

異なるプリンタによる印字結果の自動比較

合田 翔吾[†] 太田 直哉[†] 川端 一浩^{††} 石井美和子^{††}

[†] 群馬大学工学部情報工学科 〒376-8515 群馬県桐生市天神町 1-5-1

^{††} 沖データシステムズ 〒370-8585 群馬県高崎市双葉町 3-1

E-mail: goda@ail.cs.gunma-u.ac.jp, ohta@cs.gunma-u.ac.jp,
ka-kawabata@oki.com, mi-ishii@oki.com

あらまし 新たにプリンタを開発する場合、最終的なチェックとして実際に紙に印字を行って、その結果が基準となるプリンタのそれと同じであることを確認することが必要になる場合がある。この作業は数千枚の印字を行って比較をするが、現在はこれを目視で行っているので大変な労力を必要とする。この状況は、印字結果をスキャンし、画像処理により電子的に比較することによって改善できる。本稿ではこのための画像処理について述べる。画像処理の内容は比較すべき2つの画像の位置合わせを行ってから、それらの差異を検出して画面に表示するものである。開発した処理は実用的な実行時間と比較精度を持つことを確認した。

キーワード プリンタ印字, 画像比較, 画像差異, ロバストな位置合わせ, 射影変換.

Automatic Comparison between Printings by Different Printers

Shogo GODA[†], Naoya OHTA[†], Kazuhiro KAWABATA^{††}, and Miwako ISHII^{††}

[†] Gunma University, Computer Science Department

Tenjin-Cho 1-5-1, Kiryuu, Gunma, 376-8515, Japan

^{††} OKI DATA SYSTEMS Co., LTD.

Futaba-cho 3-1, Takasaki-shi, Gunma, 370-8585, Japan

E-mail: goda@ail.cs.gunma-u.ac.jp, ohta@cs.gunma-u.ac.jp,

ka-kawabata@oki.com, mi-ishii@oki.com

Abstract At the last step in development of a new printer, printings by the printer are compared with ones printed by a reference printer to check correctness of its performance. This process needs comparison of several thousand printings, and it is a very heavy burden for human operators. In order to improve this situation, we developed an image processing algorithm to help human operators. The printings are first scanned to digital images. Then, they are adjusted for positions and compared to detect differences. It is proven that the algorithm has good performance for practical use in processing time and accuracy.

Key words printings by printer, image comparison, image difference, robust image registration, homography

1. はじめに

新しいプリンタは多くの過程を経て開発されるが、いくつかの過程で実際に紙に印字を行って、その印字結果が正しいかどうかを確認する作業が必要になる場合がある。たとえばファームウェアのバグや印字機構の不具合等を検出するために、実際に印刷して確認することがあ

る。また明らかな不具合ではないが、印字の詳細(ハッチングの位置など)が所定の様式で印刷されるかの確認も、実際に印刷して行うことがある。

印字結果確認の具体的な作業は、まず数千種類の書類を開発中のプリンタで印字する。そしてある基準となるプリンタで印刷した結果も用意し、それらを目視によって比較する。このようにして基準プリンタと異なる印字

になっている部分を発見し、同一の印刷結果になるように開発中のプリンタに修正を加える。しかし、この検査には数千枚にもおよぶ印刷物の比較が必要であり、目視による差異の発見作業は単純で労力を必要とする作業であるため、大きなコストと時間を必要としている。また疲労による精度の低下という問題もある。そこで本研究では画像処理技術を用いてコンピュータ上でこの比較を行い、検査を支援する処理を開発する。

作業の実際は以下のように行われる。まずプリンタの出力結果をスキャナによりデジタル画像とし、コンピュータで比較する。そして差異のある部分を画面に表示し、その部分を人間が目視により最終的に確認して判断する。もちろん完全自動が望ましいが、現実的には人間が確認すべき部分を減らすことができれば十分に実用的なため、このような作業形態を想定している。この作業形態を前提にすると、画像処理での差異の検出は、問題となる差がないのに差があるとする誤り (false positive) はある程度許容できるが、差があるのにそれを見逃す誤り (false negative) はなるべく少なくする必要がある。

