한글 認識에 있어서의 子素抽出

(A Method of the Extraction of Phonemes in Hangeul Recognition)

崔炳旭 * , 市川忠男 **, 藤田廣一 *

(Choi, B. U. , T. Ichikawa and H. Fujita)

要 約

本 論文은 한글 認識에 있어서 問題點으로 되어 있는 組合文字에서의 字素抽出 方法의 提案에 관한 것이다. 한글의 構成原理에 立脚하여 字素의 位置情報와 stroke의 方向性을 고려하여 Top-down 的으로 한글의 構造를 分析하고, 이에 따른 3段階 處理에 의하여 字素가 順次的으로 抽出되는 過程을 보였다.

日常 生活에서 使用되고 있는 文字中, 677字에 대한 字素의 抽出實驗結果, 96%의 抽出率을 얻어, 本 方式의 有効性을 確認할 수 있었다.

Abstract

This paper describes a method of the extraction of phonemes in Hangeul recognition. We provide the direction of strokes and positional information for analyzing the structure of characters based on the regular combinational rules of Hangeul according to Top-Down processing, and show the process of phoneme extraction sequencially.

In this paper, some processing algorithms are described and simulated. The experiment of the phoneme extraction is carried out for 677 characters actully used daily, and extraction rate of 96% is obtained. The experimental results demonstrate the effectiveness of the pro posed method

1. 序 論

24基本 字素로 構成되는 한글은, 子音과 母音을 포함하여 최소 2字, 최대 7字의 基本 字素가 2次元 空間에서 規則的으로 組合됨으로서 生成된다.[6] 이와 같이 組織的으로 生成되어 지는 文字의 총수는 14,364字[1]에 달하고, 日常 生活에 使用되고 있는 文字만 해도 약 1,500字[5] 程度 된다는 事實은 이미 報告된 바 있다. 이러한 한글의 機械化 問題에 있어서 文字의 複雜함과 數의 많음은, 漢字處理에서 볼 수 있는 것과 같이, 處理 過程에서 여러 가지 問題點을 發生시켰다. 그러나, 最近 字素入力에 의한 文字의 合成方式[3][6]이 發表되어 處理對象을 字素에 限定할 수 있게 되었다. 文字認識에 있어서도 認識 對象을 24基本 字素에 限定함으로서 認識 處理가 簡單해 지고, 높은 認識率이 얻어졌다.[4] 組合文字로부터 基本字素를 抽出해 내는 방법도 3, 4편의 제안이 발표되었다. 本 論文은 組合文字에 있어서의 字素 抽出 方法의 提案에 관한 것이다.

한글에서, 各 基本 字素는 stroke 의 추가에 의하여

* 正會員, 非會員, 慶應義塾大學工學部
(Faculty of Engineering, Keio University, Yokohima, Japan)
** 非會員, 廣島大學工學部
(Faculty of Engineering, Hroshima University, Hiroshima, Japan)
接受日字: 1980年 11月 3日

生成되어 지고, 文字는 이들 基本字素의 規則的인 組合에 의해서 生成된다. 筆者들은 이와 같은 한글의 組合規則과 組織的인 構造에 主目하여, 한글의 構成原理에 立脚하여 文字 補造를 分析하였다. 나아가서, 分析 結果를 利用한 3段階 處理에 의하여 字素가 順次的으로 抽出되는 過程을 보였다. 以下, algorithm의 細部와 computer simulation에 의한 實驗結果에 대하여 記述하고, 한글 認識에 있어서의 本 方法의 有效性에 관하여 論한다.

2. 한글의 文字構造

그림 1을 예로 들어 한글의 文字構造를 物理的인 관점으로부터 음미하면 그림 2와 같은 對應關系가 얻어진다. 그림 2(a)와 같이 文字圖形은 2次元空間에서 物理的으로 접촉된 stroke의 集合體(以下, block로 부르기로 한다)로 抽出可能하다. 그림 2에서 block와 字素의 對應關系는 단수의 block 와 복수의 字素, 단수의 block와 단수의 字素, 복수의 block와 단수의 字素와 같다. 따라서, 字素를 抽出하기 위해서는 우선 block를 抽出하여 必要에 따라 分離 또는 合成하여야 한다. 이렇게 해서 字素가 抽出되면 規則的인 組合에 의하여 일정한 Form[6]으로 文字가 構成된다.

