

手走査型読書器における文字入力法

正 員 北島 宗雄[†]

A Character-Entry Method for a Reading Machine with Hand-Held Scanner

Muneo KITAJIMA[†], Member

あらまし 視覚障害者用読書支援装置には、ユーザである視覚障害者が読みたい部分を理解に見合った速度で読むことを可能とする機能が不可欠である。このような機能をもった読書器を実現するためには、入力装置を手走査型とする必要があり、また、手走査型入力装置により撮影された画像の貼り合わせを高速に行い、個々の文字をできるだけ速い時点で切り出せる文字入力法が必要となる。本論文では、このような読書器に適した文字入力法として、ガイドレール上に固定された小型カメラによりほぼ一定ピッチで撮影される画像の周辺分布と文字幅を用いた貼り合わせ入力法と、さらにそこで得られる帯状につながった画像から局所的な情報のみを利用して文字ピッチを推定し、それを基に個別文字を切り出す手法について述べる。

1. ま え が き

昭和55年度厚生省統計によると、わが国における視覚障害者はおよそ34万人である。このうち約半数の人は視覚による読書が不可能で、点字や録音テープによるか、または晴眼者による朗読で読書を代行している。これら図書の発行点数は昭和56年度前期には2200点強ほどであるが、活字図書の出版点数が雑誌類を除いても年間3万点にも達するのと比較して非常に少ないなど問題があり、視覚障害者用の安価で個人使用のできる読書器の開発が望まれている。そこで、筆者らはこのニーズに応えるために、汎用装置を用いた読書器の開発を試みている。

さて、既に第一次試作機として、文書中の文字を読み取り、音声に変換する視覚障害者用読書器実験システム⁽¹⁾を作成しているが、その装置では紙面からの文字入力をX-Yプロッタヘッドに取り付けられた光学系を自動的に走査させることによって行っていた。しかし、ユーザからは読みたい箇所を理解にあった速度で読みたい、換言すれば、より主体的に読書をしたという強い要望があり、この要求を満たすために、撮像系に小型テレビカメラモジュールを用いた手動走査方式による文字入力法を検討してきた。

このような機能をもつ読書器を実現するためには、順次撮影される画像の貼り合わせ入力法、および、貼り合わせ入力画像からの個別文字の切り出し法を確立しなければならない。貼り合わせ入力法については、貼り合わせを行う2枚の2値画像間の画素毎の差の絶対値の総和として表わされる残差を尺度として貼り合わせ点を求める方法⁽²⁾が提案されているが、そこでは、実時間処理を可能とするために専用のハードウェアが用いられている。一方、本読書器は汎用装置による構成が前提とされているので、撮影周期、計算時間の両面から上記手法をそのまま適用することは難しいと考えられ、固有の手法を開発することが必要となる。また、個別文字の切り出しについては、文字が撮影された後できるだけ速い時点でその文字を切り出せることが望ましい。したがって、従来行われているような紙面全体の情報を利用した文字の切り出し法ではなく、局所的な情報のみを利用して文字の切り出しができるような手法が必要となる。

本論文では、上に述べた要請を満足できる手法として、ほぼ一定ピッチで撮影される画像の周辺分布と文字幅を用いた貼り合わせ入力法と、その結果得られる帯状の画像から文字ピッチを推定し個々の文字を切り出す手法を提案し、さらに、本手法を用いた文字入力実験について述べる。

[†] 製品科学研究所人間工学部、茨城県
Human Factors Engineering Division, Industrial Products
Research Institute, Ibaraki-ken, 305 Japan

2. 貼り合わせによる画像入力

2.1 入力装置

文書の入力には、手走査方向に一定ピッチで溝の刻まれたガイドレール上に固定されたテレビカメラをガイドレールに沿って動かしてクリックがかかる度に撮影して行われる(図1)。ガイドレールは、手走査方向に直角な方向には自由に動かせるが、ネジで固定することによりこの方向へのずれをなくすることができる。さらに、文字サイズに合わせて拡大率を変えることができるので、小さな文字であっても十分な分解能で撮影でき、また、一行だけ撮影することが可能である。

