で前記注目格子点が前記参照表において高参照部に区分されていないと判定された場合、前記圧縮テーブルから、前記参照表において前記注目格子点が含まれる低参照部の圧縮データを伸張して部分テーブルを作成する注目伸張手段と、

前記参照表において前記注目格子点が含まれる低参照部を高参照部の区分に更新する更新手段と、を有し、

前記部分伸張手段および前記注目伸張手段で作成された部分テーブルを用いて、前記色変換を行うことを特徴とする請求項3に記載の画像処理装置。

40

【請求項5】

さらに、前記部分伸張手段で伸張された前記部分テーブルを保持するメモリの空き容量を取得する空き容量取得手段と、

前記空き容量が、前記圧縮テーブルから全格子点の圧縮データを伸張した色変換テーブルを保持するのに必要となる必要容量以上であるか否かを判定する容量判定手段と、

前記容量判定手段で前記空き容量が前記必要容量以上であると判定された場合に、前記圧縮テーブルから全格子点の圧縮データを伸張して前記色変換テーブルを作成する全伸張手段と、を有し、

前記容量判定手段で前記空き容量が前記必要容量未満であると判定された場合に、前記

50

部分伸張手段によって前記圧縮テーブルから前記部分テーブルを作成する、
ことを特徴とする請求項3または4に記載の画像処理装置。

【請求項6】

さらに、撮影モードまたは色変換条件の異なる画像種別ごとに前記サンプル画像を入力するサンプル入力手段を有し、

前記参照表作成手段および前記圧縮手段は、前記参照表および前記圧縮テーブルを画像種別ごとに作成し、

前記テーブル取得手段は、前記画像入力手段で入力された前記画像データの画像種別に
応じた前記参照表および前記圧縮テーブルを、前記保持手段から取得する
ことを特徴とする請求項3乃至5のいずれか1項に記載の画像処理装置。

10

【請求項7】

さらに、画像入力手段で入力された複数の画像データのそれぞれについて、対応する前
記参照表および前記圧縮テーブルが前記保持手段に保持されているか否かを判定する保持
判定手段と、

前記保持判定手段で前記参照表および前記圧縮データが保持されていないと判定された
画像データについて、前記色変換後の処理が開始されるまでのスプール期間内に、対応す
る該参照表および該圧縮テーブルを作成する作成手段と、を有し、

前記作成手段は、当該画像データに応じた前記参照表および前記圧縮テーブルを作成す
るのに要する予想作成時間が前記スプール期間より長い場合に、前記画像データの処理順
を変更して前記スプール期間を延長することを特徴とする請求項3乃至6のいずれか1項に
記載の画像処理装置。

20

【請求項8】

前記予想作成時間は、予め作成された、色変換条件と前記予想作成時間の対応表から、
前記画像データの色変換条件に基づいて取得されることを特徴とする請求項7に記載の画
像処理装置。

【請求項9】

参照表作成手段、変換条件入力手段、算出手段、圧縮手段、および保持手段を有し、第
1のデバイスに依存する第1の色空間から、第2のデバイスに依存する第2の色空間への色変
換用の色変換テーブルを作成する画像処理装置における画像処理方法であって、

前記参照表作成手段が、サンプル画像の色分布に応じて、前記色変換テーブルの入力値
を示す各格子点が、色変換時に高頻度で参照される高参照部とその他の低参照部のいず
れに含まれるかを示すための参照表を作成し、

30

前記変換条件入力手段が、前記第1および第2のデバイスに応じた色変換条件を入力し、
前記算出手段が、前記色変換条件に基づいて前記色変換テーブルの各格子点に対する出
力値を算出し、

前記圧縮手段が、前記色変換テーブルにおける各格子点の出力値を、前記参照表が示す
高参照部ごとにデータ圧縮し、該参照表が示す低参照部ごとにデータ圧縮した圧縮テー
ブルを作成し、

前記保持手段が、前記参照表と前記圧縮テーブルを対応付けて保持することを特徴とす
る画像処理方法。

40

【請求項10】

コンピュータ装置で実行されることにより、該コンピュータ装置を請求項1乃至8のいず
れか1項に記載の画像処理装置の各手段として機能させるためのプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、画像データに対する色変換処理を行う画像処理装置および画像処理方法に関
する。

【背景技術】

【0002】

50

従来、書籍やパンフレットなどの商業印刷物を印刷する商業印刷分野では、オフセット印刷やグラビア印刷などのアナログ印刷機が用いられてきた。しかし、多品種少部数印刷という市場ニーズの変化から、インクジェット方式や電子写真方式等のデジタル印刷機の導入が進行している。デジタル印刷においてもアナログ印刷と同様に、印刷処理の高速化が要求されることはもちろんである。

【0003】

一般にデジタル印刷処理は以下のような流れで行われる。まず、パーソナルコンピュータなどの計算機を用いて作成されたデジタル印刷データが印刷サーバに投入されると、印刷サーバはその印刷データが後続の印刷処理に必要な情報を完備しているかを検査する(プリスキャン)。そして、必要な情報が揃っていれば、ページ記述言語(Page Description Language : PDL)で記述された印刷データを解析し、印刷データ中の画像データを印刷機で再現可能な色に変換する(色変換処理)。そして、色変換後の画像データを含む印刷データをシアン、マゼンタ、イエロー、ブラック(CMYK)などの版ごとの2値データに分解して、印刷機に転送する。該転送された2値データを印刷機が印刷用紙に転写することで、印刷物が完成する。

【0004】

上記印刷処理における画像の色変換処理では、sRGB色空間で記述された画像の画素値を印刷機の色信号値に対応付ける、所謂マッピング処理が行われる。マッピング処理においては、sRGB色空間の各軸を8分割や16分割など所定数に分割することで、空間を格子状に分割する。そして、各格子点の入力値に対応する出力値を算出し、この対応表をルックアップテーブル(Look-Up Table:LUT)として作成しておく。入力画像の画素値、すなわち入力値が入力された際には、このLUTを参照して該入力値に対応する出力値を算出する。入力値が格子点と一致する場合には、その格子点における出力値が、該入力値に対応する出力値となるが、入力値が格子点と一致しない場合には、入力値を包含する格子点から補間演算等により出力値を算出する必要がある。この補間演算としては、立方体補間や四面体補間など、使用する格子点に応じて様々な補間方法が提案されている。

【0005】

ここで補間演算処理について、補間精度の高い四面体補間を例として説明する。ある入力値に対し、まず入力値が属する格子を探索し、格子内の6つの四面体のうち、入力値がどの四面体の中に存在するかの内外判定を行って、入力値を包含する四面体を特定する。次に、特定された四面体の1頂点を原点とし、原点から入力値の点までの距離と、原点から他の3頂点までのそれぞれの長さとの内分比から、入力値に対応する出力値を算出する。

【0006】

しかしながら補間演算を行う場合、入力値が属する格子の探索、格子内の四面体に対する入力値の内外判定、および内分比の計算等、複雑な演算を行う必要があるため、演算速度が遅くなってしまいう問題がある。また補間演算は、格子点から間の点を補間するため、実際の格子点間の色の変化が非線形であった場合には、補間精度が著しく低下することになる。そこで、色変換LUTを格子状に間引かず、全ての入力値に対する出力値(以下、全体LUT)を算出しておき、この全体LUTを用いることによって複雑な補間演算を行わずに出力値を得ることができる。しかしながら、例えば印刷データの用紙サイズがA4で400dpiである場合、少なくとも44MBの作業用メモリが必要である。これに対し、全入力値に対する全出力値を持つ全体LUTは、例えば各チャンネルが8ビットの色深度で3チャンネルのLUTであれば、そのデータ容量だけで48MBに達する。このため、並列計算器で全体LUTを用いて印刷データを処理すると、作業用のメモリ容量が全体LUTのみで圧迫されるため、演算装置を複数搭載した計算機であっても処理の並列度を上げられず、すなわち印刷処理速度を上げることができない。

【0007】

そこで、LUT自身によるメモリ使用量を削減するために、以下のような技術が提案されている。例えば、YCbCr色空間に対するLUTにおいて、入力画像の最大輝度に応じて、保持

10

20

30

40

50

するLUT成分を制限する技術が提案されている(例えば、特許文献1参照)。この技術によれば、例えば入力画像の最大輝度 Y_{max} が50であった場合に、 Y_{max} を超えるLUT成分を保持しないことで、LUTのデータ容量を削減する。また、LUTを粗く分割した分割領域ごとに1次直線で近似し、元のLUT成分の値と近似値との誤差値を算出して、分割領域ごとに1次直線の係数と誤差値を保持しておく技術が提案されている(例えば、特許文献2参照)。この技術によれば、入力値に対応する出力値は、対応する1次直線の係数と誤差値から算出されるため、LUTのデータ容量が削減される。

【先行技術文献】

【特許文献】

【0008】

【特許文献1】特開2008-245029号公報

【特許文献2】特許第4411230号公報

【発明の概要】

【発明が解決しようとする課題】

【0009】

しかしながら、上記特許文献1に開示された技術においては、99%の画素の色分布が偏っていたとしても、残り1%の画素がその偏りから外れていた場合には、作成されるLUTの容量は全く圧縮されない。したがってこのような場合、LUTによるメモリ使用量を削減することはできない。また、上記特許文献2に開示された技術においては、メモリ使用量の圧縮率が低く(例えば93.5%)、削減効果は僅かであった。

【0010】

本発明は上記の問題を解決するためになされたものであり、メモリ使用量を削減しつつ、高速な色変換を行うことを可能とする画像処理装置および画像処理方法を提供することを目的とする。

