

# デジタル複写機における細線判別技術

Thin-Line Recognition Technology for Digital Photocopiers

大木 亮\*  
Oki, Makoto

東浦 功典\*  
Toura, Kousuke

In the typical image processing of a digital copier, thin lines that deviate slightly from the strict horizontal or vertical become jaggy because they are mistakenly recognized as a region of text and are misrouted through an edge enhancement filter. To solve this problem for future generations of digital photocopiers, we developed a new image processing technology called "thin-line recognition" which can differentiate between lines and text and thus ensure appropriate processing. In this paper, we report on the development and effectiveness of thin-line recognition technology.

## 1 はじめに

コニカの複写機は、高信頼性と共に高画質を特徴とし、1988年のデジタルカラー複写機 = 8010 の発売以来、アナログ複写機では実現できなかった画像判別処理・空間フィルタ処理・自由度の高いガンマ補正処理等のデジタル画像処理技術を駆使し、高画質化を目指している。これらの技術の中で、今回我々はモノクロデジタル複写機に搭載されている画像判別技術の更なる改良として、「細線画像の判別技術」を開発した。

従来の画像判別処理では、読み込み画像の特徴点を自動判別し、文字領域・網点領域・写真領域の3つの領域に分離し、それぞれの特徴に最も適した画像処理を施し、文字の再現性・網点でのモアレ防止、写真の階調性を両立して高画質を得ていた。今回開発した「細線画像の判別技術」により、従来「文字領域」として判別されていた細線部分を分離・判別し、「文字の再現性」と「細線部分のジャギー防止」を両立する事が出来、更なる高画質化が達成できた。また、使用するメモリ量を極力少なくし、Gate Array化する際のコスト増も考慮した。

以下、本技術の内容を詳細に報告する。

## 2 デジタル複写機と画像処理

### 2.1 デジタル複写機

まず、コニカのデジタル複写機の、複写プロセスを説明する (Fig. 1、デジタル複写機の概略構成の例参照)。複写のプロセスは、おおよそ下記の手順で行われる。

- (1) 原稿画像を、CCDを使ったイメージスキャナ部 (以後「スキャナ」と称する) で読み取り、デジタル画像データに変換する。スキャナは原稿を静置し、原稿に対して光源ランプで入射光を発生し、反射した光をミラー・レンズを介してライン CCD で受光し、1次元画像を得る (この方向を「主走査方向」と称する)。

更に光源ランプ部を原稿に沿って走査させる事により (この方向を「副走査方向」と称する)、2次元画像を得る。

- (2) 読み取った画像データに対して画質操作・画像編集等の画像処理を行う。詳細は下記2.2に記述する。
- (3) 画像処理したデータをプリンタ部 (以後「プリンタ」と称する) で出力する。プリンタは、感光体ドラムに対してレーザー光をポリゴンミラーでラスタスキャンさせ、1次元画像を得る。更に感光体ドラムを回転させる事により2次元画像を得る。