コンピュータ内での処理の概要は (1) 「位置合わせ」、(2) 「差異の検出」という手順を経る。印刷物のスキャン時には、紙の位置ずれや伸縮によりデジタル画像上の同一の部分に同一の内容が写っているとは限らないので、まずこれらの位置を合わせる。なお、開発中のプリンタに問題があった場合、印刷すべきパターンが印刷されないことが起こり得る。したがって、位置をあわすべき2つの画像が同一の内容でない場合も対処できることがここでは要求される。差異の検出では、基本的には画像の画素値の差が大きい部分が差異があると判断するが、ここで問題となるのは、単純な画素値の差が検出すべき差異に比例しないことである。すなわち画像パターンとしては同じ差があっても、印字結果として問題となる場合とならない場合とがある。これを根本的に解決するには、人間の視覚をモデル化する必要があるので大変困難であるが、できるだけ人間の感覚に近い差異となるよう処理を工夫した。その他、処理にはある程度的高速性も要求されるが、そのために射影を用いた位置合わせ処理を用いるなどの工夫を行った。以下位置合わせ処理、印字比較処理を説明し、次に実験による性能評価と考察を述べる。

2. 位置合わせ処理

入力画像は A4 紙を 200dpi で読み込んだ画像であり、解像度 2200 × 1700 画素である。比較すべき2枚の画像のずれはほぼ平行移動であり、スキャナの機構上縦 (Y) 方向の移動が大きい^(注1)。移動の大きさは大きくても 50 画素以内である。一方回転の大きさは最大で 2 度程度である。この移動を前提に処理を構成する。なお位置合わせ

においては、高速化のために、RGB 画像のうちから輝度情報に近い G (緑) 画像のみを用いる。

画像の位置合わせはまず画像の各部分で局所的に平行移動を求め、それらのメディアンからのずれを計算することにより、正しい移動のみを選択する。次に選択された移動情報のみを使用して画像全体の変形を求め、位置合わせを行う。以下、これらの処理を順次述べる。

a) 各セクションでの位置合わせ

画像全体を縦7つ、横5つの矩形領域 (これをセクションと呼ぶ) に分け、各セクションに対し次に述べる処理を行い、画像間の移動を決定する。

— Step (1) —

一方の画像を基準とし、その画像のセクション (基準セクション) の射影 (基準射影) を作成する。他方の画像上の同一セクション (対象セクション) の射影 (対象射影) も同様に作成する。なお、X (横) 方向の射影は次式で計算される。

$$profile(v) = \sum_{u=-projlen/2}^{projlen/2} Im(x+u, y+v) \quad (1)$$

ここで、 $projlen$ は画素値を合計する X 方向の長さであり、 x, y はセクション中心の座標である。射影を計算する領域の Y 方向の長さ n とすると、 v は $-n/2 < v < n/2$ 範囲を取る。なお対象射影を基準射影上でずらしながら一致する位置を探索するために、基準射影に対する射影の幅 (n) は対象射影のそれよりも広く取る。図1の (a) および (b) にそれぞれ基準射影と対象射影の例を示す。

— Step (2) —

印字結果には空白部分が存在することがあり、その部分のセクションでは原理的に移動を決定することができない。これを判定するために基準射影および対象射影を微分し、その合計を計算する。次式に示すように少なくともどちらか一方の射影でこの値がしきい値 k_1 以下の場合、このセクションは位置合わせに必要な情報を持たないとしてこれ以上の位置合わせを行わない。

$$\frac{1}{n-1} \sum_{i=0}^{n-1} |profile(i+1) - profile(i)| < k_1 \quad (2)$$

ここでしきい値 k_1 は経験的に 1.2 に決定した。

— Step (3) —

式 (2) の条件を満たすセクションに対して、射影 $profile(i)$ の正規化を行う。正規化は基準射影と対象射影で比較する部分のみで行うが、これは位置がずれた際の差を大きくし、射影が一致する位置の決定能力を向上させる効果がある。基準セクションの射影の幅を n_1 、対象射影の幅を n_2 とすると $n_1 > n_2$ であるので、基準射影の正規化は次式で実行される。

$$\mu(t) = \frac{1}{n_2} \sum_{i=t}^{t+n_2-1} profile(i) \quad (3)$$

(注1)：スキャナは連続スキャナであり、紙の縦 (長手) 方向にベルトで送られる構造になっているので、そのスリップなどにより縦方向の誤差が大きい。横方向には位置を決定するガイドがあるので、あまり大きな位置ずれはない。

