

(19) 日本国特許庁(JP)

## (12) 特許公報(B2)

(11) 特許番号

特許第5701030号  
(P5701030)

(45) 発行日 平成27年4月15日(2015.4.15)

(24) 登録日 平成27年2月27日(2015.2.27)

(51) Int.Cl.

H04N 1/405 (2006.01)

F 1

H04N 1/40

C

請求項の数 20 (全 21 頁)

(21) 出願番号 特願2010-271696 (P2010-271696)  
 (22) 出願日 平成22年12月6日 (2010.12.6)  
 (65) 公開番号 特開2012-124584 (P2012-124584A)  
 (43) 公開日 平成24年6月28日 (2012.6.28)  
 審査請求日 平成25年12月6日 (2013.12.6)

(73) 特許権者 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 100126240  
 弁理士 阿部 琢磨  
 (74) 代理人 100124442  
 弁理士 黒岩 創吾  
 (72) 発明者 市橋 幸親  
 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  
 審査官 堀井 啓明

最終頁に続く

(54) 【発明の名称】画像処理装置、画像処理方法、コンピュータプログラム

## (57) 【特許請求の範囲】

## 【請求項1】

画素の挿抜を実行することで画像の変倍を行う画像処理装置であって、  
 変倍方向に直交する方向の第1の位置における第1の挿抜位置を決定する第1の決定手段と、

前記直交する方向の前記第1の位置とは異なる第2の位置における第2の挿抜位置を、  
前記決定された第1の挿抜位置に基づいて決定する第2の決定手段と、

前記決定された第1および第2の挿抜位置に基づいて、画素の挿抜を前記画像に対して  
 実行する挿抜手段と、

を有し、

前記決定された第2の挿抜位置は、前記決定された第1の挿抜位置からの前記変倍方向  
の距離が第1の所定の距離以上、該第1の所定の距離より大きい第2の所定の距離以内と  
なる範囲内のいずれかの位置であることを特徴とする画像処理装置。

## 【請求項2】

前記決定された第2の挿抜位置が、前記決定された第1の挿抜位置からの前記変倍方向  
の距離が第1の所定の距離以上、該第1の所定の距離より大きい第2の所定の距離以内と  
なる範囲内のいずれかの位置であるとは、

前記決定された第2の挿抜位置の前記変倍方向の位置が、前記決定された第1の挿抜位  
置の前記変倍方向の位置からの前記変倍方向の距離が前記第1の所定の距離以上、前記第  
2の所定の距離以内となる範囲内のいずれかの位置であることを特徴とする請求項1の画

像処理装置。

【請求項 3】

前記直交する方向の第1および第2の位置それは、前記直交する方向における画素の位置に対応し、

前記変倍方向の第1および第2の所定の距離それぞれの単位は、画素である請求項1または2の画像処理装置。

【請求項 4】

前記第1の決定手段は、前記第1の位置において前記変倍方向に所定の数連続して並ぶ複数の画素からなる第1の画素列のうちの1つの画素の位置を、前記第1の挿抜位置として決定し、

10

前記第2の決定手段は、前記第2の位置において前記変倍方向に所定の数連続して並ぶ複数の画素からなる第2の画素列のうちの1つの画素の位置を、前記第2の挿抜位置として決定することを特徴とする請求項1乃至3の何れか1項の画像処理装置。

【請求項 5】

前記第1の画素列と前記第2の画素列とは、前記直交する方向で互いに隣接することを特徴とする請求項4の画像処理装置。

【請求項 6】

前記所定の範囲の両端の位置のうちの前記第1の挿抜位置に近い位置は、前記第1の所定の距離によって定まり、前記所定の範囲の両端の位置のうちの前記第1の挿抜位置から遠い位置は、前記第2の所定の距離によって定まることを特徴とする請求項1乃至5の何れか1項の画像処理装置。

20

【請求項 7】

前記挿抜手段は、前記第1の挿抜位置および前記第2の挿抜位置の画素を前記画像から間引くことで、前記画像の縮小を行うことを特徴とする請求項1乃至6の何れか1項の画像処理装置。

【請求項 8】

前記挿抜手段は、前記第1の挿抜位置および前記第2の挿抜位置の画素を前記画像に挿入することで、前記画像の拡大を行うことを特徴とする請求項1乃至6の何れか1項の画像処理装置。

【請求項 9】

30

前記挿抜手段による画素の挿抜が行われた画像を印刷する印刷手段をさらに有する請求項1乃至8の何れか1項の画像処理装置。

【請求項 10】

乱数を生成する生成手段を有し、

前記第2の決定手段は、前記決定された第1の挿抜位置および前記生成された乱数に基づいて前記第2の挿抜位置を決定することで、前記決定された第2の挿抜位置を、前記決定された第1の挿抜位置からの前記変倍方向の距離が第1の所定の距離以上、該第1の所定の距離より大きい第2の所定の距離以内となる前記範囲内のいずれかの位置に決定する請求項1乃至9の何れか1項の画像処理装置。

【請求項 11】

40

前記乱数は、乱数を生成する関数に従って生成されることを特徴とする請求項10の画像処理装置。

【請求項 12】

前記決定された第1の挿抜位置からの前記変倍方向の距離が前記第1の所定の距離以上、前記第2の所定の距離以内の前記範囲は、所定の情報に従って決まることを特徴とする請求項1乃至11の何れか1項の画像処理装置。

【請求項 13】

前記第1および第2の所定の距離は、前記挿抜手段による画素の挿抜が行われた画像が印刷される用紙種に関する情報に従って決まることを特徴とする請求項1乃至12の何れか1項の画像処理装置。

50

**【請求項 1 4】**

前記第1および第2の所定の距離は、前記挿抜手段による画素の挿抜が行われた画像が印刷される用紙種を特定するために用いられる情報に従って決まることを特徴とする請求項1乃至12の何れか1項の画像処理装置。

**【請求項 1 5】**

前記第1および第2の所定の距離は、前記画像の面内情報に従って決まることを特徴とする請求項1乃至12の何れか1項の画像処理装置。

**【請求項 1 6】**

前記第2の決定手段は、前記第2の位置を含むN個の前記直交する方向の位置に対する挿抜位置を、当該N個の挿抜位置の前記変倍方向における位置が前記第2の挿抜位置の前記変倍方向における位置と同じになるように決定することを特徴とする請求項1乃至15の何れか1項の画像処理装置。

10

**【請求項 1 7】**

前記Nは、前記挿抜手段による画素の挿抜が行われた画像が印刷される用紙種に関する情報に従って決まることを特徴とする請求項16の画像処理装置。

**【請求項 1 8】**

前記Nは、前記画像の面内情報に従って決まることを特徴とする請求項16の画像処理装置。

**【請求項 1 9】**

画素の挿抜を実行することで画像の変倍を行う画像処理方法であって、  
変倍方向に直交する方向の第1の位置における第1の挿抜位置を決定する第1の決定工程と、

20

前記直交する方向の前記第1の位置とは異なる第2の位置における第2の挿抜位置を、  
前記決定された第1の挿抜位置に基づいて決定する第2の決定工程と、

前記決定された第1および第2の挿抜位置に基づいて、画素の挿抜を前記画像に対して実行する挿抜工程と、

を有し、

前記決定された第2の挿抜位置は、前記決定された第1の挿抜位置からの前記変倍方向の距離が第1の所定の距離以上、該第1の所定の距離より大きい第2の所定の距離以内となる範囲内のいずれかの位置であることを特徴とする画像処理方法。

30

**【請求項 2 0】**

請求項1乃至18の何れか1の各手段としてコンピュータを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】****【0 0 0 1】**

本発明は、画像処理装置、画像処理方法、及びそのコンピュータプログラムに関するものである。

**【背景技術】****【0 0 0 2】**

電子写真方式の印字方式は、顔料を含むトナーを用紙に定着させる現像段階において熱と圧力を利用する。熱と圧力をかけられた用紙は、用紙の縮小ないし伸張のいずれかの現象を起こす。電子写真方式の印字装置において使用可能な用紙では、この現象は一度目の定着において顕著に生じ、二回目以降はその変位はほぼ無視できる。

40

**【0 0 0 3】**

用紙の伸縮にあわせて印字画像も伸縮するので、両面印刷の実行時に表面印字と裏面印字の大きさがわずかに異なったものとなる。印字精度が重用視される分野においてはこの差を補正する必要がある。

**【0 0 0 4】**

このような課題に対して、印字画像情報を調整して用紙の伸縮に対応する方法があ

50

る。特許文献 1 には、所定数画素につき 1 画素をランダムに選択して挿入処理・間引き処理（以後、挿入処理と間引き処理を合わせて挿抜処理と呼ぶ）を実行することで、印字画像情報の拡大、縮小を行う方法が開示されている。

#### 【0005】

挿抜処理を行う画素位置をランダムにすることで、ハーフトーン画像についても挿抜処理による拡大、縮小を高画質で実現することが出来る。

#### 【先行技術文献】

#### 【特許文献】

#### 【0006】

【特許文献 1】特開昭 61 - 206365

10

#### 【発明の概要】

#### 【発明が解決しようとする課題】

#### 【0007】

しかしながら、ハーフトーン画像について、挿抜処理を行う画素の位置（挿抜位置）を乱数でランダムに決定した場合には、ディザ処理によって作成されたディザパターンが壊れてしまい、印刷時のドットが不安定になってしまうという問題がある。