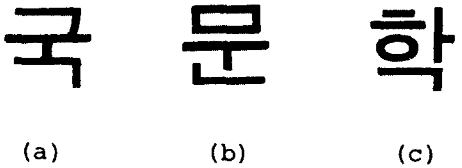


그림 1. 文字例
Fig. 1. Character samples.

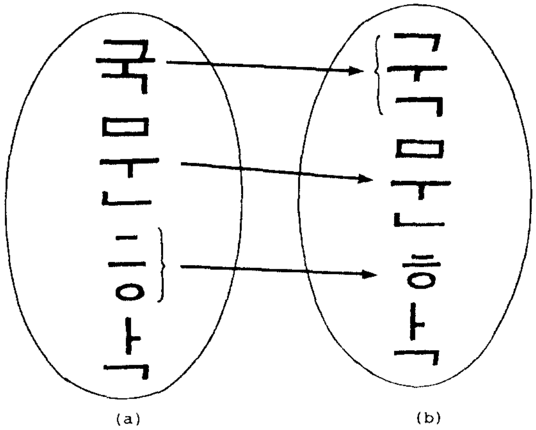


그림 2. Block와 字素의 對應關系
Fig. 2. Relation between block and phoneme

한편, 한글에 있어서 基本字素는 ㅡ, ㅣ, /, \, ㅇ 등 5種類의 stroke가 順次的으로 추가됨으로서 生成된다. 따라서, 類似한 pattern이 多數 存在한다. 母音의 경우에는 하나의 母音이 단지 位相回轉에 의하여 별개의 母音으로 生成된다.[4] 이것은 한글의 認識에 있어서 topological 特徵만으로는 字素의 識別이 어렵다는 것을 意味한다. 따라서, stroke에 方向性을 부가함으로써 字素間의 識別이 용이해 짐을 알 수 있다. 그리고, 構成되는 文字의 form에 主目하면, 基本字素는 文字形態의 複雜, 簡單에 관계없이 일정한 위치에 配列됨을 알 수 있다. 이점은 字素抽出에 있어서 情報源으로 利用可能하고, 本 硏究에서는 位置情報로 정의하여 文字의 構造 分析에 有益하게 活用한다.

位置情報는 文字 全體의 中心에 대한 block의 相對位置로 정의하고 그림 3의 d와 θ로 表示한다.

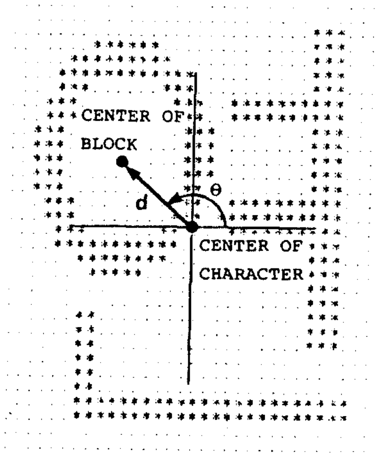


그림 3. 位置情報의 정의
Fig. 3. Definition of positional information.

이와 같이 字素에 있어서 각 stroke의 方向과 文字에 있어서 각 字素에 대한 位置情報를 導入하여 文字構造를 (2-1),(2-2),(2-3)과 같이 表現하였다.

$$S=\{B_n, C_n(n=1,2\cdots\cdots)\} \qquad (2-1)$$

$$B=\{P_i, s_j, C(i, j=0,1,2\cdots\cdots)\} \qquad (2-2)$$

$$P=\{s_k, C(k=1.2\cdots\cdots)\} \qquad (2-3)$$

여기서, S : a Syllable
B : a block
P : a phoneme(字素)
s : a stroke
C : 位置情報

單音節 S는 그림 1(b), (c)와 같이 物理的으로

독립된 몇 개의 block B와 각각의 位置情報 C를 부가 함으로서 (2-1)과 같이 表現된다. 그러나, 그림 1 (a)와 같이 單音節이 하나의 block로 構成되는 경우도 있다. 이때의 位置情報는 單音節의 中心으로 表示된다. block B는 그림 1 (b)와 같이 일반적으로 하나의 字素를 形成한다. 그러나, 그림 1 (a)와 같이 字素와 字素, 또는 字素와 stroke가 접촉된 상태로 抽出되는 경우도 있다. 따라서 B는 P_i 와 s_j 에 位置情報 C를 부가한 상태로 (2-2)와 같이 表現된다.