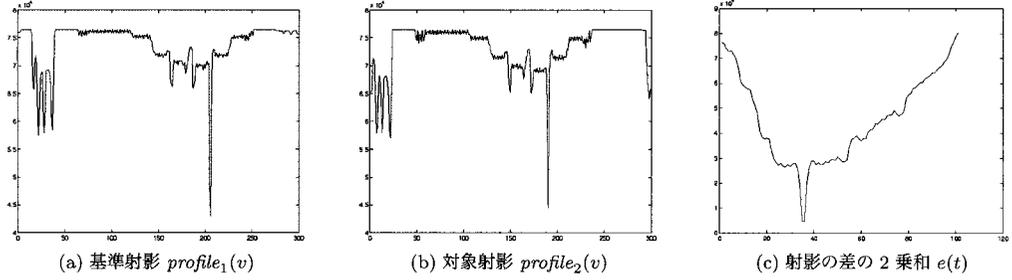


図1 2つの画像の射影とその差の2乗和の例

$$\sigma^2(t) = \frac{1}{n_2} \sum_{i=t}^{t+n_2-1} \{profile(i) - \mu(t)\}^2 \quad (4)$$

$$profile'(i, t) = \frac{profile(i) - \mu(t)}{\sigma(t) + 1} \quad (5)$$

基準射影に対しては式(3),(4),(6)の t は $0 \leq t \leq (n_1 - n_2)$ の範囲を移動する。対象射影に対しては $t = 0$ である。正規化された基準射影を $profile'_1(t+i, t)$ 、正規化された対象射影 $profile'_2(i, 0)$ とすると、それらの差の2乗和 $e(t)$ は次式で計算される。

$$e(t) = \sum_{i=0}^{n_2-1} \{profile'_1(t+i, t) - profile'_2(i, 0)\}^2 \quad (6)$$

図1の(c)に差の2乗和 $e(t)$ の例を示す。

— Step (4) —

式(6)に示した $e(t)$ を最小とする $t = t_{min}$ を、このセクションのY方向の移動量とする。ただし最小値 $e(t_{min})$ の1.05倍以内の値が他の t に対する $e(t)$ の値として存在する場合、不安定なセクションとして移動の検出を中止する。これはどちらが正しい対応か判別しにくい最小値が複数存在すると解釈されるためである。

— Step (5) —

以上述べたStep(1)～Step(4)の処理をY方向の射影に対しても行い、X方向の移動を決定する。ただしY方向の移動が既に検出されているので、射影を計算する領域の位置を、その値だけY方向に移動する。

— Step (6) —

基準セクションと、これまでに決定されたX,Y方向の移動にしたがって位置を動かした対象セクションに対して勾配法によるオプティカルフロー [3] を計算し、サブピクセル単位の移動を検出する。基準セクションの画素のX方向差分を I_x 、Y方向差分を I_y 、基準セクションと対象セクションの画素値の差を I_T として次式の行列Aとベクトルbを計算する。

$$A = \begin{pmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{pmatrix}, \quad b = \begin{pmatrix} \sum I_T I_x \\ \sum I_T I_y \end{pmatrix}. \quad (7)$$

サブピクセルレベルの移動 $V_{sub} = (V_x, V_y)$ は次式で計算される。

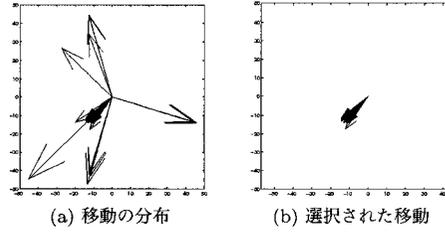


図2 メディアンによる移動の選択

$$V_{sub} = -A^{-1}b \quad (8)$$

これをStep(5)までに求めたX,Y方向の移動に加え、セクションでの移動とする。

b) 位置合わせ成功セクションの判定

これまでの過程で、移動が検出不能と判定されなかったセクションでも、その移動情報が誤っているセクションも存在する。先に述べた通り、一方の画像では存在するパターンが他方では存在しない場合など、検出不能と判定されなくても正しい移動が検出できているとは限らない。そこで各セクションの移動ベクトルのX成分、Y成分毎のメディアンを取り、メディアンから半径 k_2 の円内にある移動ベクトルを持つセクションだけを正しい移動ベクトルとして採用する。画像間の全体の移動がほぼ平行移動であると仮定すれば、この操作により、誤って検出された移動が外れ値(outlier)として排除できる。