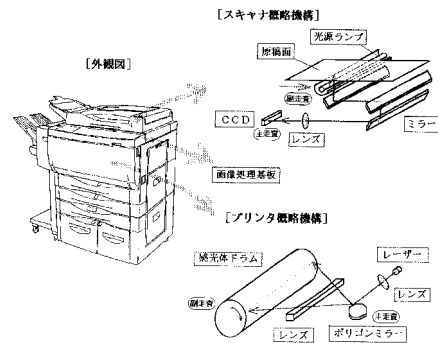


Fig. 1 Major photocopier components

### 2.2 デジタル複写機の画像処理

コニカのデジタル複写機の、画像処理回路の例を Fig. 2 に示す。また、以下に Fig. 2 の中から主な内容を説明する。

#### 画像判別処理

読み取った原稿の各領域における画像の種類を文字領域・網点領域・写真領域の3つの領域に判別する。この結果は、空間フィルタ処理及びガンマカーブ変換処理に反映される。

#### 空間フィルタ処理

スキャナ・プリンタの MTF 補正や、エッジ強調・モアレ除去等の画質補正を行う。空間フィルタの係数は画像判別処理の結果によって切り替える事ができる。

\* OD カンパニー 機器開発統括部 第3開発センター

### 誤差拡散処理（階調変換）

画像メモリへの保存ページ数確保の為、原稿画像の階調数変換（削減）を行ない、8 bit 画像をそれ以下のbit（通常は 1 bit = 2 値）画像に変換する。

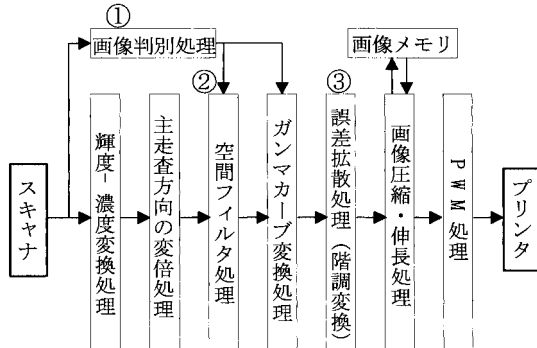


Fig. 2 Photocopier image processing

## 3 細線部分のジャギー現象

デジタル複写機ではデジタル特有の現象として「細線部分のジャギー」があった。これは主走査方向もしくは副走査方向に対し、若干斜めになった細線（以後「斜め線」と称する）をコピーすると、画素間にまたがる部分とそうでない部分とで細線の幅が周期的に変化して出力されてしまう現象である。

### 3.1 細線部分のジャギー現象の原因

スキャナで読み込んだ斜め線の原画像は Fig. 3 上側の様な 8 bit 画像であり、これに Fig. 2 に示す画像処理が施される。画像判別処理において、文字画像領域は  $M \times N$  画素領域内の濃度差から画像のエッジ周辺の領域を検出することにより判別している。従って、従来の判別処理では細線部分もエッジ成分を含んでいるので文字画像と判別される。判別結果は空間フィルタ処理に反映されるが、文字画像用の空間フィルタは文字の再現性を良くする為、エッジ強調の強い空間フィルタを使用している。この空間フィルタが細線部分にもかかり、誤差拡散処理にて 1 bit 画像になると Fig. 3 下側に示す様に、段差部分が太く、段差間部分が細い画像となり、線幅に違いが生じる。

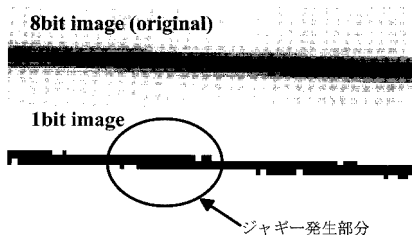


Fig. 3 Jaggies caused by a thin, slanted line

### 3.2 細線部分のジャギー現象の解決策

上述した様に、「細線部分のジャギー」現象は強いエッジ強調の空間フィルタにより起こっている。しかし単純にエッジ強調を弱めると、文字の再現性を損ねてしまう。そこで、我々は以下に示す様な方法で「細線部分のジャギー」現象の解決を試みた。

- (1) 画像判別で細線を文字領域とは別に判別する。
  - (2) 判別された細線領域に「細線部分のジャギー」現象が軽減されるエッジ強調の弱い空間フィルタを施す。
- 以下に細線検出方法の詳細を記す。

## 4 細線検出方法

### 4.1 画像判別アルゴリズムの全体構成

Fig. 4 に細線画像判別を含めた画像判別アルゴリズム構成のフローチャートを示す。細線は従来の画像判別アルゴリズムでは文字画像判別に含まれている為、文字画像検出結果より更に絞り込む形で細線領域を検出することとした。また、判別結果の合成部の優先順位は Fig. 4 の右上にある通りとした。