【課題を解決するための手段】

【0011】

上記目的を達成するための一手段として、本発明の画像処理装置は以下の構成を備える。すなわち、第1のデバイスに依存する第1の色空間から、第2のデバイスに依存する第2の色空間への変換用の色変換テーブルを作成する画像処理装置であって、サンプル画像の色分布に応じて、前記色変換テーブルの入力値を示す各格子点が、色変換時に高頻度で参照される高参照部とその他の低参照部のいずれに含まれるかを示すための参照表を作成する参照表作成手段と、前記第1および第2のデバイスに応じた色変換条件を入力する変換条件入力手段と、前記色変換条件に基づいて前記色変換テーブルの各格子点に対する出力値を算出する算出手段と、前記色変換テーブルにおける各格子点の出力値を、前記参照表が示す高参照部ごとにデータ圧縮し、該参照表が示す低参照部ごとにデータ圧縮した圧縮テーブルを作成する圧縮手段と、前記参照表と前記圧縮テーブルを対応付けて保持する保持手段と、を有することを特徴とする。

【発明の効果】

【0012】

本発明によれば、メモリ使用量を削減しつつ、高速な色変換を行うことが可能となる。

【図面の簡単な説明】

【0013】

【図1】第1実施形態における印刷システムの構成例を示すブロック図、

【図2】画像種別ごとの色分布例を示す図、

【図3】格子ごとの出現確率の例を示す図、

【図4】第1実施形態における論理構成例を示すブロック図、

【図5】第1実施形態における前処理および後処理を示すフローチャート、

【図6】第1実施形態における展開表の一例を示す図、

【図7】第1実施形態における展開表生成処理および圧縮LUT生成処理を示すフローチャート、

10

20

30

40

50

【図8】第1実施形態における展開判定処理および展開処理を示すフローチャート、
 【図9】第2実施形態における論理構成例を示すブロック図、
 【図10】第2実施形態における後処理を示すフローチャート、
 【図11】第2実施形態におけるUI例を示す図、
 【図13】第3実施形態における論理構成例および後処理を示すフローチャート、
 【図12】第3実施形態における圧縮LUT一覧表および予想処理時間表の一例を示す図、
 【図14】本発明が適用可能なクラウドシステムの構成例を示すブロック図、である。
 【発明を実施するための形態】

【0014】

以下、本発明の実施形態について、図面を参照して説明する。なお、以下の実施の形態は特許請求の範囲に関する本発明を限定するものではなく、また、本実施の形態で説明されている特徴の組み合わせの全てが本発明の解決手段に必須のものとは限らない。

【0015】

< 第1実施形態 >

システム構成

まず、本実施形態における印刷システムの構成例について、図1を用いて説明する。図1に示す印刷システム1は、前処理を行う前処理サーバ10と、後処理を行う後処理サーバ11からなる。12は、文字や画像を含む印刷情報を記録した印刷データ111を印刷システム1に転送する転送サーバである。

【0016】

前処理サーバ10において、101は複数のサンプル画像を保持するサンプル画像記憶部である。ここでサンプル画像とは、画像特徴や撮影シーン等の画像種別毎に分類された、複数の画像である。102は入出力装置の色再現特性および、色変換マッピングを行うためのマッピング情報を記録したプロファイルを保持するプロファイル記憶部である。ここで入出力装置とは、印刷対象となる印刷データ111を作成した装置(不図示のディスプレイや撮像装置等)と、該印刷データ111を印刷出力する装置(プリントエンジン116等)を示す。

【0017】

また103は前処理部であり、全ての入力色と出力色との関係を記述した全体LUTにおける各格子について、展開すべきか否かを示す展開情報を記録した展開表(参照表)を作成する。前処理部103ではまた、該展開表に応じて全体LUTの各格子の圧縮情報を記録した圧縮テーブル(圧縮LUT)を生成する。なお展開表および圧縮LUTは、上記サンプル画像の画像種別ごとに生成される。104は展開表を保持する展開表記憶部、105は圧縮LUTを保持する圧縮LUT記憶部である。なお、本実施形態における展開表および圧縮LUT詳細については後述する。

【0018】

後処理サーバ11において、112は印刷データ111について印刷に必要な情報の欠落を検査するプリスキャン部、113は印刷データ111をPDLデータに変換するPDL解釈部である。114は、PDLデータに含まれる画像に対し、適切な展開表および圧縮LUTに基づく色変換を施す後処理部である。このとき、圧縮LUTを全て展開するのではなく、必要な部分のみを展開(部分展開)することを特徴とする。115は、PDLデータおよび色変換後の画像をラスターイメージ(ビットマップ画像)データに変換するRIP(ラスターイメージプロセッサ)部である。116はRIP処理されたRIPデータを印刷メディアに印刷するプリントエンジンである。そして117が、プリントエンジン116によって印刷出力された印刷物である。

【0019】

なお、前処理サーバ10、後処理サーバ11、転送サーバ12は、以下のようなハードウェア構成からなる例えばPCによって実現される。すなわち、CPU、RAM、ROM、VRAM、キーボードコントローラ、磁気記憶媒体、光学記憶媒体、ネットワークコントローラ、入出力コントローラ、などがバスに接続された構成であるとする。しかしながら、PCの詳細構成については本発明の主たる要件ではないため、詳細な説明については省略する。

【0020】

10

20

30

40

50

上記、印刷システム1と転送サーバ12はネットワークを介して接続されており、本実施例ではインターネットを介して接続されているものとする。なお、印刷システム1と転送サーバ12は、互いにデータ交換が可能であれば、如何なる技術、手段を用いて接続されていても良い。

【0021】

次に、図1を用いて、本実施形態の印刷システム1における印刷処理の概略を説明する。印刷システム1では、まず前処理サーバ10の前処理部103において、サンプル画像記憶部101に保持された画像種別毎のサンプル画像と、プロファイル記憶部102に保持されたプロファイルに基づき、画像種別毎の展開表および圧縮LUTを生成する。生成された展開表と圧縮LUTはそれぞれ、展開表記憶部104、圧縮LUT記憶部105に保持される。すなわち、実際の印刷処理の前処理として、展開表と圧縮LUTが生成される。そして、実際の印刷対象である印刷データ111が入力されると、後処理サーバ11において、上述したようにプリスキャン、PDLの解釈、色変換、RIPデータ生成、プリント処理が行われ、印刷物117が生成される。この色変換時に、適切な展開表および圧縮LUTが参照される。

【0022】

色分布特徴

ここで、画像の色分布の特徴について、図2および図3を用いて説明する。

【0023】

図2に、画像特徴やデジタルカメラの撮影モードごと(すなわち画像種別ごと)に分類されたサンプル画像についての、RGB色空間における色分布の例を示す。すなわち、(a)は、表計算ソフトウェアやドキュメント作成ソフトウェアで作成されるグラフィックス画像についての色分布を示す。また(b)~(i)は、デジタルカメラの撮影モードをポートレートやビーチなどの種々のシーンに適した設定で撮影することによって得られた自然画像についての色分布を示す。なお色分布グラフ(a)~(i)において、グラフ左下が黒、右上が白を表す。

【0024】

一般にグラフィックス画像であれば、グラフを作成したソフトウェアのデフォルト色が最も多く出現する。またポートレート画像であれば、人物が被写体であるため、肌色の色領域の色が多く出現するし、ビーチ画像であれば、青い海や青空が被写体であるため、青からシアンにかけての色領域の色が多く出現する。しかしながら色分布グラフ(a)~(i)に示すように、各画像種別において、出現頻度の高い色以外の色については、極端に出現数が少ないか、全く出現しない。すなわち、画像の色分布は、画像の種別ごとに極端な偏りがあると言える。

【0025】

ここで図3に、画像種別ごとのサンプル画像について、RGB色空間の各軸を33分割し、各格子を画素の出現確率の大きい順に整列させて出現確率を累積したグラフを示す。同図において横軸は格子番号、縦軸は累積出現確率[%]を示す。図3(a)はグラフィックス画像、図3(b)はポートレート画像、図3(c)は極彩色画像における、それぞれの全サンプル画像についての累積出現確率を示す。図3(a)によれば、グラフィックス画像における80%の画素は、全32,768格子中の5格子に集中している。同様に図3(b)のポートレート画像では436格子に、図3(c)の極彩色画像では2,688格子に集中している。これにより、色空間の全格子に対応する全てのLUT成分をメモリに展開しても、その90%以上は使用されないことが分かる。また、画像種別によって、使用するLUT部分および使用頻度が異なる。したがって、全体LUTの全成分のうち、実際にメモリに展開すべきLUT成分を画像の種別に応じて選択することで、効率的なメモリ使用が可能となる。

【0026】

そこで本発明では、上述したような色分布の特徴に基づき、全体LUTから、入力画像の画像種別に応じてメモリに展開する部分テーブル(部分LUT)を選択することで、少ないメモリ使用量での高速な色変換を可能とする。

【0027】

10

20

30

40

50

論理構成

以下、本実施形態の印刷システムの論理構成を図4に示し、説明する。

【0028】

図4(a)は、前処理部103において展開表、圧縮LUTを作成するための論理構成を示すブロック図である。同図において、301はサンプル画像に基づき画像の種別ごとの展開表を生成する展開表生成部、302はサンプル画像やプロファイル等の演算に必要なデータを入力する入力部である。また、303は色変換テーブル(色変換LUT)を生成する際に用いる入力値を生成する入力値生成部である。304はプロファイルに基づき、入力値を、第1のデバイスに依存する第1の色空間値から第2のデバイスに依存する第2の色空間値に変換する色変換部である。また、305は色変換LUTを所定の格子ごとに圧縮する圧縮部、である。