以上が入力装置の概略であるが、本装置を使って文書の入力を行うためには、ユーザは走査を開始する前に、文書がどのような書式で印刷されているか、すなわち、縦書き、横書きの判定や、文字列の検出を行っておくことが必要である。ユーザは、その書式に合わせてテレビカメラを動かして行くことになる。本装置では、撮影画面の中心付近の適当な方形領域内の輝度に応じて音量を変化させ、それを手掛りとして与えることにより、ユーザが自分で書式の判定を行い、また、同時に拡大率を変化させることにより、撮影領域に1行だけ含まれるように撮像系を設定することができるようにしてある。

以下では、読み取り対象とする文字列だけが撮影領域に含まれるように撮像系が既にセットされていると仮定し、また、縦書きの文書を対象として議論を進め

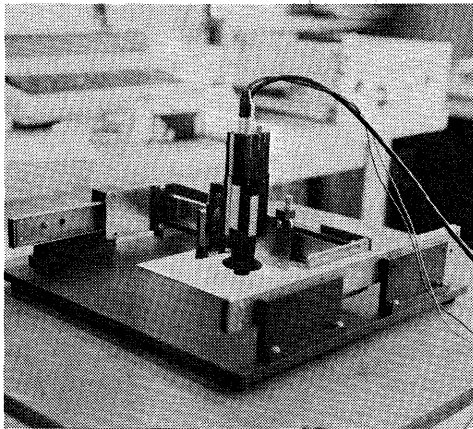


図1 文字入力部の外観

Fig.1 Outside appearance of character-entry device.

る。なお、横書きの文書に対しても紙面を90度回転して撮影することにより、以下の議論をそのまま適用することができる。

2.2 貼り合わせ入力

入力開始時には、貼り合わせピッチは光学系の拡大率によって変化するために未知である。また、連続して得られる画像のピッチはほぼ一定ではあるが、走査機構にあそびがあるので、精密な画像入力を行うためには連続する2枚の画像を相対的に数画素分動かして類似度を求めて貼り合わせる必要がある。

2枚の画像間の類似度を測る尺度の一つに残差があるが、残差計算は画像を2次元情報として用いるために計算量が多くなり処理時間がかかるので、ここでは画像を手走査に直角な方向に投影して得られる1次元情報を基に類似度を求め処理時間の短縮を図る。実際、こうすることにより処理ステップ数を残差計算の場合のほぼ1/(手走査に直角な方向の画素数)に減らすことができる。

貼り合わせに利用できる情報の一つに、手走査方向に平行な線上の黒画素の周辺分布がある。しかしながら、貼り合わせ点近傍の周辺分布が平坦になっている場合や周期性をもっている場合には、正確に貼り合わせ点を定めることが困難となる。したがって、正しく貼り合わせを行うためには他の情報を併用する必要がある。

撮影された画像から容易に得られ、しかも貼り合わせに有効と考えられる情報に文字幅がある。文字幅 w は手走査に直角な線上の最左端にある黒画素と最右端にある黒画素間の距離である。なお、この線上に黒画素が存在しない場合には $w=0$ とする。

いま、図2に示すように貼り合わせを行う2枚の画像をそれぞれ F, F' 、各画像上で手走査方向を x 軸、それに直角な方向を y 軸、上左端を原点とし、 F, F' における周辺分布をそれぞれ $h(x), h'(x)$ 、文字幅をそれぞれ $w(x), w'(x)$ とする。なお、画像のサイズは横 H 画素および縦 H 画素である。貼り合わせ候補点を s としたとき、貼り合わせのための評価関数 $d(s)$ は、周辺分布と文字幅の和

$$u(x) = h(x) + w(x), \quad u'(x) = h'(x) + w'(x)$$

を用いて、

$$d(s) = \frac{\sum_{x=s+1}^H |u(x) - u'(x-s)|}{\sum_{x=s+1}^H \{u(x) + u'(x-s)\}} + \alpha |s - \bar{s}_p|$$

$$(x_m \leq s < H - x'_m; 0 \leq x_m; 0 \leq x'_m) \quad (1)$$

と表される。ただし、分母が0の場合には、 $d(s) = D$ とする。ここで、 x_m, x'_m はそれぞれ貼り合わせの上方および下方マージンであり、貼り合わせ候補点の範囲を予め制限することにより誤った貼り合わせが行われることを防ぐために導入されている。また、 \bar{s}_p はその時点までに得られた貼り合わせ点の平均値、 α は \bar{s}_p からのずれを評価関数に反映させるときの重みであり、貼り合わせ開始時には $\alpha = 0$ 、その後は $\alpha = 1$ となっている。