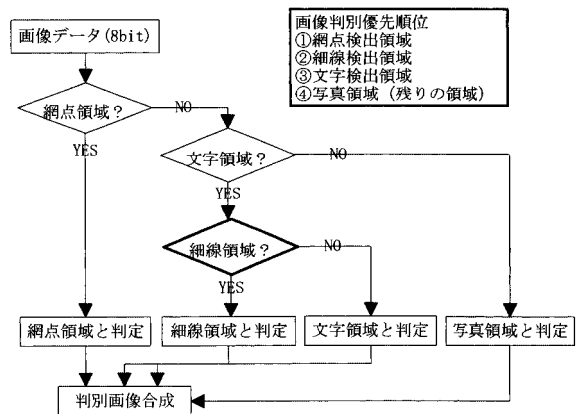


Fig. 4 Image recognition algorithm

### 4.2 細線画像と文字画像との区別

細線画像と文字画像の違いは、画像中の黒画素の直線的な連続性により判断できるが、それだけでは漢数字の「一」の様な連続部分も検出される。大きい文字ではエッジ部分の占める割合が少ないので、それほど影響はないが、細かい文字が検出されてしまうと、その再現性が悪くなる。そこで、目視観察により 14point（約 5 mm の大きさ）以上の文字は細線画像と誤判別されて、エッジ強調の弱いフィルタで処理されても文字の再現性には影響がないと判断し、14point 以上、即ち 5 mm 以上連続した黒画素を検出した場合を、細線領域の候補とする事とした。

### 4.3 斜め線の検出可能角度

「細線部分のジャギー」現象は上述した様に、斜め線において 1 画素分の段差の間隔がジャギーの周期となる。

斜め線の傾角が浅い場合、Fig. 5 の上側に示す様に、段差の間隔は長く、傾角が深くなるにつれ Fig. 5 の下側に示す様に段差の間隔は短くなる。段差の間隔が短くなると視覚的に段差を補間し「細線部のジャギー」現象が目立ちにくくなる。角度を変えて観測した結果、「細線部のジャギー」が目立つのは±10°以内の範囲であった。そこで、斜め線の検出角度のスレッシュホールドを主走査方向、副走査方向共に±10°以内とし、その検出された細線に対しエッジ強調の弱いフィルタ処理を施す事とした。

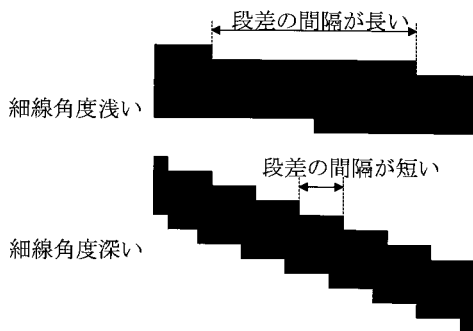


Fig. 5 Effect of line angle on jaggy step width

#### 4.4 方眼

方眼の様に、線の直交が多い部分ではジャギーが目立たない。よって、より方眼の再現性を良くする為に、細線検出では「1mm以下の間隔で書かれた方眼線は検出しない」とし、細線領域から除外した。

#### 4.5 細線検出における細線の定義

以下に細線検出における細線の定義をまとめる。

- (1) 長さ約5mm以上の線（14point以下の文字は検出しない様にする）
- (2) 傾き角が主、副走査方向より、±10°までの線
- (3) 1mm以下の間隔で書かれた方眼線は検出しない

#### 4.6 細線画像判別アルゴリズムの概略

デジタル複写機は冒頭に示したように1次元CCDにて画像を取り込んでおり、画像判別はリアルタイムの処理が必要となる。細線判別では5mm以上の黒画素の連続性を見る事により細線領域と判断している為、単純に原画像より連続性を追ってしまうと、主走査方向に原画像サイズ分、副走査方向に5mm分の8bit画像（解像度600dpiでは約822Kbyte）をメモリに記憶しなければならない。よって、今回開発した細線画像判別アルゴリズムでは、大きく2つの段階に分け、第1段階で4×4画素内の細線候補の状況を2bitデータ表現（フラグ化）する事により、検出に必要なメモリ量を削減した。各段階でのアルゴリズムの概略を以下に示す。