10

【0029】

図4(b)は、後処理部114において圧縮LUTの部分展開による色変換処理を行うための論理構成を示すブロック図である。同図において、401は色変換対象となる入力画像を入力する画像入力部、402は圧縮LUT記憶部105および展開表記憶部104から、入力画像に対応する圧縮LUTおよび展開表を入力する圧縮LUT入力部である。また、403は入力画像の画素値を包含する格子がメモリに展開されている否かを展開表に基づいて判定する展開判定部、404は圧縮LUTにおいて展開すべき格子の出力値を伸張してメモリに展開する展開部である。また、405はメモリ展開されたLUTを用いて、入力画像の画素値を第1の色空間値から第2の色空間値に変換する画像変換部、406は色変換後の画像を出力する画像出力部、である。

【0030】

なお、図4(a),(b)に示す論理構成を1台の画像処理装置において実現することも可能である。

20

【0031】

前処理(展開表,圧縮LUT作成処理)

以下、本実施形態における前処理について、図5(a)を用いて説明する。図5(a)は、前処理サーバ10の前処理部103にて行われる、展開表および圧縮LUTの作成処理を示すフローチャートである。まずS101で展開表生成部301が、サンプル画像記憶部101に保持されている複数のサンプル画像を入力し、該サンプル画像に基づき、画像種別毎の展開表(参照表)を生成して展開表記憶部104に格納する(参照表作成処理)。この処理の詳細については後述する。

30

【0032】

次にS102で入力部302が、プロファイル記憶部102から、色変換対象の画像データを生成した第1のデバイスに対応する入力プロファイル、および該画像データを出力する第2のデバイスに対応する出力プロファイルを取得する(変換条件入力処理)。またS102ではさらに、展開表記憶部104から、S101で作成した全ての画像種別についての展開表を取得する。そしてS103で入力値生成部303が、色変換用の全体LUTにおける全ての入力値を生成する。例えば、入力画像が3チャンネルで、各チャンネル8ビットの画像であれば、全体LUTの入力値として256(=2の8乗)の3乗種類が生成される。そしてS104で色変換部304が、S103で生成した全入力値に対応する出力値を算出する。すなわち、S102で入力した入力プロファイルおよび出力プロファイルに基づき、各入力値を第1の色空間の値から第2の色空間の値に変換する。これにより、全入力値に対する全出力値が揃い、すなわち全体LUTが作成される。

40

【0033】

そしてS105で圧縮部305が、画像種別毎の展開表に基づき、S104で算出した全体LUTにおける全出力値を部分ごとに圧縮することで、画像種別に応じた色変換用の圧縮LUTを作成する。ここで作成された画像種別毎の圧縮LUTは、圧縮LUT記憶部105に格納される。

【0034】

展開表生成処理(S101)

以下、上記図5(a)のS101における展開表生成処理について説明するが、それに先立ち、本実施形態で使用される展開表(参照表)について詳細に説明する。

【0035】

50

本実施形態の色変換処理では、入力値の全組み合わせに対する出力値を画像の種別に応じて部分的にメモリに展開することで、メモリ使用量を削減するとともに色変換速度の向上を図っている。本実施形態では、色変換用の全体LUTを部分的にメモリ展開するため、入力値に対応するLUTの出力値はメモリ展開すべき成分であるか否か、および既にメモリ展開されているか否かを判定する必要がある。そこで本実施形態では、以下に説明するようなビット列からなる展開表を用いることで、該判定を高速に行う。

【0036】

以下、説明を簡単にするため、1次元での色変換を行う場合を例とする。図6(a)に、展開表作成時における各格子の対応例を示す。同図は、入力色空間における1次元の軸を8分割した際の、各分割点(各格子)についての10進表記と2進表記、その上位3ビットの2進表記、10進表記、および該格子についての展開状況を表す展開フラグ、の対応を示している。図6(a)の1列目に示す格子で軸を8つの領域に分割すると、各格子の2進表記は同図2列目の値で表わされる。さらに、各格子の2進表記の上位3ビットを抽出すると、同図3列目の値となり、これを10進表記で表すと、同図4列目の値となる。そして、各領域をメモリに展開すべきか否か、または展開されているか否かを0/1の2値で表すことで、同図5列目のように8つの展開フラグが生成される。これにより、各格子についての展開/非展開を示す情報(展開フラグ)を、8ビットのビット列としてメモリ上で表現することができる。このビット列がすなわち、本実施形態において画像種別ごとに作成・参照される展開表(参照表)であって、各ビットが色変換LUTの入力格子点を示す。すなわち、展開表においてビットが1である格子点は、色変換時に高頻度で参照される高参照部に含まれ、0である格子点はその他の低参照部に含まれる。

【0037】

図6(b)に、図6(a)の5列目に対応する展開表の例を示す。同図において、1042~1049がそれぞれ、各格子の展開フラグを表している。

【0038】

本実施形態では、前処理部103にて上記展開表を作成しておく。そして後処理部114で入力画像の色変換を行う際に、対応する展開表から、画素値(入力値)に応じた展開フラグを取得することで、該入力値の属する部分LUTが既に展開されているか否かを判別する。

【0039】

ここで、後処理部114における展開フラグの具体的な取得方法について、入力値が35である場合を例として、図6(c)を用いて説明する。まず、入力値35(図6(c)1列目)を2進表記に変換し(図6(c)2列目)、該2進表記の入力値の上位3ビットを取得し(図6(c)3列目)、この3ビットを10進表記に変換する(図6(c)4列目)。この変換により得られた値がすなわち、入力値の属する格子番号を示し、この値の分だけ、図6(b)に示す展開表(展開フラグのビット列)を右シフトする。この例では、入力値35は格子番号1番に属し、シフト回数が1回となる。この右シフト処理により、展開表の最下位ビットに、入力値が属する格子の展開フラグが格納されることになる。そこで、展開表のビット列と"0000 0001"との論理積をとることで、最下位ビットに格納された展開フラグを取得することができる。

【0040】

以上、入力値35が属する格子の展開フラグを取得する方法について説明したが、この取得方法を一般化すると、次のようになる。

【0041】

まず、全LUTの各軸の分割数を s 、画像のビット深度を n とすると、入力値の上位ビットを取得するためのシフト回数 t は、以下の式(1)で表すことができる。

【0042】

$$t = n - \log_2 s \quad \dots (1)$$

また、入力値を x とすると、参照すべきビット列の番号 v は、以下の式(2)で表すことができる。なお(2)式において、 $a \ b$ は変数 a の2進表記を b 回右シフトすることを表す。また、 $a \& b$ は変数 a と変数 b の論理積をとることを表す。

【0043】

10

20

30

40

50

$$v = (x \ t) \&(s-1) \quad \dots(2)$$

また、展開フラグを格納するビット列をfとすると、入力値xが属する格子の展開フラグgは、以下の式(3)により算出することができる。

【 0 0 4 4 】

$$g = (f \ v) \&1 \quad \dots(3)$$

後処理部114では以上の演算を行うことによって、前処理部103で作成された展開表のビット列から、入力値が属する格子の展開フラグを取得することができる。そして、この展開フラグの値によって、入力値が属する格子がメモリに展開されているか否かが判定する。

【 0 0 4 5 】

次に、S101における展開表の生成処理について、図7(a)のフローチャートを用いて詳細に説明する。すなわちS301～S311により、サンプル画像記憶部101に保持された全てのサンプル画像に対応する全ての画像種別について、以下のS302～S310による展開表作成処理(参照表作成処理)を行う。

【 0 0 4 6 】

展開表作成処理としてはまずS302で、展開表の各ビットを0で初期化する。ここで0は、対応する格子についての色変換情報(部分LUT)をメモリに展開する必要がない、あるいは、未だ展開されていないことを表す。一方、1は、対応する格子についての色変換情報(部分LUT)をメモリに展開する必要がある、あるいは、既に展開されていることを表す。そしてS303～S305で、サンプル画像記憶部101に保持されている、現在処理中の画像種別に属する全てのサンプル画像を入力し(サンプル入力処理)、全サンプル画像について、各格子に属する値を有する画素を累積し、該累積値をサンプル画像の総画素数で除算することで、該サンプル画像の格子毎の出現確率を算出する。

【 0 0 4 7 】

S306で、当該画像種別に属する全てのサンプル画像における、各格子の出現確率を加算する。そしてS307で、該加算した格子ごとの出現確率を大きい順に整列させ、S308で、該整列した出現確率を、その大きい順に累積する。そしてS309で、累積値が所定の閾値以下となる格子に対し、展開表の該当ビットに1を設定する。これによりすなわち、サンプル画像における出現確率が所定値以上である格子が高参照部に区分され、出現確率が所定値未満である格子が低参照部に区分される。そしてS310で、当該画像種別に対応する展開表を、展開表記憶部104に格納する。このとき展開表には、対応する画像種別を示す識別子等を付与しておく。このS310の処理が全ての画像種別について行われることで、展開表記憶部104には画像種別毎の展開表が格納されて、S101の展開表生成処理が終了する。

【 0 0 4 8 】

なお、S309で参照される累積値の閾値としては、システムに搭載されたメモリの容量や所望する処理速度等に応じた値を予め設定しておけば良い。が大きいほど、S203においてメモリに展開する部分LUTの数が増加するため、メモリ使用量およびS203における処理時間が増加する。反面、予めメモリに展開される部分LUTの種類が増えるため、入力画像に出現する画素に対する網羅率が向上するので、画像変換処理中に部分LUT展開処理が発生しない。これにより、画像変換を高速に行うことができる。一方、が小さいほど、S203においてメモリに展開する部分LUTの数が増加するため、S203におけるメモリ使用量および処理時間を削減することができる。反面、S203でメモリに展開した部分LUTに該当しない画素が入力画像に多く含まれていた場合には、未展開の部分LUTに対する展開処理を都度行うことになるため、メモリアクセスの連続性が損なわれオーバーヘッドが発生する。したがって、の値は、変換画像数、変換画像の画素出現特性などを鑑みて設定することで、より所望のメモリ消費、処理速度で動作させることができる。