貼り合わせ点 s_p は、貼り合わせ範囲内 $[x_m, H - x'_m)$ で評価関数の値が最小となる点で、しかも、その値がしきい値 D よりも小さいという条件、

$$s_p = \arg \left\{ \min_{x_m \leq s < H - x'_m} d(s) \right\}, \text{ かつ } d(s_p) < D \quad (2)$$

を満たす点である。この条件を満たす点が存在しないときには本手法による貼り合わせ処理ができないが、その原因として、真の貼り合わせ点を \hat{s} としたとき画像 F の $\hat{s} < x \leq H$ 、および、画像 F' の $1 \leq x < H - \hat{s}$ の領域のいずれにも黒画素が存在せず貼り合わせのための情報がない場合と、少なくとも一方の領域に黒画素が存在するが本来2次元であった情報を圧縮したために上述の評価関数では不十分な場合の2つが考えられるが、これらの場合には平均貼り合わせ点 \bar{s}_p を用いて貼り合わせを行う。

貼り合わせ処理の一例を図2に示す。これらの図には貼り合わせを行う2枚の画像とそれらから得られる周辺分布 $h(x)$ 、あるいは周辺分布と文字幅の和 $u(x)$ 、および評価関数 $d(s)$ が示されている。

図2(a)は周辺分布のみでは正しい貼り合わせが行えなかった場合である。貼り合わせ点近傍(図で斜線を施した部分)で周辺分布が平坦になっているために評価関数もその近傍で平坦になり、貼り合わせ点が正確に決められていないことが分かる。

図2(b)は(a)と同じ画像を周辺分布と文字幅の和を用いて貼り合わせた結果を示している。文字幅を導入することにより評価関数に鋭い谷が現れ正確な貼り合わせが行われていることが分かる。なお、これらの例では $\alpha = 0$ となっており平均貼り合わせ点からのずれは考慮されていない。

図2(c)は $\alpha = 1$ として貼り合わせを行った例であり、貼り合わせ点近傍で周辺分布と文字幅の和が周期性をもつ場合にも平均貼り合わせ点からのずれを評価に加えることにより正確に貼り合わせが行われていることが分かる。

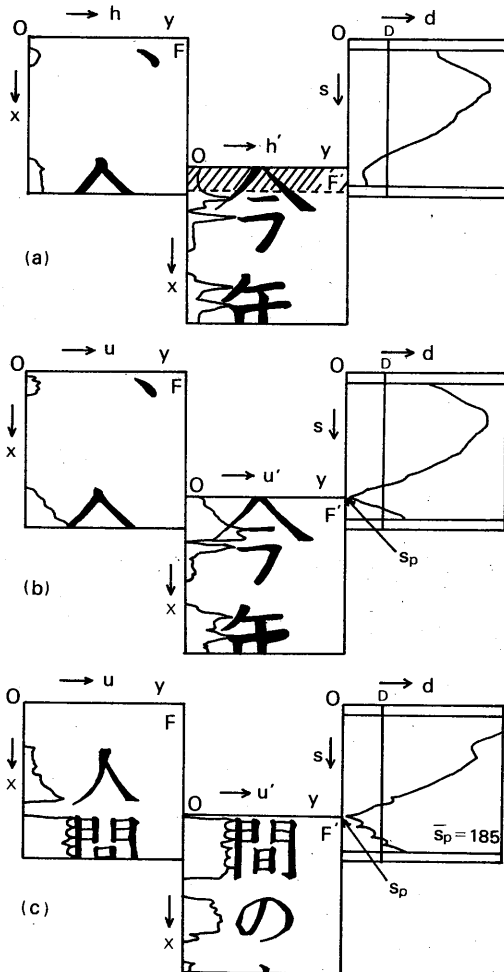


図2 貼り合わせ点の決定
Fig.2 Determination of a patching point.

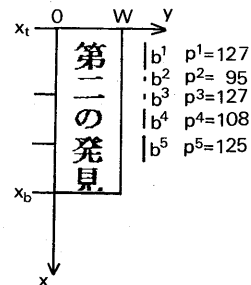


図3 候補ピッチの抽出
Fig.3 Extraction of candidate pitches.