一例を図2に示す。図の(a)に各セクションの移動ベクトルが描かれているが、これには外れ値が含まれているのが分かる。これらに対し各成分のメディアン値から半径 k_2 の円内に入る移動だけを選択すると、図の(b)に示した移動だけが選択される。なおメディアンからの半径は、ここでは経験的値として $k_2 = 7.5$ 画素を用いている。

c) 射影変換による画像の変形

ここまでの処理で決定された各セクションでの移動を用いて、最小2乗法により2つの画像間での平面射影変換(homography)を求め、それに基づいて画像を変形させることにより画像の位置合わせを行う。射影変換行列の計算にはDLT法(文献[4]、第3.1節)を用いた。ま

た画像変形の際必要となる補間法には共1次内挿法(文献[5], 442ページ)を用いた。

3. 画像の差異の検出と表示

印字結果を比較する場合、画素の単純な差と検出すべき差異の重要性とは必ずしも比例しない。そこでできるだけ検出すべき差異の重要性に応じた結果が得られるように、(1)「RGB空間での差異($diff_{RGB}$)」、(2)「 $L^*a^*b^*$ 空間での差異($diff_{L^*a^*b^*}$)」、(3)「エッジの方向と強度による差異($diff_{edge}$)」の3つの尺度で画像の差異を測り、その最大値を最終的な差異($diff$)とする方法を採用した。

$$diff = \max(diff_{RGB}, diff_{L^*a^*b^*}, diff_{edge}) \quad (9)$$

この差異($diff$)の大きさは、濃淡表示または疑似カラー表示で作業者に提示する。次に上で述べた3つの差異の計算法について述べる。

(1) RGB空間での差異

通常コンピュータで扱うカラー画像はRGB表現(ただし、測色的に正確な値であるとは限らないが)が採用されている。このRGB空間でのユークリット距離を印刷物の差異とするのが $diff_{RGB}$ である。

(2) $L^*a^*b^*$ 空間での差異

人間の感覚に近い距離を持つ色空間(均等色空間)の一つに $L^*a^*b^*$ 空間[6]がある。2番目の差異の基準 $diff_{L^*a^*b^*}$ としてこの空間でのユークリット距離を採用した。

画像の画素値(R, G, B)から(L^*, a^*, b^*)への変換は次のように行う。まず(R, G, B)値を次式で(X, Y, Z)表現に変換する。

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 2.7689 & 1.7517 & 1.1302 \\ 1.0000 & 4.5907 & 0.0601 \\ 0.0000 & 0.0565 & 5.5943 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad (10)$$

この(X, Y, Z)値を用いて次式で(L^*, a^*, b^*)値に変換する。

$$\begin{aligned} L^* &= 116 \left(\frac{Y}{Y_n} \right)^{1/3} - 16 \\ a^* &= 500 \left\{ \left(\frac{X}{X_n} \right)^{1/3} - \left(\frac{Y}{Y_n} \right)^{1/3} \right\} \\ b^* &= 200 \left\{ \left(\frac{Y}{Y_n} \right)^{1/3} - \left(\frac{Z}{Z_n} \right)^{1/3} \right\} \end{aligned} \quad (11)$$

ただし $X/X_n, Y/Y_n, Z/Z_n$ に0.008856以下のものがある場合は、式(11)の立方根の項をそれぞれ以下の式に置き換える。

$$\begin{aligned} \left(\frac{X}{X_n} \right)^{1/3} &\rightarrow 7.787 \left(\frac{X}{X_n} \right) + \frac{16}{116} \\ \left(\frac{Y}{Y_n} \right)^{1/3} &\rightarrow 7.787 \left(\frac{Y}{Y_n} \right) + \frac{16}{116} \\ \left(\frac{Z}{Z_n} \right)^{1/3} &\rightarrow 7.787 \left(\frac{Z}{Z_n} \right) + \frac{16}{116} \end{aligned} \quad (12)$$