【 0 0 4 9 】

圧縮LUT作成処理(S105)

以下、上記図5(a)のS105における、圧縮LUTの作成処理について、図7(b)のフローチャートを用いて詳細に説明する。すなわちS401～S409により、展開表記憶部104に保持され

10

20

30

40

50

た全ての展開表について、以下のS402～S408による圧縮LUT作成処理を行うことで、画像種別毎の圧縮LUTが作成される。

【 0 0 5 0 】

図7(b)ではS402～S408により、処理中である展開表の全ビットについて、以下のS403～S407による圧縮処理を行う。圧縮処理としてはまずS403で、展開表の下位ビットから走査して、同一ビットの並びを検出し、全体LUTから該同一ビット並びに対応する格子の部分LUTを抽出する。例えば図3(b)に示す展開表の例であれば、1042から走査を開始した際、1043までの2ビットが同一ビット並びとして検出され、図3(a)に示す1行目と2行目で示される2格子分の部分LUTが抽出される。

【 0 0 5 1 】

そしてS404で、該部分LUTに属する各LUT成分(出力値)について、所定方向に隣接する成分との差分値を算出する。次にS405で、S403で取得された部分LUTにおける始点成分の出力値を保存する。そしてS406で、S404で算出した各差分値の出現確率に応じて、部分LUTにおける各LUT成分に符号を割り当てる。このとき、差分値の出現確率が大きいほど、より短い符号を割り当てるが、例えばハフマン符号を適用すれば良い。なお、この符号化時に、符号と差分値との対応を示す符号表が作成されるが、これも部分LUTに添付しておく。

【 0 0 5 2 】

そしてS407で、S405、S406で作成された部分LUTの始点成分および符号成分を、圧縮LUTとして圧縮LUT記憶部105に格納する。ここで保持される圧縮LUTは、先に展開表記憶部104に格納されている、該圧縮LUTの作成に関与した展開表と対応付けて保持されることはもちろんである。例えば、展開表に付されている画像種別の識別子等を、圧縮LUTにも付与すれば良い。

【 0 0 5 3 】

なお、上記S403～S407による圧縮処理は展開表の全ビットについて行われ、S403で展開表における同一ビット群として複数ビットが検出された場合、これら複数ビットに対応する部分LUTによってひとつの圧縮LUTが作成される。したがってこの場合、ビット値が異なるビットから、次の処理を開始する。展開表における同一ビットの並びが長いほど、対応する部分LUTのサイズが大きくなるが、この部分LUTがひとつの圧縮LUTとして圧縮されるため、符号化効率は高くなる。

【 0 0 5 4 】

以上のように本実施形態では画像種別ごとに、全体LUTにおける各格子点の出力値を、展開表が示す高参照部ごと、および低参照部ごとにデータ圧縮することで、全体LUTを部分単位で圧縮した圧縮LUTが作成される。

【 0 0 5 5 】

なお本実施形態では、展開表と圧縮LUTを前処理部103において作成する例を示したが、展開表の作成と圧縮LUTの作成とを分けることも可能である。まず展開表については、例えば他装置において、複数のサンプル画像に基づいて作成しておくことが可能である。そして圧縮LUTは、該展開表と入出力プロファイルに基づいて作成される。すなわち印刷システムとしては色変換の実行時に展開表と圧縮LUTが備わっていれば良いため、他装置において作成された展開表と圧縮LUTを入力して用いることができる。この場合、前処理サーバ10に大量のサンプル画像を保持するためのサンプル画像記憶部101を備える必要がなくなり、同様にプロファイル記憶部102、および前処理部103も不要となる。

【 0 0 5 6 】

後処理(色変換処理)

以下、本実施形態における後処理について、図5(b)を用いて説明する。図5(b)は、後処理サーバ11の後処理部114において行われる色変換処理を示すフローチャートである。まずS201で画像入力部401が、色変換の対象となる入力画像と、その画像種別の情報を取得する。なお、画像種別は、画像データに関連付けられたカラープロファイルのマッピング方法(インテント)によって検出される。例えば、インテントが"Saturation"であった場合

10

20

30

40

50

には、該画像データはグラフ等のグラフィックス画像であることが想定される。また、コンテンツが"Perceptual"であった場合には、さらに画像のタグに格納されている撮影モードを参照することで、「ビーチ」「紅葉」「ポートレート」等の画像種別を知ることができる。そしてS202で圧縮LUT入力部402が、展開表記憶部104および圧縮LUT記憶部105から、S201で取得した画像種別に応じた展開表と圧縮LUTを取得する(テーブル取得処理)。

【0057】

そしてS203で展開部404が、圧縮LUTにおいて、展開表によって展開すべき旨が示されている部分LUT、すなわち展開フラグが1である全ての部分LUTを伸張してメモリに展開する(部分伸張処理)。

【0058】

その後、S204～S208において、入力画像の全画素について、以下のS205～S207による色変換処理を行う。色変換処理としてはまずS205で展開判定部403が、注目画素に対応する格子点(注目格子点)が属する部分LUTが、後処理部114の内部メモリに既に展開されているか、すなわち注目格子点の圧縮データが既に伸張されているか否かを判定する。この判定処理の詳細については後述する。

【0059】

S205で、入力画素値が属する部分LUTが既に展開されていると判定されればそのままS206に進むが、展開されていないと判定されるとS206へ進み、展開部404が、対応する部分LUTを展開する(注目伸張処理)。この展開処理の詳細については後述する。

【0060】

そしてS207で画像変換部405が、入力画素値に対応する出力値を、展開済みである部分LUTから取得する。これによりすなわち、第1の色空間で表現された入力画素値が、第2の色空間で表現される出力画素値に変換される。そしてS201～S208で、入力画像の全画素についての色変換処理が終了したと判定されたら、S209で画像出力部406が、色変換後の画像を出力する。

【0061】

展開判定処理(S205)

以下、上記図5(b)のS205における、入力画素値が属する部分LUTが展開されているか否かを判定する処理について、図8(a)のフローチャートを用いて詳細に説明する。まずS501で、入力画素の上位ビットを取得する。ここで取得するビット数 m は、LUTにおける入力色空間の各軸の分割数 s に応じて決定され、例えば、 $m = \log_2 s$ である。したがって、入力色空間の各軸を8分割して作成されたLUTであれば、上位3ビットが取得される。そしてS502で、該取得した上位ビットで示される値の分だけ、S202で取得した展開表を右シフトする。例えば、上位3ビットが"100"であれば、シフト回数は4回となる。そしてS503で、シフトした展開表のビット列と1との論理積をとることで、シフト後の展開表における最下位ビットを取得し、これを判定結果として出力する。すなわち、シフト後展開表における最下位ビットが"1"であれば対応する部分LUTが展開されている旨を示し、"0"であれば未だ展開されていない旨を示す。

【0062】

展開処理(S206)

以下、上記図5(b)のS206における部分LUTの展開処理について、図8(b)のフローチャートを用いて詳細に説明する。まずS601で、展開対象である部分LUTの始点成分を取得し、S602で該始点成分をメモリに書き出す。そしてS603～S606により、当該部分LUTの全ての符号成分について順次、以下のS604, S605による伸張処理を施すことで、メモリ上にLUT成分値(出力値)が展開される。

【0063】

S604では符号表に基づき、符号成分に対応する差分値を取得する。そしてS605で、直前に展開されたLUT成分値に、S604で取得した差分値を加算し、該加算結果を現符号成分を展開したLUT成分値としてメモリに書き出す。S604, S605による伸張処理が部分LUTの全符号成分について終了したら、S607で展開表の該当ビットは0であるから、これを1に更新す

10

20

30

40

50

る。

【0064】

以上説明したように本実施形態によれば、画像種別に応じて、全体LUTを部分ごとに圧縮した圧縮LUTと展開表を予め作成しておく。そして、処理対象となる画像の種別に応じて、全体LUTのうち展開表で示される高参照部のみをメモリに展開する。これにより、全体LUTにおける必要部分のみが展開され、該展開された部分LUTを用いた色変換を行うことで、メモリ使用量を抑えるとともに高速な色変換を行うことができる。

【0065】

また本実施形態では、展開表において連続する高参照部を一単位として圧縮するため、展開時にもその部分LUTは一括処理される。したがって、使用頻度の高い部分LUTの展開にかかる負荷を最小限に抑えることができ、色変換をより高速に行うことができる。

【0066】

<第2実施形態>

以下、本発明に係る第2実施形態について説明する。印刷システムにおいては、印刷ジョブの優先度に応じて特定の印刷ジョブ(特急ジョブ)を直ちに処理し、印刷物として出力する旨が要求される場合がある。第2実施形態ではこのような要求に対応するため、メモリの空き容量に余裕がある場合であれば、該メモリ容量を大量に消費してでも特急ジョブの処理を優先することで、高速処理を実現する。すなわち、印刷ジョブに予め設定された特急フラグの内容とメモリの空き容量に応じて、当該印刷ジョブに対する印刷速度を切り替えることを特徴とする。

【0067】

第2実施形態における印刷システムは、上述した第1実施形態とほぼ同様の構成からなるが、後処理部114における色変換処理のみが異なる。したがって第2実施形態の後処理部を後処理部30とし、他の構成には第1実施形態と同一番号を付して説明を省略する。