그러나, 그림 1 (c)와 같이 하나의 stroke만으로 抽出되는 경우도 있다. 字素 P는 複數의 stroke가 접촉된 상태에 位置情報 C를 부가하여 (2-3)과 같이 表現한다. 또한, 字素는 그림 1 (c)의 "ㅎ"과 같이 物理的으로 分離된 stroke "一", "一", "○" 를 合成함으로써 구해지는 경우도 있다.

3. 字 素 抽 出

3-1 前處理

文字 pattern은 ITV 카메라와 Image · Digitizer를 통하여 32 × 32의 畵素로 量子化되어 濃淡을 고려한 16 gray · level로 變換되어 計算機 內部에 入力된다. 이와 같이 入力된 文字 pattern은 임의로 選定된 threshold level에 의해서 "1" 과 "0"으로 2値化된다. 以下, 各畵素値 "1"을 informative, "0"을 non-informative로 가정한다. 2値化된 pattern은 noise를 줄이기 위하여 平滑化 處理를 한다. 그후, stroke를 구하기 위하여 細線化 處理[8][9]를 한다. 細線化된 文字 pattern에서 stroke는 다음과 같이 정의한다.

(a) 端點에서 端點까지의 線分(그림 4(a))

(b) 端點에서 屈折點까지의 線分(그림 4 (b))

(c) 端點에서 分枝點까지의 線分 (그림 4(c))

(d) 屈折點에서 分枝點까지의 線分 (그림 4 (d))

(a) (b) (c) (d)

그림 4. 스트로크 例

Fig. 4. Strokes.

여기서, 端點은 各線分의 끝점, 分枝點은 線分이 두갈래로 分枝되는 점을 의미한다. 또한, 屈折點은 線分이 屈折하는 점으로 曲率함수[10]를 이용하여 計算으로 구하였다. 그림 5의 口를 stroke를 構成하는 informative 畵素의 일부분으로 하고, 畵素 P에 있어서의 曲率 θ는 線分 PQ, PR이 이루는 角의 補角으로 정의한다. Q, R은 각각 P에서 一定數만큼 떨어진 畵素이다. 따라서 曲率함수는 연속되는 線分의 屈折部分에서 變化의 peak 値를 나타내고, 이 peak를 檢出함으로서 屈折點이 抽出된다.

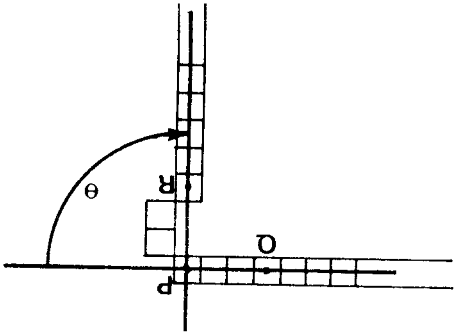


그림 5. 曲率의 表現

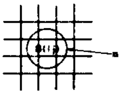
Fig. 5. Representation of curvature.

3-2. 字素抽出

前述한 文字 構造의 分析結果를 이용하면, 各 字素는 以下의 3段階 處理에 의하여 順次的으로 抽出可能하다. (1) (2-1)을 적용하여 文字에서 block 抽出, (2) 必要에 따라 (2-2)를 적용하여 不必要한 stroke를 分離除去한 字素抽出, (3) (2-3)을 적용하여 stroke 合成에 의한 字素抽出, (2)에서 分離된 stroke를 (3)에서 合成하는 경우에는 (2)와 (3)은 서로 關連性을 유지하면서 處理된다.