なお、カラー画像の(R, G, B)値は必ずしも測色的に正確な値ではないので、このようにして計算された

(L^*, a^*, b^*)値も測色的なものではないが、このように計算した差異も実用的には有効に機能した。

(3) エッジの方向と強度による差異

上記の2つの差異は画素単位の差異であるが、たとえば線の方向が異なるなどの場合、画素としての差が少なくても人間の目には大きな差異と写る。このような状況を反映させるため、画像エッジの方向を用いた差異 $diff_{edge}$ を計算する。

まずRGBの画素値の平均を取ることによってグレースケールの画像とし、次にノイズを低減するために標準偏差0.7画素のガウスフィルタにより平滑化する。そして画像のX方向およびY方向の偏微分 $f_x(i, j)$ および $f_y(i, j)$ を次式で計算する。

$$f_x(i, j) = s_x * Im(i, j) \quad (13)$$

$$f_y(i, j) = s_y * Im(i, j) \quad (14)$$

ここで s_x および s_y はsobelオペレータで用いられる微分マトリクスである。次にエッジ強度 $edge(i, j)$ およびエッジ方向 $ori(i, j)$ を求める。

$$edge(i, j) = \sqrt{f_x(i, j)^2 + f_y(i, j)^2} \quad (15)$$

$$ori(i, j) = \tan^{-1} \{ f_y(i, j) / f_x(i, j) \} \quad (16)$$

エッジ強度とエッジ方向に基づいて差異 $diff_{edge}$ を次のように計算する。まず、エッジが存在するか否かを次式で判定する。

$$edge(i, j) \geq k_3 \quad (17)$$

式(17)を満たす場合にエッジが存在すると判定し、満たさない場合にはエッジは存在しないとす。なお式(17)のしきい値は $k_3 = 10$ とした。基準画像と対象画像のどちらにもエッジが存在しない場合 $diff_{edge} = 0$ とする。基準画像と対象画像のどちらか一方にエッジが存在する場合、 $diff_{edge} = 255$ とする。基準画像と対象画像の両方にエッジが存在する場合、差異 $diff_{edge}$ を次式で計算する。

$$p_e = \begin{cases} \frac{|edge_1 - edge_2|}{\max(edge_1, edge_2)} & (|edge_1 - edge_2| \geq 50.0) \\ \frac{|edge_1 - edge_2|}{50.0} & (|edge_1 - edge_2| < 50.0) \end{cases} \quad (18)$$

$$p_\theta = \frac{\min(|ori_1 - ori_2|, 2\pi - |ori_1 - ori_2|)}{\pi} \quad (19)$$

$$diff_{edge} = 0.3 * 255.0 * p_\theta^2 + 0.7 * 255.0 * p_e \quad (20)$$

ここで基準画像のエッジ強度とエッジ方向を $edge_1$ と ori_1 、対象画像のそれを $edge_2$ と ori_2 で表している。 p_e および p_θ の値域は $[0, 1]$ であるので、 $diff_{edge}$ の値域は $[0, 255]$ である。

式(18)は、大きなエッジ強度の差異であってもエッジ

表 1 位置合わせの成功率

	総数	成功数	失敗数
カラー文書	301	269 (89.4%)	32 (10.6%)
白黒文書	241	144 (59.8%)	97 (40.2%)

表 2 差異の検出精度

	総数	検出	未検出	誤検出
カラー文書	97	90 (92.8%)	7 (7.2%)	106
白黒文書	7	6 (85.7%)	1 (14.3%)	42

強度自身が大きければ評価を下げ、逆に小さなエッジ強度の中であれば強調する効果を生む。ただし極端な強調を避けるため、差異が小さい場合には分母を一定の大きさにする。

4. 実 験

本章では開発した処理の検出精度を調べるための実験と、その結果について述べる。実験の手順は以下の通りである。基準となるプリンタで印字した結果と、開発途中のプリンタでの結果をスキャンしてビットマップファイルとし、それぞれ基準画像、対象画像とする。これらの画像を開発した処理で比較し、ディスプレイに表示する。なおここでの表示方法は疑似カラー表示を用いた。この表示は最も小さい差異が黒で、値が大きくなるに従って青、水色、緑、黄、赤と変化する。この表示で赤く表示された部分が差異が検出された部分と判定し、人間が判定した正解と一致するかどうかを調べる。実験で使用した文書は A4 の大きさで、カラーのもの 301 枚、白黒のもの 241 枚である。カラーの文書は一般に写真やイラストが中心であるのに対し、白黒の文書は文字や線画が主体である。