【0068】

論理構成

以下、第2実施形態の印刷システムにおける論理構成を図9に示し、説明する。なお、前処理部103における処理は第1実施形態と同様であるため、ここでは後処理部30における構成のみを示す。なお図9において、第1実施形態の図4(b)と同様の構成には同一番号を付し、説明を省略する。図9において、501はメモリの空き容量を取得する空き情報取得部、502はメモリの空き容量に応じて色変換LUTの全LUT成分をメモリ展開するか否かを判定する全展開判定部、503は全LUT成分をメモリに展開する全展開部、である。

【0069】

後処理(色変換処理)

以下、第2実施形態の後処理部30における色変換処理について、図10のフローチャートを用いて説明する。なお図10において、第1実施形態で示した図5(b)のフローチャートと同様の処理工程には同一ステップ番号を付す。

【0070】

まずS201で画像入力部401が入力画像とその画像種別情報を取得するが、第2実施形態ではさらに、該入力画像に関連付けられた特急フラグを取得する。ここで特急フラグとは、当該入力画像に特急処理対象であるか否かを示すフラグである。そしてS202で圧縮LUT入力部402が、展開表記憶部104および圧縮LUT記憶部105から、画像種別に応じた展開表と圧縮LUTを取得する。

【0071】

そしてS701で全展開判定部502が、特急フラグが1であるか否かを判定し、1であれば圧縮LUTの全展開を行うためにS702へ進み、0であれば圧縮LUTの部分展開を行うためにS203に進む。

【0072】

S702では空き情報取得部501が、システムにおける現在のメモリの空き容量を取得する(空き容量取得処理)。そしてS703で全展開判定部502が、該空き容量が所定の必要容量以上

10

20

30

40

50

であるか否かを判定する(容量判定処理)。ここで必要容量とは、圧縮LUTの全格子点を展開した全体LUTを保持可能な容量であり、例えば全体LUTが c チャンネル、 1 成分 n ビットである場合、必要容量 $r=c \times 2n$ として算出される。メモリの空き容量が必要容量以上である場合にはS704に進み、圧縮LUTの全展開処理(全伸張処理)を行う。すなわちS704では全展開部503が、圧縮LUTから全てのLUT成分をメモリに展開することで、全体LUTを取得する。その後、S207で画像変換部405が、該全体LUTから、入力画像における全入力画素値に対応する出力値を取得することで、入力画像全体に対する色変換がなされ、S208で画像出力部406が、色変換後の画像を出力する。

【0073】

一方、S703でメモリの空き容量が必要容量未満である場合にもS203に進む。S203以降では、上述した第1実施形態と同様に、圧縮LUTの部分展開を行う。すなわちS203～S207で展開表に基づいて圧縮LUTを部分展開して部分LUTを取得し、色変換を行った後、S208で色変換後の画像を出力する。

10

【0074】

このように第2実施形態では、特急処理すべき印刷ジョブについては圧縮LUTを全展開した全体LUTを用いることで高速処理を行い、そうでない印刷ジョブについては第1実施形態と同様に圧縮LUTを部分展開した部分LUTを用いた省メモリ処理を行う。第1実施形態でも説明したように、部分LUTおよび展開表を参照した色変換処理では、入力画像の画素ごとに、部分LUTが展開済みであるか否かを確認する必要があるため、相応の処理時間がかかる。これに対し全体LUTを参照した色変換処理では、最初に全展開処理を行う必要があるものの、展開後は高速処理が可能となるため、部分展開時に対して処理時間が大幅に短縮される。これは、処理対象となる入力画像の画素数が多いほど顕著である。

20

【0075】

なお第2実施形態では、全展開を行うか否かを特急フラグに基づいて判定する例を示したが、メモリの空き容量に基づいて自動的に判定することもできる。例えば、メモリ空き容量に余裕があれば、常に全展開を行って処理速度を優先するようにしても良い。また、図11に示すように、メモリ使用状況と当該印刷ジョブの優先度を設定するチェックボックスを備えたUIをユーザに提示し、当該印刷ジョブを優先的に処理するか否かをユーザに指定させるようにしても良い。このようなUIを提示することにより、オペレータの判断によって指定の印刷ジョブを高速に印刷出力することができる。

30

【0076】

以上説明したように第2実施形態によれば、印刷ジョブの優先度とメモリの空き状況に応じて、メモリに展開するLUT成分を制御することにより、優先度の高い印刷ジョブを高速に印刷することができる。

【0077】

< 第3実施形態 >

以下、本発明に係る第3実施形態について説明する。上述した第1および第2実施形態の印刷システムでは、画像種別に応じた圧縮LUTを予め作成し、圧縮LUT記憶部105で保持しておく例を示した。なお、この圧縮LUTは入出力プロファイルに応じて生成されているため、入力装置や出力装置および印刷用紙、およびマッピング方法などにも依存していることは言うまでもない。上記各実施形態において、印刷対象として入力された画像データに添付された画像種別情報に対し、適切な圧縮LUTが保持されていない場合には、圧縮LUTを新規に作成する必要がある。印刷システムにおいては通常、後処理サーバ11においてRIP部115の前段に印刷ジョブをストックしておくための不図示のプールが設けられている。したがって、ある印刷データを処理する際に、印刷ジョブのプール量が多い場合には、当該印刷データをそのままプールしてもRIP開始までに圧縮LUTを準備することができる。しかしながら、プール量が少ない場合には、圧縮LUTの準備がRIP開始までに間に合わず、結果的に印刷動作が停止してしまうことが想定される。

40

【0078】

そこで第3実施形態では、新規に圧縮LUTを作成する必要がある印刷ジョブについて、そ

50

の作成時間と、RIP開始までの待ち時間(以下、スプール期間)に応じて、該印刷ジョブの処理順を制御する。これにより、圧縮LUTの作成を効率的に行い、印刷動作の停止を防止することができる。すなわち、圧縮LUTの作成時間と当該印刷ジョブのスプール期間を取得し、作成時間の方がスプール期間よりも長い場合には、当該印刷ジョブの処理順が後となるように変更する。これにより、RIP開始までのスプール期間が延長され、該スプール期間内で圧縮LUTの作成を完了させることができる。

【0079】

圧縮LUT一覧表および予想処理時間表

上述した第1実施形態では図5(a)のS104において、入力プロファイルおよび出力プロファイルに基づき、色変換における全入力値に対する全出力値、すなわち色変換LUTの全成分を算出して全体LUTを作成する例を示した。第3実施形態ではS104におけるLUT出力値の算出を、入出力プロファイルと、入力側および出力側での観察条件、およびマッチング方法、の組み合わせ(以下、色変換条件)に基づいて行うとする。そして、これら組み合わせと、作成された圧縮LUTとの対応を示す圧縮LUT一覧表を作成し、圧縮LUT記憶部105に格納する。ここで図12(a)に、圧縮LUT一覧表の例を示す。同図に示すように圧縮LUT一覧表は、入力プロファイル、出力プロファイル、入力側観察条件、出力側観察条件、マッチング方法、の組み合わせによる色変換条件と、該色変換条件を用いて作成・保持された圧縮LUT、から構成される。すなわち圧縮LUT一覧表には、圧縮LUT記憶部105に保持されている全ての圧縮LUTの情報が記録されている。

10

【0080】

第3実施形態ではさらに、マッチング方法と、圧縮LUTを作成するのに要する予想処理時間との対応表である予想処理時間表を予め作成し、例えば圧縮LUT記憶部105内に格納しておく。図12(b)に、予想処理時間表の例を示す。第3実施形態では、後述するS803で圧縮LUTの作成時間を算出する際に、この予想処理時間表を参照することで、入力画像の色変換に必要な処理時間を取得することができる。

20

【0081】

論理構成

第3実施形態における印刷システムは、上述した第1実施形態とほぼ同様の構成からなるが、後処理部114における処理のみが異なる。したがって第3実施形態の後処理部を後処理部50とし、他の構成には第1実施形態と同一番号を付して説明を省略する。

30

【0082】

以下、第3実施形態の後処理部50における論理構成を図13(a)に示し、説明する。なお図13(a)において、第1実施形態の図4(b)と同様の構成には同一番号を付し、説明を省略する。図13(a)において、701は入力画像のマッチング方法に応じて圧縮LUTを作成するのに要するであろう予想作成時間(圧縮LUT作成時間)を算出し、かつ印刷が開始されるまでの時間(スプール期間)を取得する時間算出部である。また、702はスプール期間と圧縮LUT作成時間を比較し、印刷ジョブの順序を変更するか否かを判定する判定部、703は印刷ジョブの順序を変更する整列部、704は色変換条件に基づき圧縮LUTを作成する圧縮LUT作成部、である。

【0083】

後処理(色変換処理)

以下、第3実施形態の後処理部50における色変換処理について、図13(b)のフローチャートを用いて説明する。なお、図13(b)において上述した第1実施形態で示した図5(b)のフローチャートと同様の処理工程には同一ステップ番号を付す。

40

【0084】

まずS201で画像入力部401が、印刷ジョブに含まれる入力画像とその種別情報を取得する。そしてS801～S804で、印刷ジョブに含まれる全ての入力画像について、S802、S803による圧縮LUT作成時間の算出を行う。すなわちS802で圧縮LUT入力部402が、入力画像に添付された色変換条件に基づき、図12(a)に示す圧縮LUT一覧表から、対応する圧縮LUTを探査する。これによりすなわち、該圧縮LUTが圧縮LUT記憶部105に保持されているか否かが