##### (1) 第1段階

第1段階では文字画像と判別された領域より、細線画像の可能性のある領域を分離する。

主走査方向を例に説明する。原画像において Fig. 6 にある様に、注目画素に対し斜線が引かれた4画素ずつ

（検出可能角度を10°とした為、「4.8 検出可能角度」参照）の領域H1及びH2の輝度値を調べる。そして、H1及びH2双方で4画素中1画素以上の輝度値が原稿下地との濃度差のスレッシュホールド値以上であれば Fig. 6 中の太線で囲った4×4=16画素領域を主走査方向に対して「細線の可能性がある」と判断する。この16画素分の判断結果は1つのフラグで表現し、その結果を記憶する。

副走査方向についても Fig. 6 中のV1及びV2の領域に対して同様の処理を施す。

これにより、原画像に対し1/16サイズの「細線の可能性がある領域」を表わした画像（以後「フラグ画像」と称する）ができる。1つのフラグは以下に示す様に2bitにて表現可能である。

- 0：主・副走査方向共に細線の可能性あり
- 1：主走査方向のみ細線の可能性あり
- 2：副走査方向のみ細線の可能性あり
- 3：主・副走査方向共に細線の可能性なし

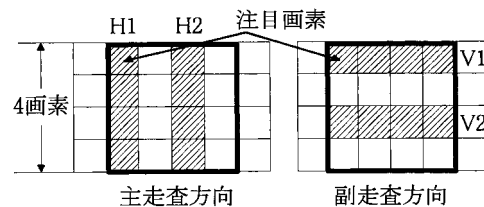


Fig. 6 Thin-line recognition algorithm, Stage 1

##### (2) 第2段階

ここでも、主走査方向を例に説明する。第1段階で得たフラグ画像において、Fig. 7 左側に示す様に注目フラグに対し、斜線の引かれた4フラグ（検出可能角度を10°とした為、「4.8 検出可能角度」参照）のFH領域を調べる。その4フラグ中1フラグ以上において「主走査方向細線の可能性あり」と判断されていれば、細線は連続しているものとしてカウントする。これを主走査方向に繰り返し、カウントが5mm（=30フラグ）以上続いた場合、その領域を細線画像として判別する。

副走査方向でも Fig. 7 中右側のFV領域に対して同様の判別をし、副走査方向へカウントしていき、その連続性で判別を行う。

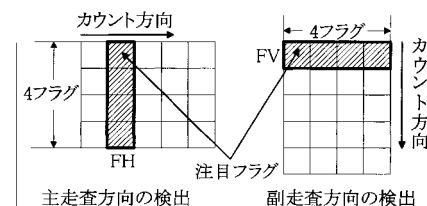


Fig. 7 Thin-line recognition algorithm, Stage 2

以上より細線画像が判別される。また、方眼を検出しない様にする為、「5mm検査する間に4回以上交線があっ

た場合は細線画像と判別しない」様にした。

#### 4.7 細線画像判別アルゴリズムに必要なメモリ量

今回開発した細線判別アルゴリズムでは、フラグ画像を用いる事により、8 bit 原画像が 1/16 サイズの 2 bit 画像にて表わされ、それを記憶すれば良い為、1/64 のメモリ量 ( 15KB ) で実機搭載可能となる。

#### 4.8 検出可能角度

細線判別アルゴリズムでは、第 2 段階において、4 × 30 フラグ分の領域、実画像の画素に換算すると、16 × 120 画素分より細線画像かどうかの判断をしている。これより、検出対象細線の線幅を 4 画素とすると、細線の検出可能角度は理論的には ± 9 °程度となり、「4.5 細線の定義」で示した目標検出可能角度の ± 10 °をほぼ満足している。