なお、パラメータの値は次のようになっている。

画像サイズ： $H=256$ ； $W=256$ ；

マージン： $x_m=16$ ； $x'_m=16$ ；

しきい値： $D=50$ ；

3. 文字の切り出し

本章では、貼り合わせをして得られる帯状の画像からの文字の切り出し法について述べる。

既に提案されている文字の切り出し法としては、周辺分布を用いる方式⁽³⁾、8連結ブロックを用いる方式⁽⁴⁾があるが、いずれも文書全体の画像情報から文字ピッチの推定、および、個別文字の切り出しを行っている。一方、手走査型読書器においては、文書全体の情報を利用できないばかりでなく、取り込まれた画像から文字をできるだけ速く切り出し、文字認識、読みの決定を行い音声として出力する、言い換えれば、撮像部に可能な限り追従して音に出して読むという機能が最も必要とされるため、順次入力される画像からの文字ピッチの推定、および、個別文字の切り出しは、極めて局所的な情報のみを利用して行われなければならない、従来の手法をそのまま適用することは困難であると考えられる。

以下、文字ピッチの推定、文字の切り出しについて順に述べる。なお、ここでは、文庫本等の一般書籍に含まれる漢字、仮名文字を切り出しの対象に限定しており、英小文字やアラビア数字等の半角文字や、文字同士が接触している場合の個別文字の切り出しは考慮されていない。

3.1 文字ピッチの推定

図3に示すように入力画像の手走査方向を x 軸、それに直角な方向を y 軸とする。いま、文字ピッチの推定のために利用される画像領域を $g=\{(x,y) | x_i \leq x \leq x_b; 1 \leq y \leq W\}$ とする。ここで、 x_i は、処理対象領域の始点であり、文字ピッチが決定される度に更新される。また、 x_b は、処理対象領域の終点で、最後に入力された画像の最下端に等しい。文字ピッチは、貼り合わせの場合と同様に周辺分布と文字幅を利用して候補文字ピッチ列を求め、それをもとに推定される。

x における周辺分布および文字幅をそれぞれ $h(x)$ 、 $w(x)$ 、また、 g において周辺分布が0とならない i 番目の連続領域 $[x_i^i, x_b^i]$ を、文字塊領域 b^i とし、こうして得られた文字塊領域の総数を N_g とする。 i 番目の候補文字ピッチ p^i は、領域 b^i の区間長とその領域における最大文字幅の大きい方、

$$p^i = \max(x_b^i - x_i^i, \max_{x \in b^i} w(x))$$

として得られ、順次候補文字ピッチを求めることにより、候補文字ピッチ系列、

$$P_g = \{p^1, p^2, \dots, p^{N_g}\}$$

が作成される。図3に上記手続きの一例として、貼り合わせ入力画像、“第二の発見の大きな波は、第一のと

このようにして作成された候補文字ピッチ系列 P_g からの文字ピッチの推定は次のようになされる。

(ステップ1) $\hat{p} = \max_{p^i \in P_g} p^i$

候補ピッチ系列のなかの最大候補ピッチを求める。

(ステップ2) $\tilde{P}_g = \{p^i | |\hat{p} - p^i| \leq t(\hat{p}); p^i \in P_g\}$
 $= \{p^{i_1}, p^{i_2}, \dots, p^{i_{|\tilde{P}_g|}}\}$

P_g のなかで \hat{p} との差がしきい値以下である候補ピッチを要素とする集合 \tilde{P}_g を求める。なお、しきい値は \hat{p} の関数となっている。

(ステップ3) 集合 \tilde{P}_g の要素数が予め設定されたしきい値 m 以上ならば、 \hat{p} を文字ピッチ p と推定し、処理対象領域の始点をピッチ推定に寄与した候補ピッチのなかで後ろから m 番目のものに対応する文字塊領域 $b^{i_{|\tilde{P}_g| - m + 1}}$ の直下、 $x_i = x_b^{i_{|\tilde{P}_g| - m + 1}} + 1$ に更新し処理を終える。この条件が満たされない場合は、 P_g から