50

判定される(保持判定処理)。そして、圧縮LUTが探索できない、すなわち該圧縮LUTが圧縮LUT記憶部105に未だ保持されていない場合にはS803に進み、時間算出部701が、対応する圧縮LUTの作成時間を算出する。S803ではすなわち、入力画像のマッチング方法に対応した予想処理時間を、図12(b)に示す予想処理時間表から取得し、これを当該印刷ジョブにおける圧縮LUT作成時間として累積する。一方、S802で圧縮LUTが探索できた場合には、上記S803における圧縮LUT作成時間の累積を行わない。

【0085】

S801～S804で圧縮LUT作成時間が算出されると、次にS805において判定部702が、当該印刷ジョブについてのスプール期間と、圧縮LUT作成時間を比較する。なお、スプール期間すなわちRIP開始時間(印刷開始時間)に関する情報は、後処理サーバ11における不図示の制御部(スプール管理部)から取得することができる。圧縮LUT作成時間がスプール期間よりも長い場合にはS806に進み、整列部703が、当該印刷ジョブの処理順序をスプールの最後尾に移動した後、S807に進む。一方、S805で圧縮LUT作成時間がスプール期間に満たない場合にはそのままS807に進む。S807では圧縮LUT作成部704が、プロファイル、観察条件、マッピング方法等の色変換条件に基づき、当該入力画像に対する圧縮LUT、およびその展開表を作成する。

10

【0086】

なお、ここではS806での処理順変更により、S807での圧縮LUTおよび展開表の作成に十分なスプール期間が確保されるものとして説明した。実際には、S806で印刷ジョブの処理順を最後尾に変更した後であっても、まだスプール期間が十分でない場合も考えられる。このような場合には、何らかのエラー処理を適用することが適当であると考えられる。

20

【0087】

その後S808において、上述した第1実施形態の図5(b)と同様に、圧縮LUTを部分展開して入力画像の色変換を行う。

【0088】

以上説明したように第3実施形態によれば、印刷ジョブのRIP開始時に印刷条件に合致した圧縮LUTが存在しなかった場合に、該圧縮LUTの作成時間に応じて印刷ジョブの処理順序を変更することで、印刷処理が停止してしまうことを回避できる。

【0089】

なお、第3実施形態では、印刷ジョブのスプール順を変更するか否かの判定を、圧縮LUT作成時間と印刷待ち時間に基づいて自動判定する例を示した。この判定を、図12(c)に示すようなUIによって行うようにしても良い。すなわち該UIによれば、圧縮LUT作成時間と印刷待ち時間(印刷開始時間)をユーザに表示し、当該印刷ジョブを直ちにスプールするかをユーザに判断させ、入力された判断結果に基づき、印刷ジョブの順序を変更する。このようなUIを提示することにより、オペレータの判断によって指定の印刷ジョブを高速に印刷することができる。

30

【0090】

また、マッチング方法に対する予想処理時間表を参照して圧縮LUTの作成時間を予想する例を示したが、他の方法によって圧縮LUT作成の予想時間を取得しても良い。例えば、何色かのサンプル色を選択して該サンプル色の変換を実行し、該変換に要した時間から、実際に圧縮LUTを作成する際に要するであろう処理時間を予想するようにしても良い。

40

【0091】

<他の実施形態>

上述した各実施形態においては、色変換用の全体LUTの入力値として、入力画像の各チャンネルの取り得る値の全ての組み合わせを生成する例を示した。しかしながら、本発明における全体LUTの入力値は該「全ての組み合わせ」に限定されず、例えば1点おきなど適当な点数おき、すなわち所定の間隔おきで入力値を生成するようにしても良い。その場合、生成されていない入力値に対する出力値をLUTから得るには、入力値を囲む格子点に対応する出力値の平均値や加重平均値などをその出力値とすれば良い。

【0092】

50

また、上述した各実施形態では、入力画像をRGB色空間のデータとして説明したが、sRGBやAdobeRGB色空間等の色空間のデータであっても良い。

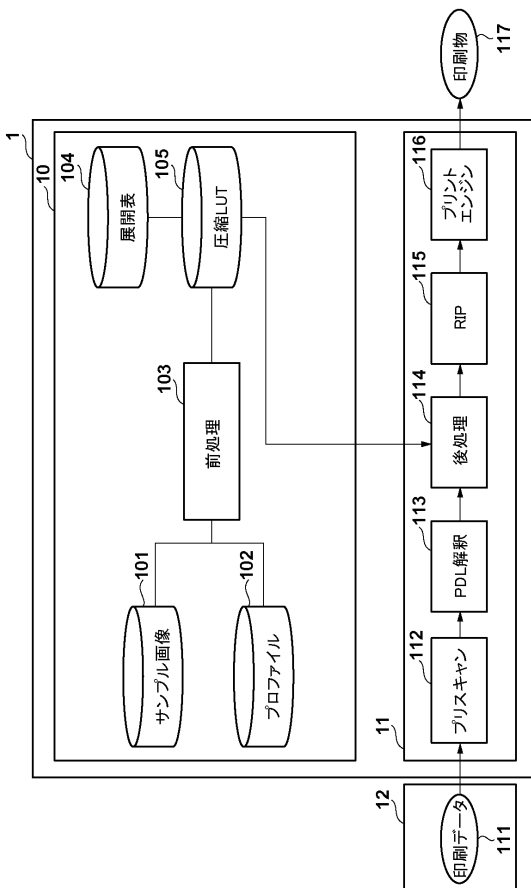
【0093】

また各実施形態が図1に示す印刷システムにおいて実現される例を示したが、本発明は図14に示すようなクラウドシステムとしても実現可能である。図14において、2がクラウドシステムであり、転送サーバ12から転送された印刷データ111が、後処理サーバ13内の後処理部114で第1実施形態と同様の色変換処理が施された後、店舗や印刷会社に設置されているプリントサーバ14に出力される。他の構成については図1と同様であるため説明を省略する。

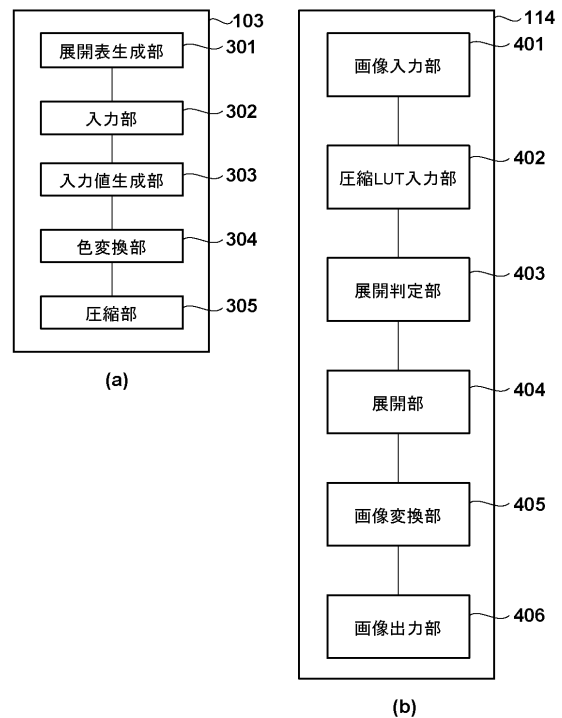
【0094】

また、本発明は、以下の処理を実行することによっても実現される。即ち、上述した実施形態の機能を実現するソフトウェア(プログラム)を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステム或いは装置のコンピュータ(またはCPUやMPU等)がプログラムを読み出して実行する処理である。

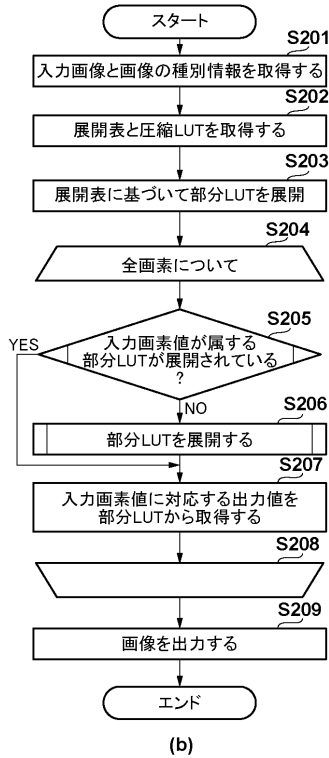
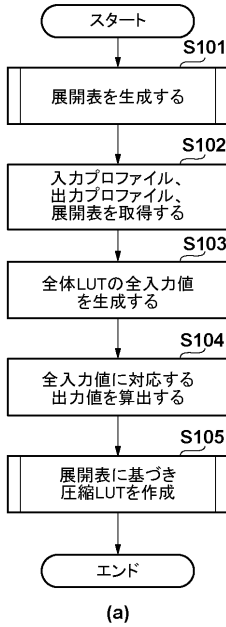
【図1】