実験の結果を表 1 および 2 に示す。開発した処理で、画像の位置合わせ処理で失敗し、差異の検出が全く行えないことがある。この数を示したのが表 1 で、カラー文書と白黒文書に対して、位置合わせが成功した数と失敗した数が示されている。位置合わせに成功したものに対しては差異の検出を行うが、その結果が表 2 である。この表の「総数」の欄が検出すべき差異(正解)の数である。次の欄が検出すべきもののうち検出した差異の数、次が検出できなかった差異(false negative)の数、最も右が差異がないにもかかわらず差異があると判定された数(false positive)である。なおここで検出すべき差異とした 97 例の中には、予め目視では発見されておらず、本処理で検出されて初めて検出すべき差異であると確認された例が 17 件含まれている。

まず表 1 の位置合わせの結果であるが、カラー文書に対しては 9 割ほど成功しているのに対し、白黒文書では 4 割が失敗している。これは、先にも述べたように、白黒文書には非常に単純な線で構成される文書が多く含まれているのが主な原因である^(注2)。その例を示したのが図 3

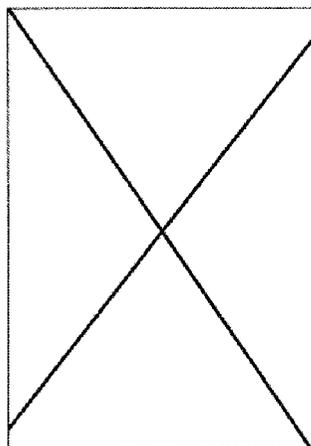


図 3 対応点決定のための情報が不足した画像

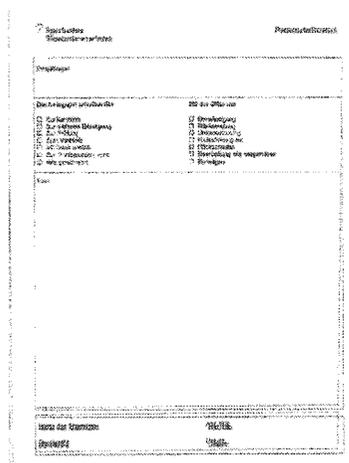


図 4 正しい移動が外れ値として誤判定される画像

および図 4 である。図 3 の例では、画像の大部分が空白か直線で、移動を決定できるセクションがほとんど無い。それでも画像四隅のコーナーで対応が求めれば位置合わせは可能であるが、画像の隅であるため位置合わせのテンプレート領域に入らず、移動を決定するのに必要な情報が得られない。また図 4 の例ではメディアンによる移動ベクトルの選別が正しく機能しなかった。この画像の場合には、上部と下部に移動を決定するのに有効な情報があり、実際にこれらを含むセクションでは正しく移動が決定されているのであるが、下部に位置するセクションの数が上部のセクションの数に比べて少なく、また移動も異なっているため、メディアンからの距離による選別を行うと下部のセクションの移動情報が外れ値として利用されず、画像の下部で正しい位置合わせに失敗する。

(注2)：移動の検出には G(緑)画像しか用いていないので、含まれる色

のバリエーションには関係がない。



図5 差異検出の失敗例

これらの問題に対しては、現在次のような方策を検討している。まず図3に示した問題には、画像の隅の画像パターンまで利用することを考えている。また図4の問題にはRANSAC(Random Sample Consensus)アルゴリズムを利用して改善することを試みている。このRANSACアルゴリズムの概要は以下の通りである。平面上の4点の移動が決まれば射影変換が一意に決まる。そこで、各セクションで検出された移動の中からランダムに4つを選択し、それによって射影変換を決定する。これを多数回繰り返して、変換を決定するのに使用しなかった移動と最も多く一致する射影変換を選択する。そして選択された射影変換に一致した全ての移動を使用して、最小2乗法で最終的な射影変換を計算する。初期的な実験の結果、これらの方策によって位置合わせの誤りを大きく改善できる見込みを得ている。