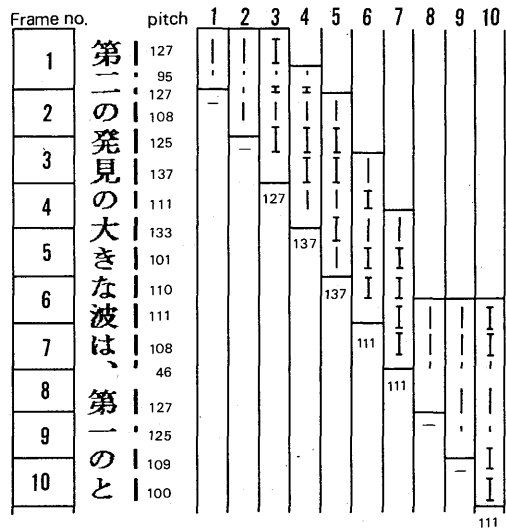


図4 文字ピッチの推定
 Fig.4 Determination of character's pitch.

\tilde{P}_g を除いた集合を新たに P_g として、また、ピッチが推定される可能性があるとき、すなわち、 $|P_g| \geq m$ のときは、ステップ1に戻り処理を継続し、その他のときは、処理を打ち切り、次の画像が入力されるのを待つ。

図4は撮影を連続して行ったときに得られる貼り合わせ画像から文字塊とそれに対応する候補文字ピッチが抽出され、それらから文字ピッチが推定されていくようすを示している。さらに、同図には、各フレーム撮影時の文字ピッチ推定のための処理対象領域とその領域に含まれる文字塊、および、文字ピッチが推定された場合にはその値と推定に寄与した文字塊(図中Iで示した部分)もまた示されている。なお、この例では $m=3$ 、 $t(\hat{p}) = \hat{p}/8$ としてある。

次に、ピッチ推定の動きを追ってみる。フレーム2まで撮影した時点では候補ピッチの変動が大きくピッチが推定されていないが、フレーム3を取りこむことにより候補文字ピッチ列 P_g は{127, 95, 127, 108, 125}となりこれから $p=127$ が推定される。このとき処理対象領域の始点は先頭の文字塊の直下に更新されるので、フレーム4を取りこんだときには $P_g = \{95, 127, 108, 125, 137, 111\}$ となりこれから $p=137$ が得られる。以下、同様にピッチが推定されていくが局所的な文字の大きさの変動に従って異なった値のピッチが推定されていることが分かる。

3.2 文字の切り出し

入力された帯状の画像からの個別文字の切り出し処理は、前節で述べた文字ピッチの推定が成功する度に起動される。文字の切り出しは、周辺分布より作成される文字塊の上下端の座標値 x_t, x_b と推定された文字ピッチ p を利用し、基本的には注目している文字塊の上端とそれに続く文字塊の下端の距離が p より大きいとき上方の文字塊を切り出すという方法により、以下に示す手順で行われる。

まず、入力画像中でまだ文字の切り出しが行われていない領域に属する文字塊を $\{b^1, b^2, \dots\}$ としておく。

(ステップ1) 統合処理

非常に近接した文字塊については、それらに間に切り出し線が引かれることがないので、それらを統合してしまう。すなわち、文字塊 b^i, b^{i+1} において、

$$x_t^{i+1} - x_b^i \leq I$$

ならば、これらを統合し、 $[x_t^i, x_b^{i+1}]$ を一つの文字塊領域とし、こうして得られた文字塊系列を文字切り出しの対象とする。なお、統合のしきい値 I の初期値

は p の関数となっていて、 $I = p/16$ としてある。

(ステップ2) 上下切り出し線の決定

上切り出し線を処理対象領域の最上位にある文字塊の上端 x_t^n 、また、下切り出し線を、処理対象領域内の隣接文字塊 b^n, b^{n+1} で、それらの上端 x_t^n と下端 x_b^{n+1} の距離が p を越すような最初の文字塊 b^n の下端 x_b^n とする。下切り出し線が決まればステップ3に進み、決まらなければ処理を打ち切る。

(ステップ3) 切り出し-1

上下切り出し線の幅 $x_b^n - x_t^n$ が p 以下ならばこの区間を1文字として切り出し、 b^n の次の文字塊を新たに処理対象の先頭としてステップ2に戻る。 p 以上ならばステップ4に進む。この段階で切り出される文字は文字の高さが p 以下の文字(図5(a))と分離文字と分離していない文字が連続している場合(図5(b))である。