### 5 結果

#### 5.1 細線判別結果

Fig. 8 に主走査方向細線の検出結果を示す。原画像に対し、グレーに塗られている部分が細線の検出結果である。

原稿は細線を主走査方向に対して傾角 0 ~ 10 °まで変化させたものであるが、細線が約 9 °まで良好に検出されており、「4.8 検出可能角度」で求めた理論値とも一致している。また、今回開発した細線判別アルゴリズムでは主走査方向、副走査方向で異方性はないので、副走査方向も ± 9 °程度の検出角度である。

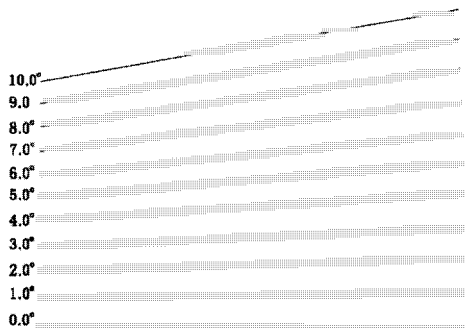


Fig. 8 Segments recognized as thin lines

Fig. 9 に文字の検出結果を示す ( 左側 : 17point、右 : 14point )。14point 文字は細線として検出されず、17 point文字では 5 mm以上連続した部分が検出されている事が分かる。

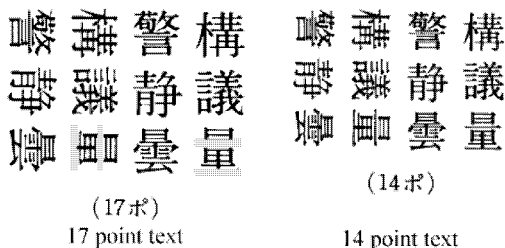


Fig. 9 Text processed with thin-line recognition

Fig.10 に方眼の検出結果を示す。1 mm間隔で線が直交している方眼の内側部分は検出されない事が分かる。

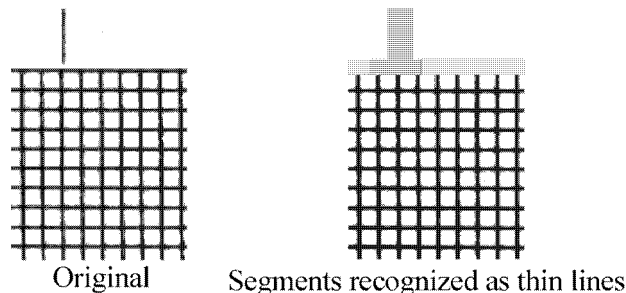


Fig.10 Graph paper processed with thin-line recognition

#### 4.2 細線判別に細線用空間フィルタを施した出力結果

Fig.11 に従来処理と、細線判別処理の処理結果を示す。細線判別処理の方がエッジ強調の弱い空間フィルタ処理により、画素間にまたがっている部分が滑らかにつながっている事が分かる。出力結果も Fig.12 に示すように、(a)の通常のコピーに比べ、(b)の細線判別処理の方が「細線部分のジャギー」が軽減されている事が分かる。

Conventional image processing



Thin-line recognition image processing



Fig.11 Image processing with and without thin-line recognition

<b>Konica</b>	
訪問年月日	- -
機械番号	
T.C	

(a) Conventional copy

<b>Konica</b>	
訪問年月日	- -
機械番号	
T.C	

(b) Thin-line recognition copy

Fig.12 The effectiveness of thin-line recognition

### 6 まとめ

- (1) 細線判別アルゴリズムを確立することができた。
- (2) 細線判別領域を別の空間フィルタで処理する事により、「細線部分のジャギー」現象を軽減できた。
- (3) 細線判別アルゴリズムに必要なメモリは約 15KB であり、Gate Array 化としても現実的なアルゴリズムを開発する事ができた。