#### 【0008】

例えば、主走査位置を  $x$  、副走査位置を  $y$  とし、副走査方向についての間引き処理を実施したとする。例えば縮小率が 1 % とすると 100 ラインに一回の間引き処理とした場合、乱数の取り得る値は、

20

`Random(x) : 0 ~ 99`

である。なお、関数 `Random(x)` は、引数  $x$  に基づく乱数を生成する関数である。

#### 【0009】

今、間引き処理をする主走査位置  $x_0$  と  $x_1$  は隣り合う列であった場合に `Random(x_0) = 2` 、 `Random(x_1) = 96` となつたと仮定すると、間引きする画素の座標は、 $(x, y) = (x_0, y_2), (x_1, y_{96})$  となり、図 11 (a) に示すように、間引き位置が 94 ライン分離れる事になる。

#### 【0010】

図 11 (b) では、実際に指定の画素が間引きされて、画素がシフトされた状態である。 $x_0$  は  $y_2$  ライン目が間引きされるので、 $y_3$  ライン目以降の出力位置は 1 ラインずつ繰り上がる。一方、 $x_1$  は  $y_{96}$  ライン目が間引きされるので、 $y_{97}$  ライン目以降の出力位置は 1 ラインずつ繰り上がる。

30

#### 【0011】

図 11 (a) の状態での  $y_{97} \sim y_{99}$  ライン目が図 11 (b) の状態の  $y_{96} \sim y_{98}$  ライン目にシフトしただけなので、図 11 (b) の  $y_{96}$  ライン目以降のラインの主走査位置  $x_0$  と  $x_1$  の間のディザパターンは崩れない。また、 $y_{99}$  ライン目（図 11 (b) で破線で囲まれた領域）は無くなるので、1 % の縮小には成功している。一方、 $x_0$ 、 $x_1$  について、 $y_2$  ライン目～ $y_{95}$  ライン目までは、間引きによる影響で隣り合う  $x_0$ 、 $x_1$  の位置がずれるので、ディザパターンはずれた状態で形成される。また、図 11 (a) に示すように主走査位置  $x_1$ 、 $x_2$ 、 $x_3$  の間引く画素位置がそれぞれ異なるので、図 11 (b) に示す間引き処理後のパターンは、図 11 (a) に示す間引き処理前のディザパターンとは異なるパターンとなり、このパターンによって画像形成が実施されるために電子写真的画像形成は不安定になってしまう。

40

#### 【課題を解決するための手段】

#### 【0012】

上記課題を解決するため、本発明の画像処理装置は、画素の挿抜を実行することで画像の変倍を行う画像処理装置であって、変倍方向に直交する方向の第 1 の位置における第 1 の挿抜位置を決定する第 1 の決定手段と、前記直交する方向の前記第 1 の位置とは異なる第 2 の位置における第 2 の挿抜位置を、前記決定された第 1 の挿抜位置に基づいて決定する第 2 の決定手段と、前記決定された第 1 および第 2 の挿抜位置に基づいて、画素の挿抜

50

を前記画像に対して実行する挿抜手段と、を有し、前記決定された第2の挿抜位置は、前記決定された第1の挿抜位置からの前記変倍方向の距離が第1の所定の距離以上、該第1の所定の距離より大きい第2の所定の距離以内となる範囲内のいずれかの位置であることを特徴とする。

【発明の効果】

【0013】

本発明によれば、挿抜処理の対象となる画素の相対位置を制限することで、画像内の画素のパターンの崩れを抑制することができる。

【図面の簡単な説明】

【0014】

【図1】本発明による画像処理装置のブロック図である。

10

【図2】実施例1の画像処理装置の動作を示すフローチャートである。

【図3】S2005を詳細に説明した実施例1のフローチャートである。

【図4】S2008を詳細に説明した実施例1のフローチャートである。

【図5】S2005を詳細に説明した実施例2のフローチャートである。

【図6】S2005を詳細に説明した実施例3のフローチャートである。

【図7】実施例4のハーフトーン画像と画像情報を説明した図である。

【図8】S2005を詳細に説明した実施例4のフローチャートである。

【図9】S2005を詳細に説明した実施例5のフローチャートである。

【図10】本発明の画素の間引きによる変倍処理を表す図である。

20

【図11】従来の画素の間引きによる変倍処理を表す図である。

【発明を実施するための形態】

【0015】

以下、本発明を実施するための最良の形態について図面を用いて説明する。

【0016】

(実施例1)

本発明を実施する為の実施例を説明する。

【0017】

図1は、本発明における画像処理装置のブロック図である。画像処理装置は、CPU101、RAM103、ROM104を有している。CPU101は、ROM104に格納されている制御プログラムをRAM103にロードし、展開、実行することで、当該画像処理装置を動作させる。また、この制御プログラムの実行において、RAM103は、CPU101がこの制御プログラムを実行する際のワーク用メモリとしても利用され、ビットマップイメージ等のデータの一時保存に利用される。

30

【0018】

また、本画像処理装置は、操作部102、ネットワークI/F106を有している。操作部102は、ユーザからの変倍率設定を受け付ける。CPU101は、この操作部102を介してユーザによって設定された変倍率をRAM103に格納する。この変倍率設定は、プリンタユニット(印字機構)111で画像の両面印刷を行うような場合、裏面に印字される画像を表面に印字される画像の位置・大きさに合わせるような値に設定される。例えば変倍率が99%で指定された場合、裏面に印字される画像(ハーフトーン画像)の画素に間引き処理を施して裏面画像の縮小を図る。また例えば変倍率が101%で指定された場合、裏面に印字される画像(ハーフトーン画像)の画素に挿入処理を施して裏面画像の拡大を図る。変倍率が100%の場合、挿抜処理を実行せずに等倍の裏面画像を印字することになる。画像を拡大する処理及び縮小する処理をあわせて変倍処理と呼ぶ。また、この操作部102及びネットワークI/F106は、ユーザからの印刷実行命令を受ける。例えば紙媒体の原稿をコピーする場合、CPU101は、操作部102を介して印刷実行命令を受け、スキャナユニット(不図示)で読み取った原稿のビットマップ画像をRAM103へ格納する。また、例えばすでにRAM103に格納されているビットマップ画像をプリントする場合、ユーザはホストコンピュータ(不図示)からこのビットマップ

40

50

画像のプリントの印刷実行命令をネットワークI/F106を介してCPU101へ送信し、この画像をプリントする。以下の本実施例では、印刷対象であるビットマップ画像がRAM103にすでに格納されているものとする。

#### 【0019】

また、本画像装置は、画像処理コントローラ107、乱数生成器108、乱数バッファ109、出力バッファ110、プリンタユニット（印字機構）111を有している。画像処理コントローラ107は、RAM103に格納されているビットマップ画像に対してデイザ処理を実行してハーフトーン画像を生成する。画像処理コントローラ107は、この生成されたハーフトーン画像を、出力バッファ110へ格納する。また、画像処理コントローラ107は、このハーフトーン画像の大きさをRAM103に格納する。以降の説明において、ハーフトーン画像の大きさについて、ハーフトーン画像の主走査方向の長さを主走査入力サイズ、副走査方向の長さを副走査入力サイズとする。10

#### 【0020】

乱数生成器108は乱数列を生成し、この乱数列を乱数バッファ109へ格納する。この乱数生成器108の乱数列の生成手法については、ボックス・ミュラー法などの公知の手法を用いればよい。なお、この乱数生成器108の乱数列の生成手法は、本発明を限定するものではない。そして、CPU101は、出力バッファ110に格納されたハーフトーン画像を走査して、乱数バッファ109に格納されている乱数列に基づいた変倍処理を実行する。そしてこの変倍処理が施されたハーフトーン画像をプリンタユニット111へ転送する。変倍処理の処理フローは図2に示される処理フローである。プリンタユニット111は、この変倍処理後のハーフトーン画像を取得し、取得した画像に基づいて画像形成が実行する。このプリンタユニットは紙などの記録媒体の両面を印字可能である。20

#### 【0021】

以上説明した本画像処理装置が有する構成（CPU101、操作部102、RAM103、ROM104、ネットワークI/F106、画像処理コントローラ107、乱数生成器108、乱数バッファ109、出力バッファ110、プリンタユニット111）の間でデータの送受信を行う場合、バス105を介してデータの送受信が行われる。

#### 【0022】

なお、前述のハーフトーン画像に対する画素の挿入・間引きを実施する為には、挿入・間引きされる画素の座標を計算する事が必要である。本実施例では、この計算はCPU101によって実行されるが、この計算を実行する装置を限定するものでは無く、例えば専用の演算回路等の別個のハードウェア回路を設けてこの計算を実行する事も出来る。30