(ステップ4) 切り出し-2

切り出し線幅が p より大きい場合として、文字高が p より大きい場合と図5(c)に示すような分離文字が連続して現れている場合がある。これら2つの場合の区別は n と n' を比較することによって行われ、両者が等しいときは前者、等しくないときは後者である。前者の場合には文字塊 b^n を1つの文字領域として切り出し、 b^{n+1} を次の処理の先頭としてステップ2に戻る。後者の場合には、 n から n' までの相隣接する文字塊の中で上方文字塊の下端と下方文字塊の上端の距離の最小値 I を求めてステップ1に戻る。

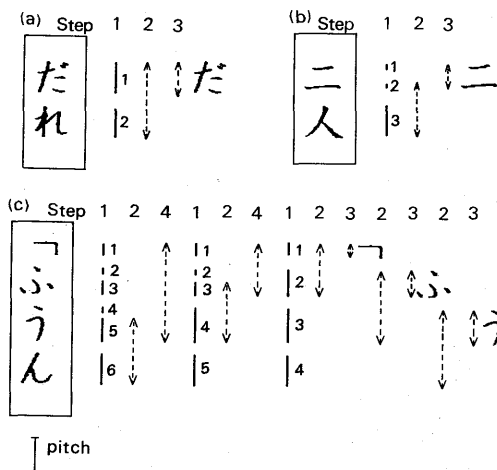


図5 個別文字切り出しの原理
Fig.5 Principles of character segmentation.

図5(c)は分離文字が連続して現れる場合に個々の文字が切り出されていくようすを示している。この例では、貼り合わせ入力画像に統合処理を施して得られる6個の文字塊が切り出しの対象となっている。これら文字塊に対してステップ2では文字塊5の下端が下切り出し線として得られるが、上下切り出し線幅が p よりも大きいのでステップ4に進み、そこでは統合のしきい値 I に文字塊4の下端と文字塊5の上端の距離を代入してステップ1に戻る。再び統合処理が行われそこで得られる5個の文字塊を対象として下切り出し線が決定されるが、上下切り出し線幅が p よりも大きいため I を求めステップ1に戻る。統合処理を施して新たに得られる4個の文字塊に対しては順次ステップ2と3が適用され、“一”、“ふ”、“う”が切り出されていく。

図6は連続して画像を入力したときに文字ピッチが推定され、それを基に文字が切り出されていくようすを示している。図には文字ピッチの推定値とそれを用いて決定された上下切り出し線、およびそれらが実行されたときのフレーム数が示されている。図から、文字ピッチが推定されると2、3文字が同時に切り出されていくことが分かる。

4. 実験結果と考察

単行本を対象として画像の貼り合わせ入力と文字の

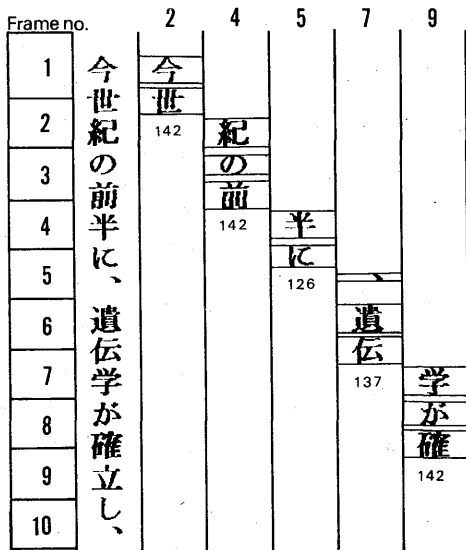


図6 個別文字切り出し結果
Fig.6 Results of character segmentation.

切り出し実験を行った。紙面に含まれている文字数は1114文字、撮影したフレーム数は730フレームである。

貼り合わせについては、概ね良好に処理が行われており、明らかに失敗していると思われる例は見当らなかった。また文字の切り出しについては、全ての文字が正確に切り出せた。

既に述べたように読書器における文字入力部として特に重要なのは、撮像系のある位置と切り出された文字の位置ができるだけ近いことである。したがって、文字の切り出し能力は、ある文字が切り出されたときにその文字の後に入力されている文字数で表される切り出しの遅れ n_d によって測ることができる。図7の実線で示したのは、 n_d の頻度分布である。全体の約半分の文字は、その文字に続いて1文字入力されることにより、また、約1/3の文字は2文字入力されることによって切り出されていることがわかる。なお、平均値は、1.86文字である。