位置合わせが成功した画像の差異検出に関しては、表2に示したように9割程度の検出率が得られており、実用的な値になっている。ただし誤検出(false positive)の数は正しく検出した数を上回り、特に白黒文書では数倍にもなっている。もちろん差異検出のしきい値(画面に赤く表示されるdiffの値)を大きくすればこの数は減るが、正しく検出される数も減ることになる。最低でも検出率9割を保つことは必要であるので、現状ではこの誤検出率を許容しなくてはならない。この誤検出率は許容限度であり、実際には差異を確認するオペレータの疲労させることになるので、改善が望まれる。

検出すべき差異にもかかわらず検出できなかった画像部分の例を図5に示す。図5は左から順に基準画像、対象画像、コンピュータの画面に表示する差異画像である。なお、これらの画像は実際にはカラー画像である。図では分かりにくいだが、基準画像では左右対称なグラデーションであるのに対し、対象画像では中央で上下を向く直線を境に色の変化がある。差異画像でこの変化はある程度は検出されているが、差異があると判定される(赤で表示される)までには至っていない。グラデーションの色自体が基準画像と対象画像である程度異なる場合には差異と判定するべきではないが、同程度の差でも一方に段差が無く、他方に段差があるときには差異があると判定したい。これは検出が困難な例であるが、現状ではこのような差異の検出に失敗する。

差異の検出性能に関しては、現状では実用の観点からは最低限度のところにある。未検出(false negative)はほぼ0にすることが理想で、誤検出(false positive)も検出数に対して半分以下にしたい。このため、次のステップとしてできるだけ多くの差異の判定基準(差異を表す特徴)を適用し、学習によって、その特徴から正解となるべ

き差異を出力する関数を決定することを計画している。

なお本稿で述べた処理の実行時間は、PowerMac G5 2.5GHzでは一つの書類の比較をするのに約13秒、Celeron 1.3GHzのPCでは約27秒であった。これらの処理時間のうち、位置合わせと画像比較の実行時間の割合はほぼ1対1である。

5. おわりに

本稿では、プリンタの開発時に必要となるプリンタ出力の比較を画像処理によって半自動化する試みを述べた。まずスキャナによって電子化された印刷結果は、位置合わせの後に差異の検出が行われる。検出された差異は疑似カラー画像によってオペレータに提示され、実際に問題とすべき差異あるかどうか確認される。これによって人間の労力低減と、差異の見落としを防ぐことが可能となる。

現在のアルゴリズムで検出すべき差異の9割程度が検出可能であり、実用的な精度が得られている。特に人間が見落としした差異を、本アルゴリズムによって発見できたことは重要である。しかし誤検出が正しい検出数と同程度あるのが問題で、改善が必要である。また位置合わせの失敗により、差異検出の処理まで実行できない例があるのも問題で、これに関しても改善する必要がある。次のステップとして、位置合わせの問題に対してはRANSACアルゴリズムを利用したアルゴリズムの開発、差異検出の精度向上に関しては、多くの特徴の利用と学習による差異検出を試みる予定である。

謝 辞

本研究の初期の段階は、2007年3月に群馬大学大学院を修了し、現在東芝テック株式会社の松岡恵美子氏によって進められた。ここに記して感謝する。

文 献

- [1] 中居友弘, 黄瀬浩一, 岩村雅一, 特徴点の局所的配置に基づく位置合わせを用いた文書からの書き込み抽出法, 電子情報通信学会技術研究報告 PRMU, Vol.106, No.605, 2006, pp. 61-66.
- [2] 太田直哉, 蜂須洋平, 阿久津達也, 藤山秋佐夫, DNA マイクロアレイ画像の解析, 電子情報通信学会技術研究報告 PRMU, Vol.100, No.43, 2000, pp 57-64.
- [3] N. Ohta, Image Movement Detection with Reliability Indices, IEICE Transactions, Vol. E74, No. 10, Oct. 1991, pp. 3379-3388.
- [4] R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, 2000.
- [5] 高木幹雄, 下田陽久監修, 画像解析ハンドブック, 東京大学出版会, 1991.
- [6] 大田登著, 色彩工学(第2版), 東京電機大学出版局, 2001.