#### 【0023】

図2は、前述のCPU101が出力バッファ110に格納されたハーフトーン画像を走査しながら実行する挿入・間引き画素の座標演算のフローを示した図である。図2に示される処理フローは、ROM104に格納されている制御プログラムをCPU101がRAM103にロードし、展開、実行することで実現される。なお、図2の処理フローは副走査方向を変倍処理方向とした変倍処理を実行するものであるが、本発明は主走査方向の変倍処理にも適用できる。なお、図2の処理フローを実行する前に、変倍率が操作部102を介してユーザによって設定されており、また、CPU101はユーザによる印刷実行命令を受けて、変倍処理前のハーフトーン画像が出力バッファ110に格納されているものとする。40

#### 【0024】

また、図10は、図2の処理フローによる変倍処理の様子を示した図の一例である。図10(a)は、図2の処理フローを実行する前に出力バッファ110に格納されているハーフトーン画像である。また、図10(a)には挿抜処理（ここでは間引き処理）を施す画素位置が示されている。図10(b)は、図2の処理フローを実行して図10(a)のハーフトーン画像に対して変倍処理が施された後のハーフトーン画像である。この変倍処理後のハーフトーン画像がプリンタユニット111へ転送される。なお、本実施例では図10(a)に示すNはN=1である。50

**【0025】**

S2001において、CPU101は、初期化処理を実行する。初期化処理とは、変倍処理後のハーフトーン画像の座標位置を表すパラメータx、yのリセット( $x = y = 0$ )やRAM103の記憶領域の確保などが含まれる。また、CPU101は、乱数生成回数カウンタcを $c = -1$ で初期化する。なお、パラメータxは変倍処理後のハーフトーン画像面内の主走査方向の座標位置を表すもので、以降の説明にて主走査位置xと表される。パラメータyについても同様で、パラメータyは変倍処理後のハーフトーン画像面内の副走査方向の座標位置を表すものであり、以降の説明に副走査位置yと表現される。

**【0026】**

S2002において、CPU101は、操作部102を介して設定されてRAM103 10に格納された変倍率から、変倍処理単位を以下の式で求める。

$$\text{変倍処理単位(ライン)} = 1 \div \text{倍率}$$

**【0027】**

また、S2002において、CPU101は、RAM103から主走査入力サイズと副走査入力サイズを取得し、変倍処理によって生成される変倍処理後のハーフトーン画像の大きさ(主走査出力サイズ、副走査出力サイズ)を演算によって求める。S2002の後、処理はステップS2003へ移行する。

**【0028】**

変倍処理単位とは、例えば、1%(0.01倍)の変倍処理(ここでは拡大・縮小は限定しない)の場合、変倍処理単位は100( $= 1 \div 0.01$ )ラインとなる。つまり、1 200ライン毎に挿入・間引きを実行することになる。

**【0029】**

S2003において、CPU101は、副走査位置yが副走査出力サイズよりも小さいかを判断する。S2003の後、処理はステップS2004へ移行する。

**【0030】**

S2004において、CPU101は、副走査位置yとステップS2002で求めた変倍処理単位との剰余算が0であるかを判定する。ステップS2004において副走査位置yと変倍処理単位との剰余算が0である場合(S2004でYESである場合)、処理は、S2005の乱数テーブル生成処理に進む。ステップS2004において副走査位置yとステップS2002で求めた変倍処理単位との剰余算が0ではない場合(S2004でNOである場合)、処理は、ステップS2007へ進む。 30

**【0031】**

S2005における乱数テーブル生成処理の処理フローを、図3のフローチャートに示す。図3に示す処理フローは、乱数生成器108の処理と、ROM104に格納されている制御プログラムをCPU101がRAM103にロードし、展開、実行することで実現される処理とによって実行される。図3の処理フローを実行することで、どの位置の画素を挿抜処理の対象とするか、挿抜位置が決定される。

**【0032】**

ステップS3000において、CPU101は、乱数位置カウンタpの初期化( $p = 0$ )を実行し、また、乱数バッファ109のバッファサイズを主走査出力サイズで確保する。本実施例において、この乱数バッファ109上の位置は、乱数位置カウンタpによって一意に決定される。なお、この乱数位置カウンタpの値を1インクリメントすることは、主走査方向に沿って主走査位置xを1インクリメントすることに対応し、乱数位置カウンタpによって決定される挿抜位置は、乱数位置カウンタpによって決定される挿抜位置から見て、直前に決定された挿抜位置となっている。 40

**【0033】**

S3001において、CPU101は、乱数位置カウンタpがRAM103に格納されている主走査出力サイズより小さいかを判断する。乱数位置カウンタpが主走査出力サイズより小さい場合(S3001でYESの場合)、処理はステップS3002へ移行する。乱数位置カウンタpが主走査出力サイズ以上の場合(S3001でNOの場合)、乱数 50

テーブル生成処理は終了し、処理はステップ S 2 0 0 6 へ移行する。

**【 0 0 3 4 】**

S 3 0 0 2 において、乱数生成器 1 0 8 が乱数を生成し、C P U 1 0 1 は、乱数生成器 1 0 8 で生成した乱数を取得する。S 3 0 0 2 の後、処理はステップ S 3 0 0 3 へ移行する。

**【 0 0 3 5 】**

S 3 0 0 3 において、C P U 1 0 1 は、S 3 0 0 2 で取得した乱数が変倍処理単位より小さいかを判断する。取得した乱数が変倍処理単位より小さい場合 (S 3 0 0 3 において Y E S の場合)、処理は、ステップ S 3 0 0 4 へ移行する。取得した乱数が変倍処理単位以上の場合 (S 3 0 0 3 において N O の場合)、処理はステップ S 3 0 0 2 へ移行する。

10

**【 0 0 3 6 】**

S 3 0 0 4 において、C P U 1 0 1 は、乱数位置カウンタ p が 0 であるかを判断する。乱数位置カウンタ p が 0 である場合 (S 3 0 0 4 で Y E S の場合)、処理は、S 3 0 0 5 に移行する。乱数位置カウンタ p が 0 でない (開始位置でない) 場合 (S 3 0 0 4 で N O の場合)、処理は、S 3 0 0 5 に移行する。

**【 0 0 3 7 】**

S 3 0 0 5 において、C P U 1 0 1 は、乱数位置カウンタ p に対応する乱数バッファ 1 0 9 の位置に、取得した乱数を上書き格納する。この乱数バッファ 1 0 9 に格納された乱数が挿抜位置を決定するために用いられる。S 3 0 0 5 の後、処理は、S 3 0 0 7 へ移行し、S 3 0 0 7 において、C P U 1 0 1 は、乱数位置カウンタ p を 1 だけインクリメントして、処理は、S 3 0 0 1 に進む。

20

**【 0 0 3 8 】**

S 3 0 0 6 において、C P U 1 0 1 は、取得した乱数と、1つ前の乱数位置カウンタ p - 1 に対応する乱数バッファ 1 0 9 の位置に格納された乱数 (隣接乱数) との差の絶対値 (図 1 0 ( a ) に示す距離 L に相当) が所定の最小値から所定の最大値の間、即ち所定の範囲内であるかを判定する。この差の絶対値が所定の範囲内である場合 (S 3 0 0 6 で Y E S の場合)、処理は、ステップ S 3 0 0 5 へ移行する。また、この差の絶対値が所定の範囲内でない場合 (S 3 0 0 6 で N O の場合)、処理は、S 3 0 0 2 へ移行する。この所定の最小値及び最大値は、隣接する画素の挿抜位置間で取り得る相対距離を第 1 の距離 (最小値) と第 2 の距離 (最大値) の間に制限するものであり、例えばハーフトーン画像の線数と解像度とによって決定される。解像度 6 0 0 d p i 、線数 1 0 0 1 p i のハーフトーン画像よりも、6 0 0 d p i 、2 0 0 1 p i のハーフトーン画像の方が、ある位置の画素の挿入・間引きが、他の位置のドット形成への影響が大きくなってしまう。そのため、解像度に対する線数が多いハーフトーン画像の最大値に対しては、それが少ないハーフトーン画像で設定される最大値よりも小さく設定する。

30

**【 0 0 3 9 】**

図 2 の説明に戻る。S 2 0 0 6 において、C P U 1 0 1 は、乱数生成回数カウント値 c を + 1 だけインクリメントする。この乱数生成回数カウント値 c は、後述のステップ S 2 0 0 8 の座標演算処理で、変倍処理前のハーフトーン画像に対する画素の挿抜処理を行う時に用いる。S 2 0 0 6 の後、処理はステップ S 2 0 0 7 へ進む。

40

**【 0 0 4 0 】**

ステップ S 2 0 0 7 ~ S 2 0 1 0 は、主走査出力サイズ分の画素の値を計算するステップである。

**【 0 0 4 1 】**

S 2 0 0 7 において、C P U 1 0 1 は、主走査位置 x が主走査出力サイズより小さいかを判断する。主走査位置 x が主走査出力サイズより小さい場合 (S 2 0 0 7 で Y E S の場合)、処理は、ステップ S 2 0 0 8 へ移行する。主走査位置 x が主走査出力サイズ以上である場合 (S 2 0 0 7 で N O の場合)、処理は、ステップ S 2 0 1 1 へ移行する。処理が S 2 0 1 1 へ移行した場合、C P U 1 0 1 は、主走査位置 x に 0 を代入し、副走査位置 y を 1 だけインクリメントして、処理を S 2 0 0 3 へと進める。