次に、 n 個の文字が m 枚のフレーム中に等間隔に配置されている場合について切り出しの遅れの平均を求めてみる。フレーム数が文字数よりも多いとき ($m > n$) は、各文字の切り出しの遅れは1文字なのでその平均値も1文字となるが、1枚のフレームに1個以上の文字が含まれている場合には、各フレーム撮影時に切り出される文字の切り出しの遅れは次のようになる。まず、第一フレームを撮影したときには、そのフレームに含まれる $[n/m]$ 文字の中で最後の1文字を除く $[n/m] - 1$ 文字が切り出され、1番目の文字の切り出しの遅れは $[n/m] - 1$ 文字、2番目の文字は、 $[n/m] - 2$ 文字、 $[n/m] - 1$ 番目の文字は1文字となる。 f 番目 ($2 \leq f \leq m$) のフレーム撮影時には、 $[(f-1)n/m]$ 番目から $[fn/m] - 1$ 番目の文字が切り出され、切り出しの遅れは、 $[fn/m] - [(f-1)n/m]$ 、 \dots 、1文字となる。また、 n 番目の文字は最終フレーム撮影時には切り出されないが、次のフレームを撮影することにより切り出され、その遅れは $[n/m]$ 文字である。以上まとめると、平均切り出しの遅れは、

$$\bar{n}_d(m, n) = \begin{cases} 1 & (n < m) \\ \frac{1}{n} \sum_{f=1}^m \sum_{i=1}^{[fn/m] - [(f-1)n/m]} i & (n \geq m) \end{cases}$$

となる。ここで、 $[]$ はガウス記号である。さて、本実験の場合にこの式をあてはめてみると、 $\bar{n}_d(730, 1114) = 1.34$ が得られる。図7に点数で示したものが、ここで仮定した場合の頻度分布である。実験では

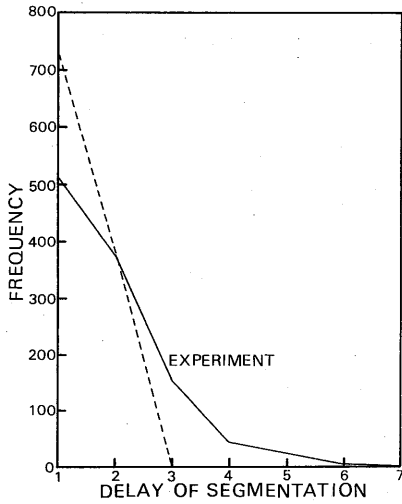


図7 切り出し遅れの分布
Fig.7 Distribution of segmentation delay.

$n_d \geq 3$ の文字が全体の20%近く存在するが、理想的な場合には存在しないのが大きな相違点である。

この違いが生じるのは、局所的なピッチ変動があるときに文字ピッチの推定に多くの候補ピッチを要し、その結果、文字の切り出しがなかなか起動されないことが原因である。まず、ピッチの頻度分布を調べてみる。図8の破線は候補ピッチの頻度分布を表しており、いくつかのピークが存在していることが分かる。これらは、ピッチの小さい方から順に、句読点、よう音、促音に使われる小さな文字、平仮名と片仮名、漢字にそれぞれ対応している、候補ピッチの値はかなりの広がりを持っている。また、実線は推定されたピッチの頻度分布を表し、最初の2つのピークは仮名に、残りのピークは漢字に対応している、ピッチ推定に利用される候補ピッチ列の違いにより異なったピッチが推定されていることが分かる。したがって、文字の切り出しを行おうとしている領域に含まれる文字の性質が急に変化するとき、例えば、漢字を多く含む文字列から平仮名を多く含む文字列に変わるときに、文字ピッチを推定するのに必要とされる文字数が多くなり、文字の切り出しがなかなか起動されず、その結果、大きな切り出しの遅れが生じることになる。

図9に文字が切り出された結果と各文字の切り出しの遅れを示す。この例に含まれる66文字の中には切り出しの遅れが4以上となる文字が6文字あるが、これらはいずれもピッチの小さい文字列からピッチの大