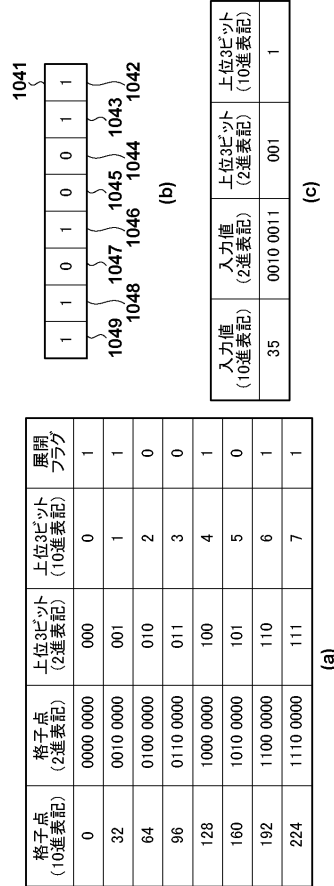
【図4】



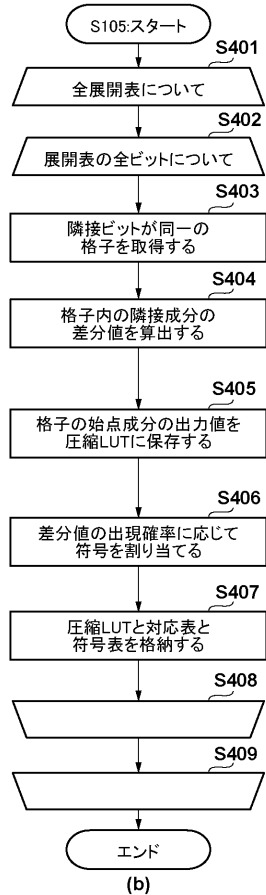
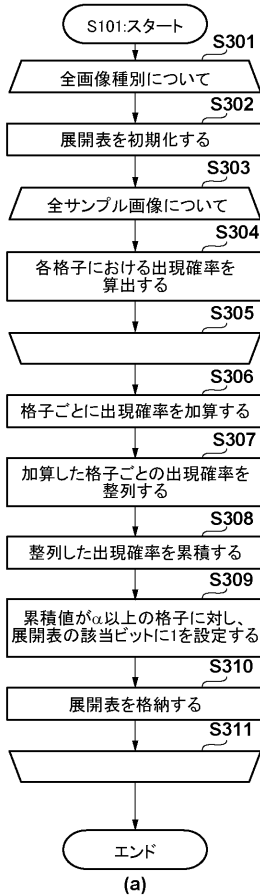
【 図 5 】



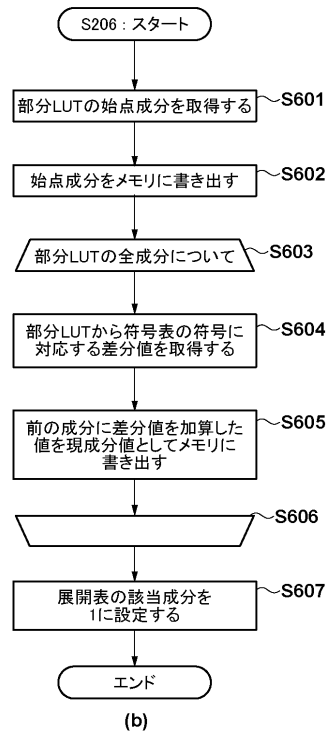
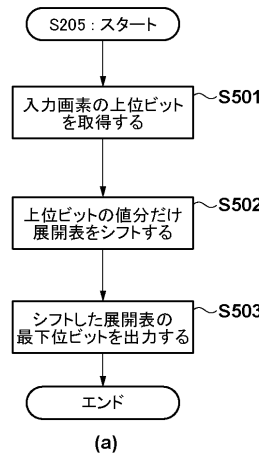
【 図 6 】