50

**【0042】**

S2008において、CPU101は、乱数バッファ109に蓄積された乱数を利用して、出力バッファ110に格納された変倍処理前のハーフトーン画像に対してどの画素をプリントユニット111へ転送するかを決定する処理（座標演算処理）を実行する。図4のフローチャートは、この座標演算処理の処理フローを説明した図である。

**【0043】**

図4のフローチャートの説明を行う。S4001において、CPU101は、変倍処理が拡大処理であるかを判断する。変倍処理が拡大処理である場合（S4001でYESの場合）、処理は、S4002に移行する。変倍処理が拡大処理でない場合（S4001でNOの場合）、処理は、S4003へ移行する。

10

**【0044】**

ステップS4002において、CPU101は、主走査位置xに対応する乱数として、S3005にて乱数位置カウンタ $p = x$ のときに格納した乱数を乱数バッファ109から読み込み、この読み込んだ乱数が、副走査位置yと変倍処理単位との剰余算の結果より小さいかを判断する。つまり、走査しているハーフトーン画像の主走査位置xにおいて、変倍処理単位内で画素の挿入が終わっているかどうかの判断をしている。この読み込んだ乱数がこの剰余算の結果より小さい場合（S4002でYESの場合）、処理は、S4004へ移行する。また、この読み込んだ乱数がこの剰余算の結果以上の場合（S4002でNOの場合）、処理は、S4005へ移行する。

**【0045】**

S4004において、CPU101は、副走査位置yから乱数生成回数カウント値cと1画素挿入分を考慮した値 $y - c - 1$ の画素を指定する演算を行う。つまり、このS4004の処理において、CPU101は、変倍処理前のハーフトーン画像の面内座標で（x、y - c - 1）の画素を、変倍処理後のハーフトーン画像の座標（x、y）の画素として指定する演算を行う。S4004の処理の後、処理は、図2のステップS2009へ移行する。

20

**【0046】**

S4005において、CPU101は、副走査位置yから、乱数生成回数カウント値cを減算した値 $y - c$ の画素を指定する演算を行う。つまり、このS4005の処理において、CPU101は、ハーフトーン画像の面内座標（x、y - c）の画素を、変倍処理後のハーフトーン画像の座標（x、y）の画素として指定する演算を行う。S4005の処理の後、処理は、図2のステップS2009へ移行する。

30

**【0047】**

一方、ステップS4003において、CPU101は、主走査位置xに対応する乱数として、S3005にて乱数位置カウンタ $p = x$ のときに格納した乱数を乱数バッファ109から読み込み、この読み込んだ乱数が、副走査位置yと変倍処理単位との剰余算の結果より小さいかを判断する。つまり、走査しているハーフトーン画像の主走査位置xにおいて、変倍処理単位内で画素の挿入が終わっているかどうかの判断をしている。この読み込んだ乱数がこの剰余算の結果より小さい場合（S4003でYESの場合）、処理は、S4006へ移行する。また、この読み込んだ乱数がこの剰余算の結果以上の場合（S4003でNOの場合）、処理は、S4007へ移行する。

40

**【0048】**

S4006において、CPU101は、副走査位置yから乱数生成回数カウント値cと1画素間引き分を考慮した値 $y + c + 1$ の画素を指定する演算を行う。つまり、このS4006の処理において、CPU101は、ハーフトーン画像の面内座標で（x、y + c + 1）の画素を、変倍処理後のハーフトーン画像の座標（x、y）の画素として指定する演算を行う。S4006の処理の後、処理は、図2のステップS2009へ移行する。

**【0049】**

S4007において、CPU101は、副走査位置yから、乱数生成回数カウント値cを加算した値 $y + c$ の画素を指定する演算を行う。つまり、このS4005の処理におい

50

て、CPU101は、ハーフトーン画像の面内座標( $x, y + c$ )の画素を、変倍処理後のハーフトーン画像の座標( $x, y$ )の画素として指定する演算を行う。S4007の処理の後、処理は、図2のステップS2009へ移行する。

#### 【0050】

S2009において、CPU101は、ステップS2008での演算により求めた座標値を出力バッファ110に指示し、出力バッファ110に格納されたハーフトーン画像の当該座標値の画素をプリントユニット111転送する。

#### 【0051】

S2010において、CPU101は、主走査位置 $x$ を1だけインクリメントする。S2010の後、処理はS2007へ移行する。

10

#### 【0052】

以上の処理フローを実行すると図10(a)のように画素の挿抜位置を、1つ前の挿抜位置との変倍処理方向での距離Lを制御して決定することができ、図10(b)に示すように挿抜処理後のハーフトーン画像を出力できる。図10(b)に示すハーフトーン画像は、ディザパターンの崩れでドットが不安定になる箇所の変倍処理方向の長さを抑えつつ、変倍処理方向に適度に分散させたものとなる。

#### 【0053】

以上、本実施例によれば、乱数を利用して安定したハーフトーン画像を提供する事が可能となる。

#### 【0054】

20

##### (実施例2)

実施例1の乱数テーブル生成処理において、乱数生成器108で生成した乱数とその生成した乱数の隣接乱数との差が所定の範囲内に収まるように、CPU101は主走査位置 $x$ が1増えるごとに乱数を取得していた。本実施例では、乱数テーブル生成処理において、CPU101は主走査位置 $x + 1$ に対応する乱数テーブルの値に、直前の主走査位置 $x$ に対応する乱数テーブルの値をコピーした同じ値を使う実施例を説明する。乱数テーブルの値と同じにすることで、画素の挿抜位置は、主走査方向に連続して隣接することになる。本実施例では、この同じ値をどの程度連続して使うかを決定するために、主走査処理単位N(Nは自然数)が設定される。つまり、本実施例において、この設定された主走査処理単位のN回(回数N)、連続して画素の挿抜位置が隣接するように、挿抜位置は決定される。

30

#### 【0055】

なお、主走査処理単位とは、変倍処理方向(本実施例において副走査方向)に対して垂直方向に、挿抜位置を隣接して連続させる量であり、乱数テーブル生成処理で、乱数バッファ109に格納する値を何回連続で同じ値をコピーするかを決定する値である。例えば、主走査処理単位が5である場合、乱数バッファ109に格納される値は5つごとに変わることになる。一般的な乱数生成器108はシフトレジスタで構成され、生成する乱数は疑似乱数である。よって、シフトレジスタの大きさによっては、生成される乱数がパターンとして周期性をもってしまう。主走査処理単位は、生成する乱数の量を調整して乱数パターンが周期性を持たないようにするためのものである。

40

#### 【0056】

本実施例において、主走査処理単位は挿入・間引きが人の目に見えにくいように、ハーフトーン画像の線数に応じて、所定の値として予めROM104に格納されている。なお、主走査処理単位としてCPU101は操作部102を介してユーザが設定した値を取得しても良い。

#### 【0057】

以下、本実施例における乱数テーブル生成処理ステップ(S2005)について説明する。なお、本実施例において乱数テーブル生成処理以外の他の構成は、実施例1の構成と同様のものとする。

#### 【0058】

50

図5は、本実施例における乱数テーブル生成処理(S2005)の処理フローを示したフローチャートである。図5に示す処理フローは、乱数生成器108の処理と、ROM104に格納されている制御プログラムをCPU101がRAM103にロードし、展開、実行することで実現される処理とによって実行される。

#### 【0059】

ステップS5000において、CPU101は、乱数位置カウンタ $p$ の初期化( $p = 0$ )を実行し、また、乱数バッファ109のバッファサイズを主走査出力サイズで確保する。本実施例において、この乱数バッファ109上の位置は、乱数位置カウンタ $p$ によって一意に決定される。また、CPU101は、ROM104から主走査処理単位Nを読み出してこれを参照する。ステップS5000の次にCPU101はステップS5001に処理を進める。  
10

#### 【0060】

S5001において、CPU101は、乱数位置カウンタ $p$ がRAM103に格納されている主走査出力サイズより小さいかを判断する。乱数位置カウンタ $p$ が主走査出力サイズより小さい場合(S5001でYESの場合)、処理はステップS5002へ移行する。乱数位置カウンタ $p$ が主走査出力サイズ以上の場合(S5001でNOの場合)、乱数テーブル生成処理は終了し、S2006へ処理は移行する。

#### 【0061】

S5002において、CPU101は、乱数位置カウンタ $p$ と主走査処理単位Nとの剰余算が0であるかを判断する。この乱数位置カウンタ $p$ と主走査処理単位Nとの剰余算が0である場合(S5002でYESの場合)、処理は、ステップS5003へ移行する。また、この乱数位置カウンタ $p$ と主走査処理単位Nとの剰余算が0でない場合(S5002でNOの場合)、処理は、ステップS5007へ移行する。  
20

#### 【0062】

S5007において、CPU101は、乱数位置カウンタ $p - 1$ に対応する位置の乱数バッファ109の格納された乱数を取得する。そして処理はS5006へ移行する。