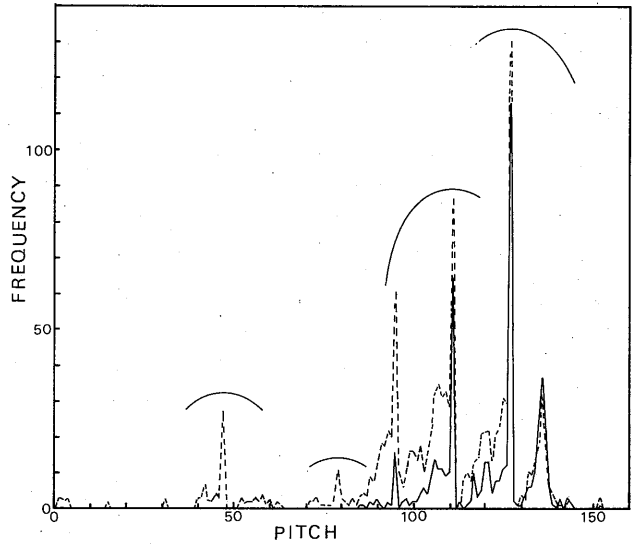


図8 候補ピッチと推定ピッチの分布
Fig.8 Distribution of candidate and conjectured pitch.

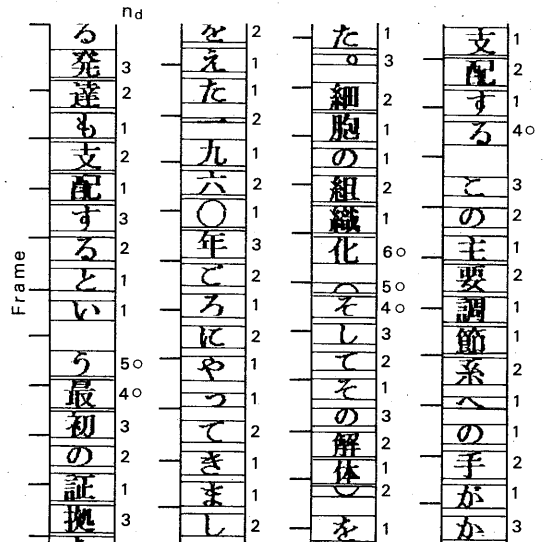


図9 切り出し遅れの例
Fig.9 Example of segmentation delay.

きい文字列に変わる部分、またはその逆の部分に存在している。

本手法ではピッチの決定を局所的に行っているために大きなピッチ変動がある部分で文字の切り出しに遅れが生じてしまう。しかしながら、このような文字が

現れる割合は小さいこと、また、処理が局所的であるために例えば図9の括弧内の文字のように大きさの異なる文字列が現れても切り出しが行えるということから、本論文で提案した文字の切り出し法は手走査型読書器における文字入力に適した手法であると考えられる。

5. む す び

手走査型読書器における文字入力法として、連続して撮影される画像を貼り合わせるにより帯状の入力画像を得るための貼り合わせ入力法と、入力された文字列から文字ピッチを推定し文字を切り出す手法について述べた。

貼り合わせ入力法の特徴は、処理に時間のかかる残差による貼り合わせを避け、周辺分布と文字幅を利用している点である。

また、文字の切り出し法は、読書器に特に必要とされる撮像部と切り出された文字の位置の近接性を考慮

して構成されているのが特徴であり、単行本を対象とした実験により本手法が読書器に適用可能であることが確認された。

今後の課題としては、本論文で述べた文字入力部に文字認識部および音声出力部を結合し、実際に読書器を構成してシステム全体としての評価を行うことが挙げられる。

文 献

- (1) 北島, 吉岡, 佐川, 増田: “文字・音声変換実験システムの概要”, 昭58信学総全大, 1454.
- (2) 岡田, 大田, 坂井: “視野の移動する画面の実時間貼合わせによる文書画像入力法”, 信学論(D), J65-D, 10, pp.1280-1285 (昭57-10).
- (3) 秋山, 内藤, 増田: “縦・横書き文書からの個別文字切り出し法”, 信学技報, PRL83-7(1983-05).
- (4) 豊田, 野口, 西村: “日本語印刷文書における文字切り出し”, 情報処理学会論文誌, 24, 4, pp.481-487.

(昭和59年8月6日受付, 11月7日再受付)