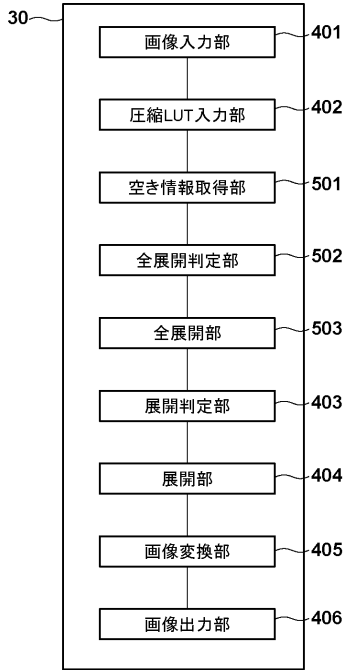
【 図 7 】



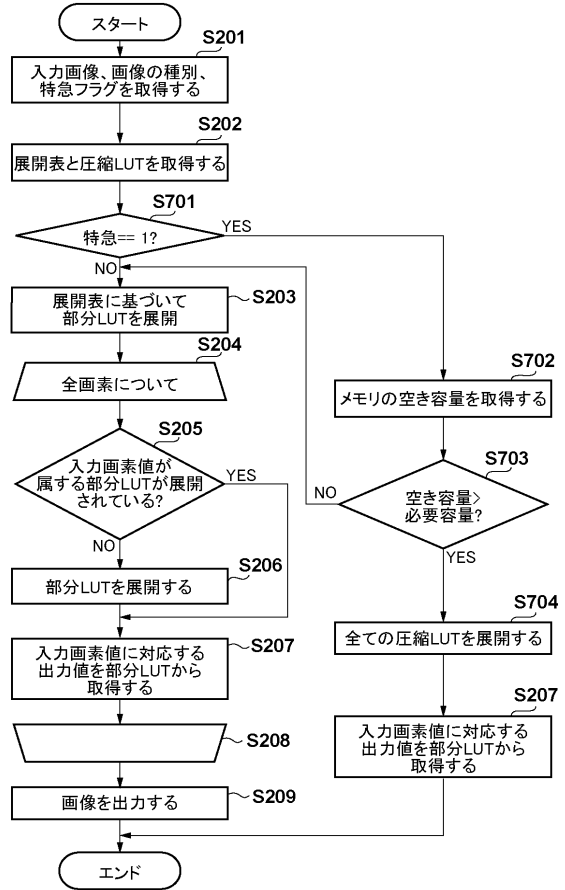
【 図 8 】



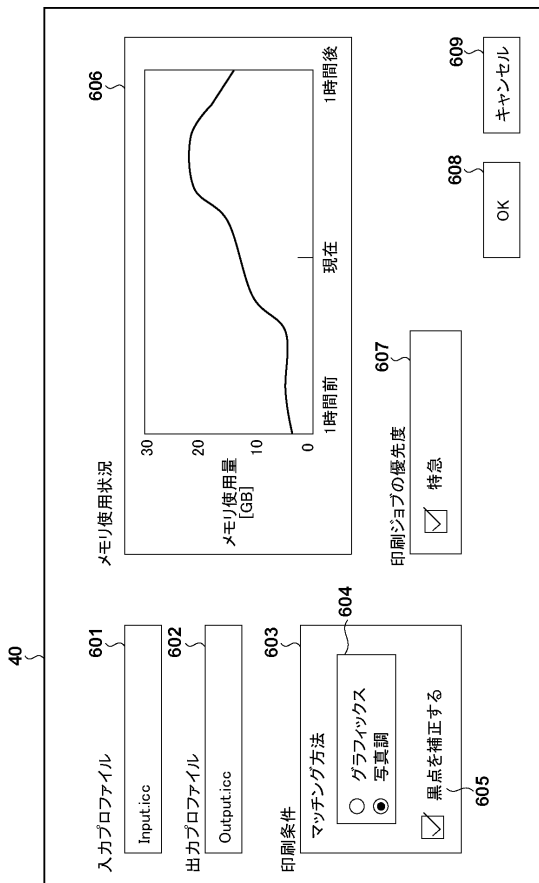
【図9】



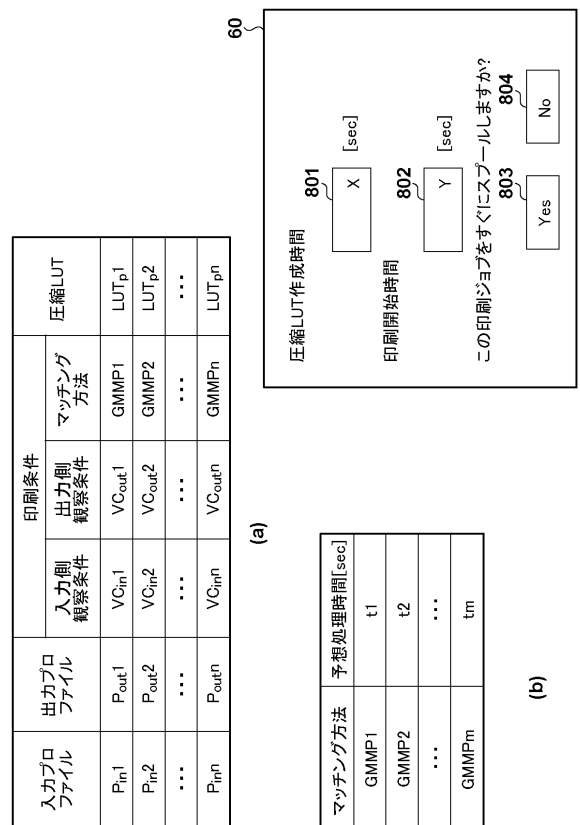
【図10】



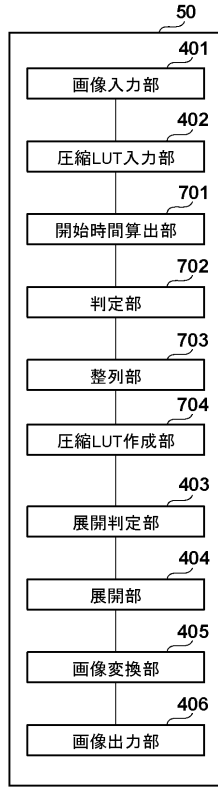
【図11】



【図12】

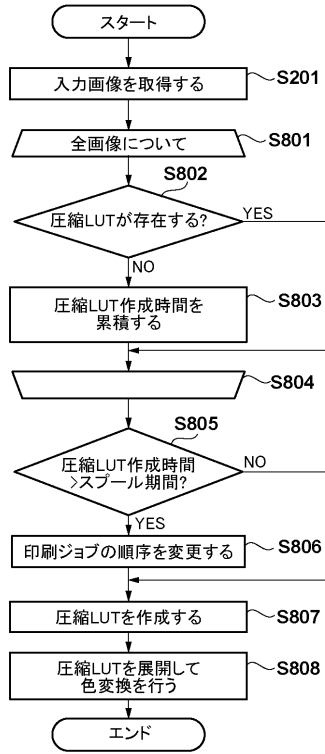


【図 1 3】

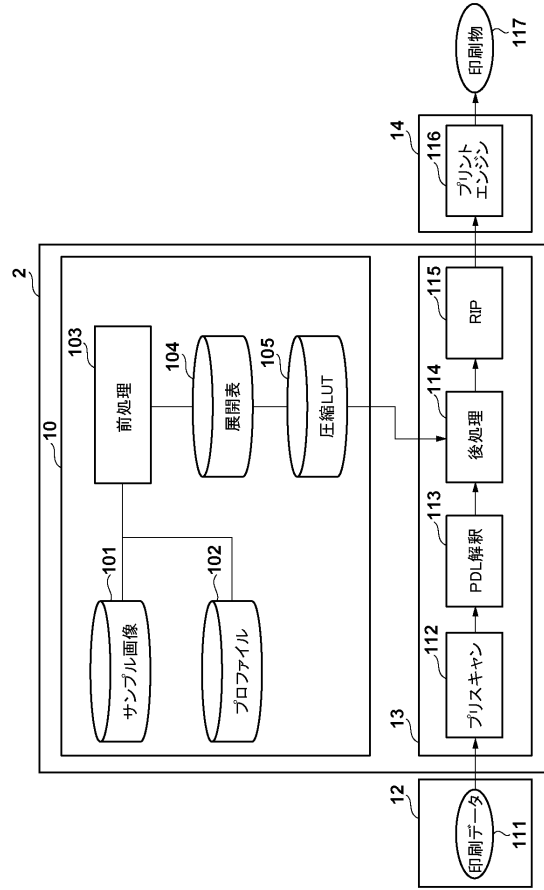


(a)

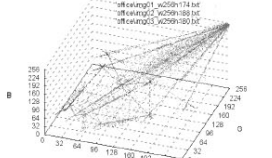
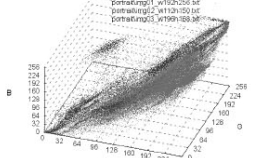
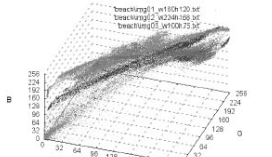
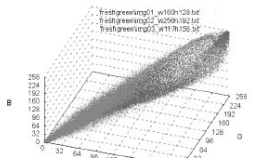
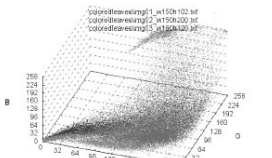
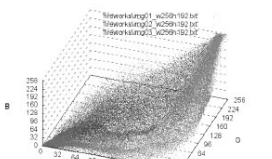
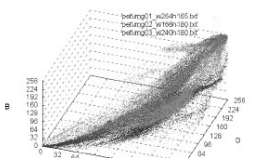
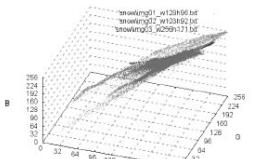
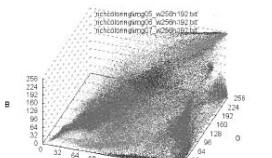
【図 1 4】



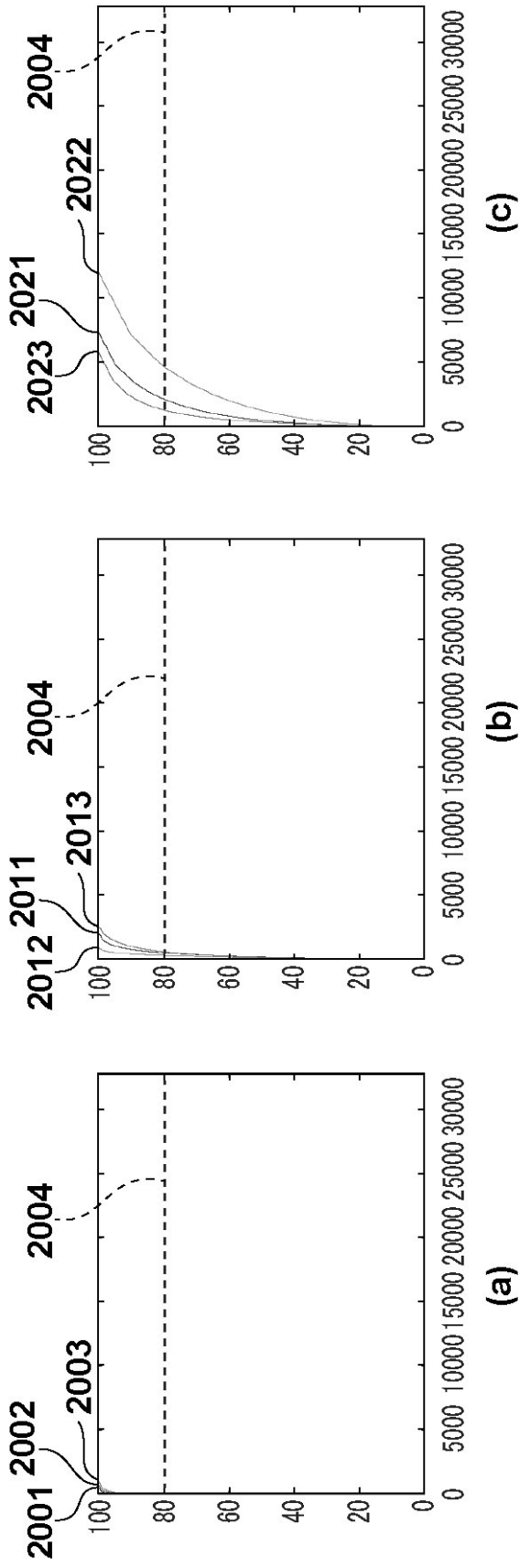
(b)



【 図 2 】

グラフィックス	 <p>3D plot showing a distribution of data points in a 3D space with axes R, G, and B. The plot is titled with three lines: 'graf_colorimg1_w10291174 bit', 'graf_colorimg2_w10291188 bit', and 'graf_colorimg3_w10291180 bit'. The axes range from 0 to 256.</p>
ポートレート	 <p>3D plot showing a distribution of data points in a 3D space with axes R, G, and B. The plot is titled with three lines: 'port_colorimg1_w11921258 bit', 'port_colorimg2_w11921200 bit', and 'port_colorimg3_w11991188 bit'. The axes range from 0 to 256.</p>
ビーチ	 <p>3D plot showing a distribution of data points in a 3D space with axes R, G, and B. The plot is titled with three lines: 'beach_colorimg1_w11001130 bit', 'beach_colorimg2_w12201185 bit', and 'beach_colorimg3_w10291213 bit'. The axes range from 0 to 256.</p>
新緑	 <p>3D plot showing a distribution of data points in a 3D space with axes R, G, and B. The plot is titled with three lines: 'shinryu_colorimg1_w11001130 bit', 'shinryu_colorimg2_w11001130 bit', and 'shinryu_colorimg3_w11701155 bit'. The axes range from 0 to 256.</p>
紅葉	 <p>3D plot showing a distribution of data points in a 3D space with axes R, G, and B. The plot is titled with three lines: 'akikoba_colorimg1_w11001130 bit', 'akikoba_colorimg2_w11001130 bit', and 'akikoba_colorimg3_w11001130 bit'. The axes range from 0 to 256.</p>
花火	 <p>3D plot showing a distribution of data points in a 3D space with axes R, G, and B. The plot is titled with three lines: 'hanabi_colorimg1_w10291192 bit', 'hanabi_colorimg2_w10291192 bit', and 'hanabi_colorimg3_w10291192 bit'. The axes range from 0 to 256.</p>
ペット	 <p>3D plot showing a distribution of data points in a 3D space with axes R, G, and B. The plot is titled with three lines: 'pet_colorimg1_w10291185 bit', 'pet_colorimg2_w11891180 bit', and 'pet_colorimg3_w10291180 bit'. The axes range from 0 to 256.</p>
雪景色	 <p>3D plot showing a distribution of data points in a 3D space with axes R, G, and B. The plot is titled with three lines: 'yuki_colorimg1_w11201166 bit', 'yuki_colorimg2_w11201166 bit', and 'yuki_colorimg3_w10291171 bit'. The axes range from 0 to 256.</p>
極彩色	 <p>3D plot showing a distribution of data points in a 3D space with axes R, G, and B. The plot is titled with three lines: 'kyokushoku_colorimg5_w10291192 bit', 'kyokushoku_colorimg6_w10291192 bit', and 'kyokushoku_colorimg7_w10291192 bit'. The axes range from 0 to 256.</p>

【 図 3 】



フロントページの続き

(72)発明者 玉川 慶

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

Fターム(参考) 5B057 AA11 CA01 CA08 CA12 CA16 CB01 CB08 CB12 CB16 CE16
CG01 CH07 CH11 CH18 DA17 DB02 DB06 DB09 DC25
5C077 LL17 MP08 PP32 PP33 PP37 PP65 PQ08 PQ21 PQ22 RR21
TT02
5C079 HB01 HB03 LA05 LA27 LA31 LB01 MA02 MA04 MA06 NA10
PA03