#### 【0063】

S5003において、乱数生成器108が乱数を生成し、CPU101は、乱数生成器108で生成した乱数を取得する。そして処理はステップS5004へ移行する。

#### 【0064】

S5004において、CPU101は、S5003で取得した乱数が変倍処理単位より小さいかを判断する。取得した乱数が変倍処理単位より小さい場合(S5004においてYESの場合)、処理は、ステップS5005へ移行する。取得した乱数が変倍処理単位以上の場合(S5004においてNOの場合)、処理はステップS5003へ移行する。  
30

#### 【0065】

S5005において、CPU101は、乱数位置カウンタ $p$ が0であるかを判断する。乱数位置カウンタ $p$ が0である場合(S5005でYESの場合)、処理は、S5006に移行する。乱数位置カウンタ $p$ が0でない場合(S5005でNOの場合)、処理は、S5008に移行する。

#### 【0066】

S5008において、CPU101は、取得した乱数と、1つ前の乱数位置カウンタ $p - 1$ に対応する乱数バッファ109の位置に格納された乱数(隣接乱数)との差の絶対値が所定の最小値から所定の最大値の間、即ち所定の範囲内であるかを判定する。この差の絶対値が所定の範囲内である場合(S5008でYESの場合)、処理は、ステップS5006へ移行する。また、この差の絶対値が所定の範囲内でない場合(S5008でNOの場合)、処理は、S5003へ移行する。この所定の最小値及び最大値については実施例1と同様なものである。  
40

#### 【0067】

S5006において、CPU101は、乱数位置カウンタ $p$ に対応する乱数バッファ109の位置に、取得した乱数を上書き格納する。そして処理は、S5009へ移行し、S  
50

5009において、CPU101は、乱数位置カウンタ $p$ を1だけインクリメントして、処理は、S5001に進む。

#### 【0068】

以上の処理フローを実行すると、図10(a)のように挿抜位置を変倍処理方向に垂直な方向にN回連続して隣接する。

#### 【0069】

以上、本実施例によれば、乱数生成器108で生成する乱数量を抑制しながら、挿入・間引き座標を制御する事が可能となり、安定したハーフトーン画像を提供する事が可能となる。

#### 【0070】

10

##### (実施例3)

実施例2では主走査処理単位Nを所定の定数としていたが、これにはハーフトーン画像の線数に応じて決定する必要がある。本実施例では、ハーフトーン画像の線数が分からぬ場合に対応するため、主走査処理単位N(回数N)に所定の定数を用いずに、挿抜位置を隣接させるか否かの決定に乱数を利用する。なお、本実施例において、ROM104には所定の規定値(例えば「7」)が格納されている。

#### 【0071】

以下、本実施例における乱数テーブル生成処理ステップ(S2005)について説明する。なお、本実施例において乱数テーブル生成処理以外の他の構成は、実施例1の構成と同様のものとする。

20

#### 【0072】

図6は、本実施例における乱数テーブル生成処理(S2005)の処理フローを示したフローチャートである。図6に示す処理フローは、乱数生成器108の処理と、ROM104に格納されている制御プログラムをCPU101がRAM103にロードし、展開、実行することで実現される処理とによって実行される。

30

#### 【0073】

ステップS6000において、CPU101は、乱数位置カウンタ $p$ の初期化( $p=0$ )を実行し、また、乱数バッファ109のバッファサイズを主走査出力サイズで確保する。本実施例においても、この乱数バッファ109上の位置は、乱数位置カウンタ $p$ によって一意に決定される。また、CPU101は、ROM104から所定の規定値を読み出し、RAM103へ格納する。このRAM103に格納された所定の規定値はCPU101によって適宜参照される。

#### 【0074】

S6001において、CPU101は、乱数位置カウンタ $p$ がRAM103に格納されている主走査出力サイズより小さいかを判断する。乱数位置カウンタ $p$ が主走査出力サイズより小さい場合(S6001でYESの場合)、処理はステップS6002へ移行する。乱数位置カウンタ $p$ が主走査出力サイズ以上の場合(S6001でNOの場合)、乱数テーブル生成処理は終了し、CPU101は処理をステップS2006へ進める。

30

#### 【0075】

S6002において、CPU101は、乱数を生成する。そして処理は、ステップS6003へ移行する。

40

#### 【0076】

S6003において、CPU101は、S6002で生成した乱数と所定の規定値との剰余算が0であるかを判断する。この剰余算の結果が0である場合(S6003でYESの場合)、処理は、ステップS6004へ移行する。また、この剰余算の結果が0出ない場合(S6003でNOの場合)、処理は、ステップS6008へ移行する。

#### 【0077】

このステップS6003の判断が挿抜位置をどの程度連続して隣接させるかを決定することになる。つまり主走査処理単位Nを乱数によって制御していることになる。なお、このステップS6003では、剰余算の結果が0でなくとも、例えば剰余算の結果が所定の

50

規定値の 1 / 2 以下の場合にステップ S 6 0 0 4 へ処理を進め、1 / 2 より大きい場合にステップ S 6 0 0 8 へ処理を進めるようにしても良く、本実施例の C P U 1 0 1 が生成した乱数の利用の仕方に制限されない。

#### 【 0 0 7 8 】

S 6 0 0 8 において、C P U 1 0 1 は、乱数位置カウンタ  $p - 1$  に対応する位置の乱数バッファ 1 0 9 の格納された乱数を取得する。そして処理はステップ S 6 0 0 7 へ移行する。

#### 【 0 0 7 9 】

S 6 0 0 4 において、乱数生成器 1 0 8 が乱数を生成し、C P U 1 0 1 は、乱数生成器 1 0 8 で生成した乱数を取得する。

10

#### 【 0 0 8 0 】

S 6 0 0 5 において、C P U 1 0 1 は、S 6 0 0 4 で取得した乱数が変倍処理単位より小さいかを判断する。取得した乱数が変倍処理単位より小さい場合 (S 6 0 0 5 において YES の場合)、処理は、ステップ S 6 0 0 6 へ移行する。取得した乱数が変倍処理単位以上の場合 (S 6 0 0 5 において NO の場合)、処理はステップ S 6 0 0 4 へ移行する。

#### 【 0 0 8 1 】

S 6 0 0 6 において、C P U 1 0 1 は、乱数位置カウンタ  $p$  が 0 であるかを判断する。乱数位置カウンタ  $p$  が 0 である場合 (S 6 0 0 6 で YES の場合)、処理は、S 6 0 0 7 に移行する。乱数位置カウンタ  $p$  が 0 でない場合 (S 6 0 0 6 で NO の場合)、処理は、S 6 0 0 9 に移行する。

20

#### 【 0 0 8 2 】

S 6 0 0 9 において、C P U 1 0 1 は、取得した乱数と、1 つ前の乱数位置カウンタ  $p - 1$  に対応する乱数バッファ 1 0 9 の位置に格納された乱数 (隣接乱数) との差の絶対値が所定の最小値から所定の最大値の間、即ち所定の範囲内であるかを判定する。この差の絶対値が所定の範囲内である場合 (S 6 0 0 9 で YES の場合)、処理は、ステップ S 6 0 0 7 へ移行する。また、この差の絶対値が所定の範囲内でない場合 (S 6 0 0 9 で NO の場合)、処理は、S 6 0 0 4 へ移行する。この所定の最小値及び最大値については実施例 1 と同様なものである。

#### 【 0 0 8 3 】

S 6 0 0 7 において、C P U 1 0 1 は、乱数位置カウンタ  $p$  に対応する乱数バッファ 1 0 9 の位置に、取得した乱数を上書き格納する。そして処理は、S 6 0 1 0 へ移行して、C P U 1 0 1 は、乱数位置カウンタ  $p$  を 1 だけインクリメントする。そして、処理は、S 6 0 0 1 に進む。

30

#### 【 0 0 8 4 】

以上、本実施例を用いれば、ハーフトーン画像に利用されている線数が不明であった場合に、乱数生成器 1 0 8 の生成する乱数の量を抑制しながら、挿入・間引き座標を制御する事が可能となる。

#### 【 0 0 8 5 】

##### ( 実施例 4 )

実施例 2 では、所定の主走査処理単位 N にもとづいて乱数生成器 1 0 8 の生成する乱数生成量を抑制した。しかしながら、一般に画像処理装置内の画像処理コントローラ 1 0 7 は、連続諧調の画像からハーフトーン画像を生成する時に、ハーフトーン処理を連続諧調の画像の面内情報に応じて処理を切りかえる。面内情報とは、連続階調画像のある処理対象領域が写真であるといった情報である。図 7 ( a ) は、この時のハーフトーン画像を示した図であり、図 7 ( b ) はこのハーフトーン画像の面内情報を示した図である。この面内情報に応じて、画像処理コントローラ 1 0 7 は、処理対象領域のハーフトーン処理の線数や角度を切りかえる。このため、画素を挿入・間引きすべき位置の範囲、主走査処理単位は処理対象領域の面内情報に応じて、適した値にすることが好ましい。

40

#### 【 0 0 8 6 】

本実施例では、画像の面内情報に応じて変倍処理に関わる各係数が変更される。

50

**【0087】**

以下、本実施例における乱数テーブル生成処理ステップ(S2005)について説明する。なお、本実施例において乱数テーブル生成処理以外の他の構成は、実施例1の構成と同様のものとする。なお、本実施例において、表1に示されるような面内情報と変倍処理に関わる各係数との関係を示すテーブルが予めROM104に格納されている。

**【0088】****【表1】**

表1

イメージ種類	最小値	最大値	主走査処理単位
写真	a	b	c
グラフィック	d	e	f
文字	g	h	i
背景	j	k	l

10

**【0089】**

この表1について説明する。イメージ種類は「写真」「グラフィック」「文字」「背景」の4種類あり、イメージ種類ごとに最小値、最大値、主走査処理単位が設定されている。例えばイメージ種類「写真」には、最小値a、最大値b、主走査処理単位cが設定されている。他のイメージ種類についても同様である。これら最小値、最大値、主走査処理単位は、後述するステップS8009、S8003で用いられる値であり、実施例2におけるステップS5008の所定の最小値、所定の最大値、S5002の主走査処理単位と同様のものである。

20

**【0090】**

図8は、本実施例における乱数テーブル生成処理(S2005)の処理フローを示したフローチャートである。図8に示す処理フローは、乱数生成器108の処理と、ROM104に格納されている制御プログラムをCPU101がRAM103にロードし、展開、実行することで実現される処理とによって実行される。

20

**【0091】**

ステップS8000において、CPU101は、乱数位置カウンタpの初期化( $p = 0$ )を実行し、また、乱数バッファ109のバッファサイズを主走査出力サイズで確保する。本実施例において、この乱数バッファ109上の位置は、乱数位置カウンタpによって一意に決定される。そしてCPU101は、処理をステップS8001へ進める。

30

**【0092】**

S8001において、CPU101は、乱数位置カウンタpがRAM103に格納されている主走査出力サイズより小さいかを判断する。乱数位置カウンタpが主走査出力サイズより小さい場合(S8001でYESの場合)、処理はステップS8002へ移行する。乱数位置カウンタpが主走査出力サイズ以上の場合(S8001でNOの場合)、乱数テーブル生成処理は終了し、処理はステップS2006へ進む。

**【0093】**

S8002において、CPU101は、乱数位置カウンタpに対応する主走査位置x、副走査位置yに対応するハーフトーン画像の座標(p, y)の面内情報(イメージ種類)を取得して、ROM104に格納されている表1から所定の最小値、最大値、および主走査処理単位といった各係数を取得する。そして処理はステップS8003へ移行する。

40

**【0094】**

S8003において、CPU101は、乱数位置カウンタpとS8002で取得した主走査処理単位との剰余算が0であるかを判断する。この乱数位置カウンタpと主走査処理単位との剰余算が0である場合(S8003でYESの場合)、処理は、ステップS8004へ移行する。また、この乱数位置カウンタpと主走査処理単位との剰余算が0でない場合(S8003でNOの場合)、処理は、ステップS8008へ移行する。

**【0095】**

50

S 8 0 0 8において、C P U 1 0 1は、乱数位置カウンタ p - 1に対応する位置の乱数バッファ 1 0 9の格納された乱数を取得する。そして処理は S 8 0 0 7へ移行する。

#### 【0 0 9 6】

S 8 0 0 4において、乱数生成器 1 0 8が乱数を生成し、C P U 1 0 1は、乱数生成器 1 0 8で生成した乱数を取得する。そして処理はステップ S 8 0 0 5へ移行する。

#### 【0 0 9 7】

S 8 0 0 5において、C P U 1 0 1は、S 8 0 0 4で取得した乱数が変倍処理単位より小さいかを判断する。取得した乱数が変倍処理単位より小さい場合 (S 8 0 0 5において YES の場合)、処理は、ステップ S 8 0 0 6へ移行する。取得した乱数が変倍処理単位以上の場合 (S 8 0 0 5において NO の場合)、処理はステップ S 8 0 0 4へ移行する。

10

#### 【0 0 9 8】

S 8 0 0 6において、C P U 1 0 1は、乱数位置カウンタ p が 0であるかを判断する。乱数位置カウンタ p が 0である場合 (S 8 0 0 6で YES の場合)、処理は、S 8 0 0 6に移行する。乱数位置カウンタ p が 0でない場合 (S 8 0 0 6で NO の場合)、処理は、S 8 0 0 9に移行する。

#### 【0 0 9 9】

S 8 0 0 9において、C P U 1 0 1は、取得した乱数と、1つ前の乱数位置カウンタ p - 1に対応する乱数バッファ 1 0 9の位置に格納された乱数 (隣接乱数)との差の絶対値が S 8 0 0 2で取得した所定の最小値から所定の最大値の間、即ち所定の範囲内であるかを判定する。この差の絶対値が所定の範囲内である場合 (S 8 0 0 9で YES の場合)、処理は、ステップ S 8 0 0 7へ移行する。また、この差の絶対値が所定の範囲内でない場合 (S 8 0 0 9で NO の場合)、処理は、S 8 0 0 4へ移行する。この所定の最小値及び最大値について実施例 1と同様なものである。

20

#### 【0 1 0 0】

S 8 0 0 7において、C P U 1 0 1は、乱数位置カウンタ p に対応する乱数バッファ 1 0 9の位置に、取得した乱数を上書き格納する。そして処理は、S 8 0 1 0へ移行し、S 8 0 1 0において、C P U 1 0 1は、乱数位置カウンタ p を 1だけインクリメントして、処理は、S 8 0 0 1に進む。

#### 【0 1 0 1】

以上、本実施例によれば、面内情報すなわちハーフトーン処理方法に応じた挿入・補間処理方法を利用する事ができる。結果、安定したハーフトーン画像を提供する事が可能となる。

30

#### 【0 1 0 2】

なお、イメージ種類と各係数の関係を表す表は表 1だけに限らず、イメージ種類の数に応じて表の大きさを変えても良い。

#### 【0 1 0 3】

##### (実施例 5)

実施例 2では、所定の主走査処理単位にもとづいて乱数生成器 1 0 8の生成する乱数生成量を抑制した。しかしながら、電子写真では、プリンタユニット 1 1 1が利用する記録メディア (記録媒体)の種類に応じて、記録時の電圧や定着温度などの環境を変更する。この記録時の環境 (電圧、定着温度など)がハーフトーン画像のドット形成に影響を与え、本発明による挿入・間引き方法による変倍処理に影響を与える。

40

#### 【0 1 0 4】

そこで、本実施例ではこれに対応する為、記録メディアに応じて変倍処理に関わる各係数が変更される画像処理装置について説明する。

#### 【0 1 0 5】

以下、本実施例における乱数テーブル生成処理ステップ (S 2 0 0 5)について説明する。なお、本実施例において乱数テーブル生成処理以外の他の構成は、実施例 1の構成と同様のものとする。なお、本実施例において、表 2に示されるような記録メディアと変倍処理に関わる各係数との関係を示すテーブルが予め R O M 1 0 4に格納されている。また

50

、記録メディアの種類の情報は操作部 102 を介してユーザが設定し、この設定情報を C P U 1 0 1 は R A M 1 0 3 に格納している。

### 【0106】

#### 【表2】

表2

メディア種類	最小値	最大値	主走査処理単位
紙種A	a	b	c
紙種B	d	e	f
紙種C	g	h	i
紙種D	j	k	l

10

### 【0107】

この表2について説明する。メディア種類は「紙種A」「紙種B」「紙種C」「紙種D」の4種類あり、メディア種類ごとに最小値、最大値、主走査処理単位が設定されている。例えばメディア種類「紙種A」には、最小値a、最大値b、主走査処理単位cが設定されている。他のメディア種類についても同様である。これら最小値、最大値、主走査処理単位は、後述するステップS9009、S9003で用いられる値であり、実施例2におけるステップS5008の所定の最小値、所定の最大値、S5002の主走査処理単位と同様のものである。

### 【0108】

20

図9は、本実施例における乱数テーブル生成処理(S2005)の処理フローを示したフローチャートである。図9に示す処理フローは、乱数生成器108の処理と、R O M 1 0 4 に格納されている制御プログラムをC P U 1 0 1 がR A M 1 0 3 にロードし、展開、実行することで実現される処理とによって実行される。

### 【0109】

ステップS9000において、C P U 1 0 1 は、乱数位置カウンタ p の初期化(p = 0)を実行し、また、乱数バッファ109のバッファサイズを主走査出力サイズで確保する。本実施例において、この乱数バッファ109上の位置は、乱数位置カウンタ p によって一意に決定される。

### 【0110】

30

S9001において、C P U 1 0 1 は、乱数位置カウンタ p がR A M 1 0 3 に格納されている主走査出力サイズより小さいかを判断する。乱数位置カウンタ p が主走査出力サイズより小さい場合(S9001でY E S の場合)、処理はステップS9002へ移行する。乱数位置カウンタ p が主走査出力サイズ以上の場合(S9001でN O の場合)、乱数テーブル生成処理は終了し、処理はステップS2006へ進む。

### 【0111】

S9002において、C P U 1 0 1 は、メディア種類(紙種)をR A M 1 0 3 から読み出して、R O M 1 0 4 に格納されている表2から読み出したメディア種類に対応する所定の最小値、最大値、および主走査処理単位といった各係数を取得する。

### 【0112】

40

S9003において、C P U 1 0 1 は、乱数位置カウンタ p とS9002で取得した主走査処理単位との剰余算が0であるかを判断する。この乱数位置カウンタ p と主走査処理単位との剰余算が0である場合(S9003でY E S の場合)、処理は、ステップS9004へ移行する。また、この乱数位置カウンタ p と主走査処理単位との剰余算が0でない場合(S9003でN O の場合)、処理は、ステップS9008へ移行する。

### 【0113】

S9008において、C P U 1 0 1 は、乱数位置カウンタ p - 1 に対応する位置の乱数バッファ109の格納された乱数を取得する。そして処理はS9007へ移行する。

### 【0114】

S9004において、乱数生成器108が乱数を生成し、C P U 1 0 1 は、乱数生成器

50

108で生成した乱数を取得する。そして処理はステップS9005へ移行する。

【0115】

S9005において、CPU101は、S9004で取得した乱数が変倍処理単位より小さいかを判断する。取得した乱数が変倍処理単位より小さい場合(S9005においてYESの場合)、処理は、ステップS9006へ移行する。取得した乱数が変倍処理単位以上の場合(S9005においてNOの場合)、処理はステップS9004へ移行する。

【0116】

S9006において、CPU101は、乱数位置カウンタ $p$ が0であるかを判断する。乱数位置カウンタ $p$ が0である場合(S9006でYESの場合)、処理は、S9006に移行する。乱数位置カウンタ $p$ が0でない場合(S9006でNOの場合)、処理は、S9009に移行する。

10

【0117】

S9009において、CPU101は、取得した乱数と、1つ前の乱数位置カウンタ $p$ -1に対応する乱数バッファ109の位置に格納された乱数(隣接乱数)との差がS8002で取得した所定の最小値から所定の最大値の間、即ち所定の範囲内であるかを判定する。この差が所定の範囲内である場合(S9009でYESの場合)、処理は、ステップS9007へ移行する。また、この差が所定の範囲内でない場合(S9009でNOの場合)、処理は、S9004へ移行する。この所定の最小値及び最大値については実施例1と同様なものである。

【0118】

20

S9007において、CPU101は、乱数位置カウンタ $p$ に対応する乱数バッファ109の位置に、取得した乱数を上書き格納する。そして処理は、S9010へ移行し、S9010において、CPU101は、乱数位置カウンタ $p$ を1だけインクリメントして、処理は、S9001に進む。

【0119】

以上、本実施例はメディア種類に応じて各係数を取得するが、実施例4の技術思想と組み合わせて、面内情報とメディア種類との両方を考慮したテーブルを予めROM104に格納しておき、CPU101はこのテーブルを適宜参照するようにしても良い。

【0120】

以上、本実施例によれば、記録メディアの種類に応じた挿入・補間処理方法を利用する事ができる。結果、安定したハーフトーン画像を提供する事が可能となる。

30

【0121】

なお、メディア種類と各係数の関係を表す表は表2だけに限らず、メディア種類の数に応じて表の大きさを変えても良い。

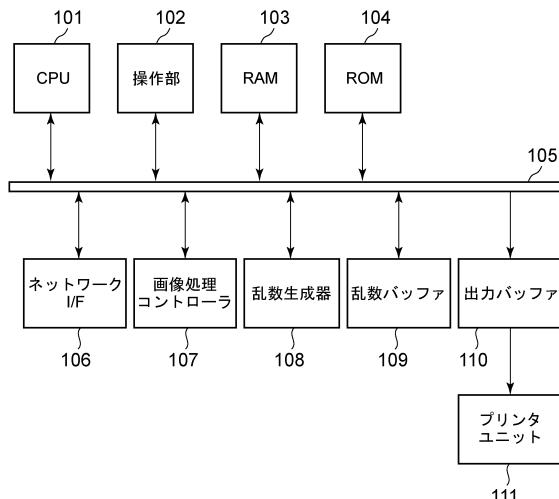
【0122】

(その他の実施例)

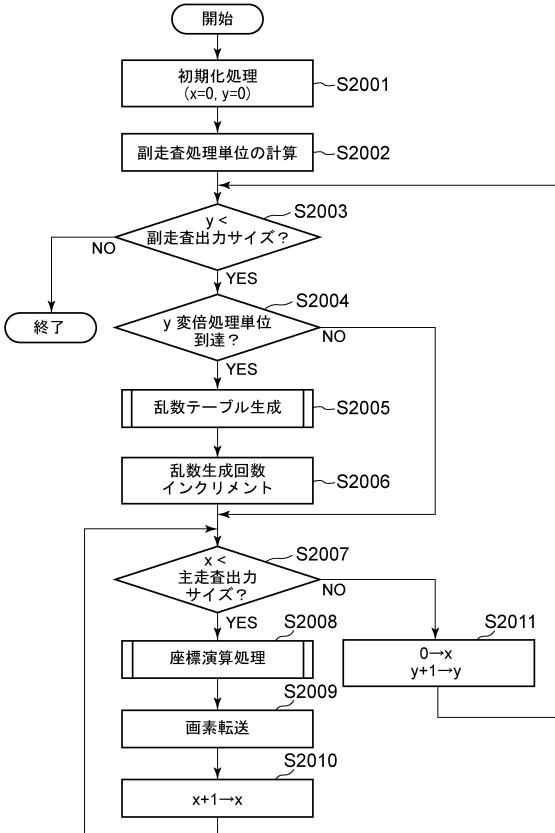
また、本発明は、以下の処理を実行することによっても実現される。その処理は、上述した実施例の機能を実現させるソフトウェア(プログラム)を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステム或いは装置のコンピュータ(またはCPUやMPU等)がプログラムを読み出して実行する処理である。

40

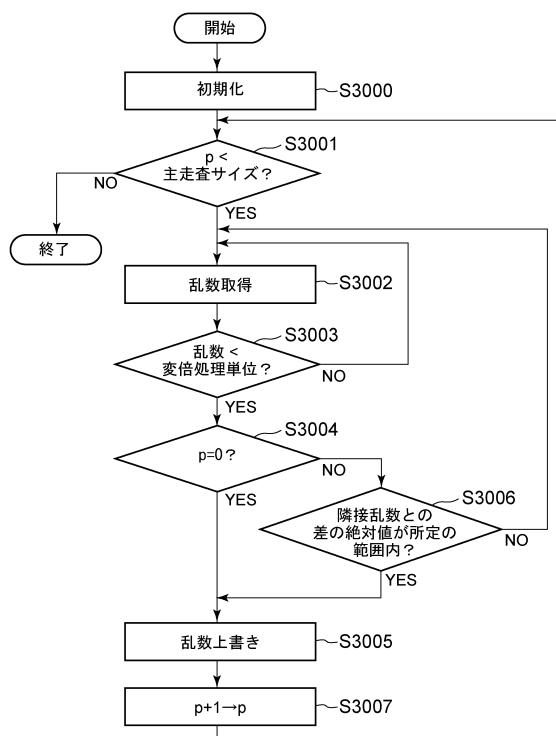
【図1】



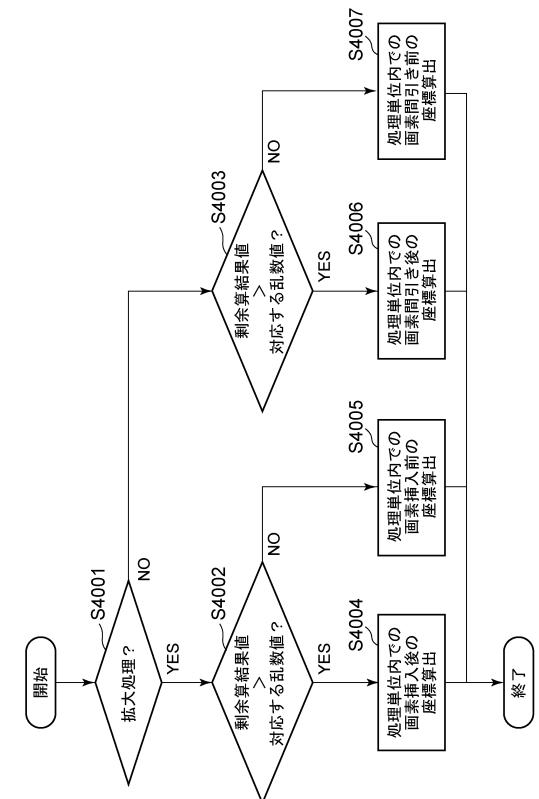
【図2】



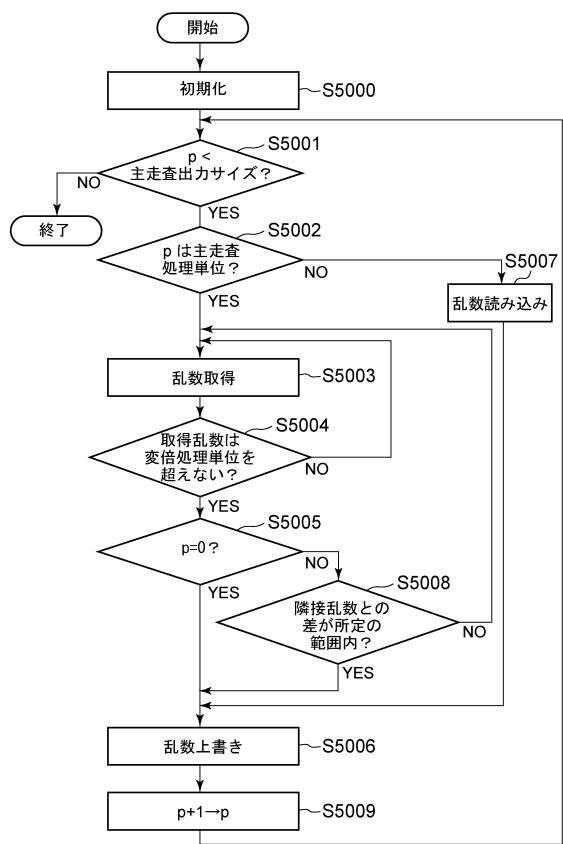
【図3】



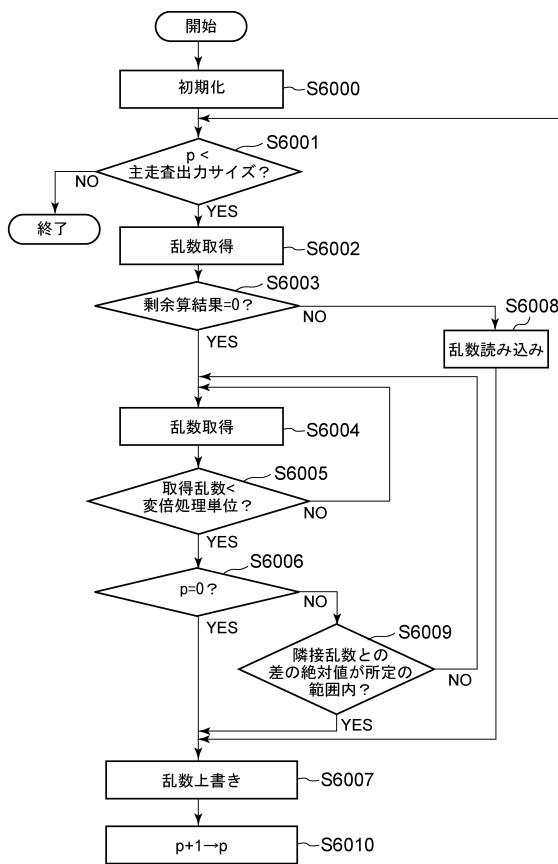
【図4】



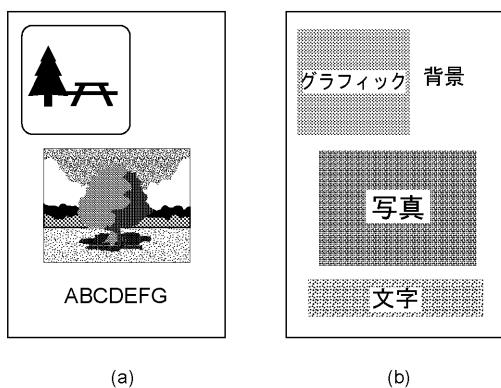
【図5】



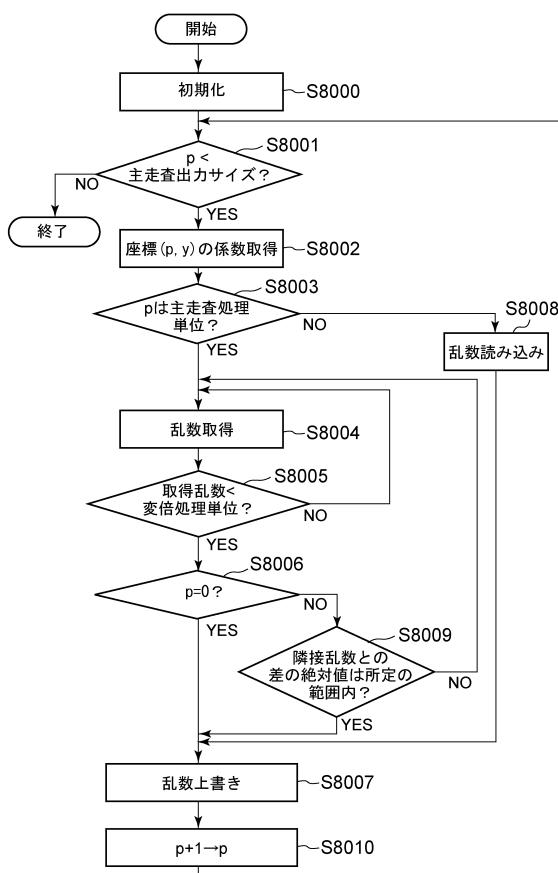
【図6】



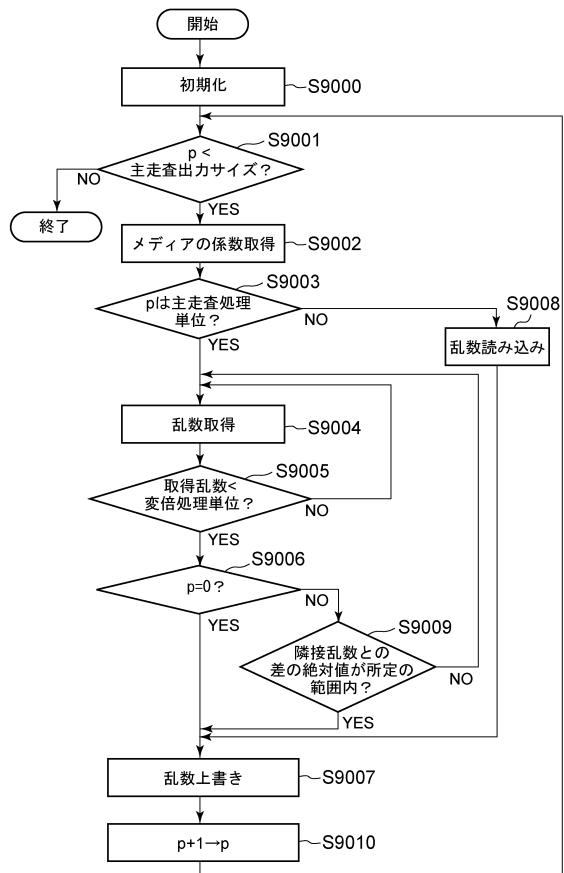
【図7】



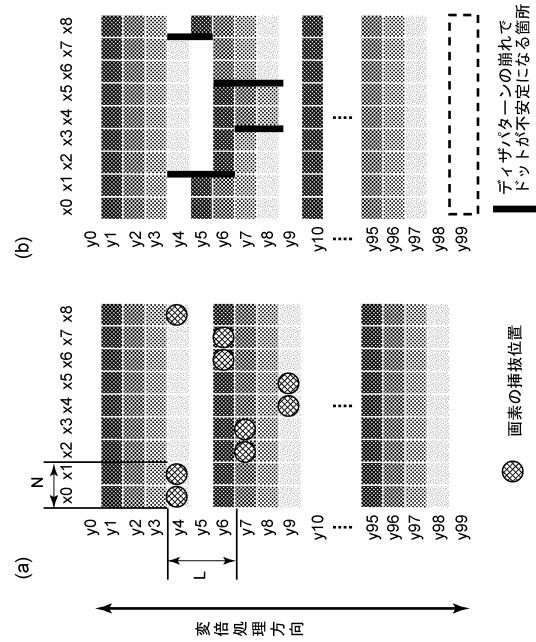
【図8】



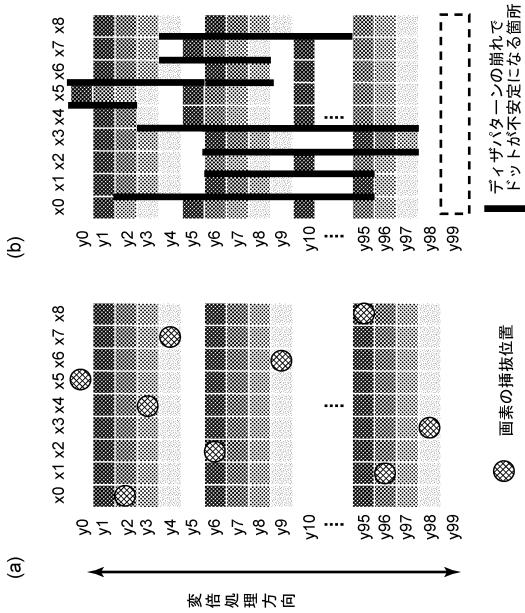
【図9】



【図10】



【図11】



---

フロントページの続き

(56)参考文献 特開昭64-047167(JP,A)  
特開昭61-206365(JP,A)  
特開昭63-232749(JP,A)  
特開2005-053106(JP,A)  
特開2000-276606(JP,A)  
特開2007-237504(JP,A)

(58)調査した分野(Int.Cl., DB名)

H04N1/405  
H04N1/38-1/393  
G06T1/00-1/40  
G06T5/00-5/50  
G06T9/00-9/40