(1) Block 抽出

Block 는 informative 畵素를 近傍畵素에 傳播시켜, 傳播의 終了를 檢査함으로서 抽出한다. 畵素의

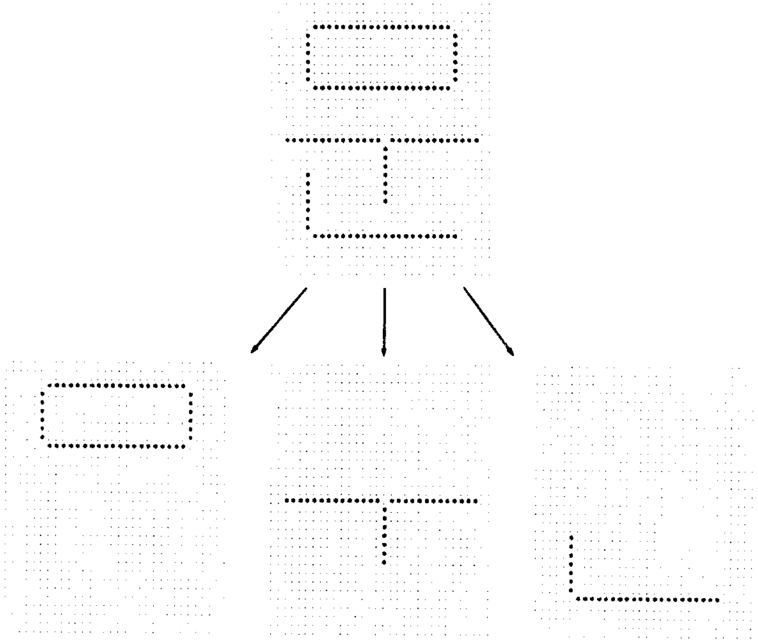


```
0 ; non-informative state of a pixel
1 ; informative state of a pixel
1'; propagated state of a pixel

S(i,j); states of neighborhood (i,j)^n
                                  (n=0,1, ... ,7)
Sp(i,j); present state of a pixel(i,j)
SN(i,j); next state of a pixel(i,j)

For all (i,j);
  (1) if Sp(i,j)=0 then SN(i,j)=0
  (2) if Sp(i,j)=1' then SN(i,j)=1'
  (3) if Sp(i,j)=1  then SN(i,j)=1'
                (when ∃((i,j)|Sp(i,j)^n=1'))
                      =1 (otherwise)
```

그림 6. Block 抽出의 앨고리즘

Fig. 6. Algorithm of block extraction.

傳播過程에 있어서 各 画素는 "0", "1", "1′" (傳播되어진 画素) 중, 어느 한 값을 갖는다. 處理 Algorithm을 그림 6에 보인다. Algorithm의 記述에 있어서는 並列處理를 想定했다. 그림 6에서 (3)이 画素의 傳播을 나타낸다. 그림 7에 그림 1(b)의 處理例를 보인다. 前述한 바와 같이 block는 일반적으로 하나의 字素로서 形成된다. 따라서, 抽出되어진 block는 正規化(後述)된 후에 미리 記億되어진 字素의 標準 pattern과의 對照에 의해서 字素로서의 抽出可能性이 檢討된 다. 標準 pattern은 24 基本字素에 重子音 "ㄲ", "ㅆ", "ㅉ"과 複母音 "ㅐ", "ㅔ"를 포함한 29字素에 限定하여, stroke 數와 方向을 고려하여 正規化한 후 位置別로 分類하여 記憶시킨다. 本 論文에서는 pattern 間의 對照의 細部에 대하여는 紙面上 文獻〔7〕에 양보한다.



(a) (b) (c)

그림 7. Block 抽出의 例

Fig. 7. Extraction of blocks.

(2) Stroke 分離

抽出되어진 block는 標準 pattern과의 對照후 必要에 따라 不必要한 stroke의 分離除去가 행하여 진다. stroke 分離는, Gestalt 心理學에서 지적되는 "매끄러운 連續의 法則, 또는 보기 좋은 形態의 法則"[11]을 工學的으로 實現한 圖形分節[12]의 概念을 도입하여, 分枝點에서 각 stroke의 進行方向을 參照하며 處理된다.

그림 8의 "국"에서 分枝點은 Q_1, Q_2, Q_3으로 抽出된다. 첫째, 分枝點 Q_1에서 stroke의 進行方向은 "←→"과 같고, 여기에 매끄러운 連續의 法則을 이용하여 "ㄱ"와 "ㅜ"으로 分離된 pattern은 平滑化 處理되어 그림 8(a),(b)과 같이 되고, 각 block는 標準 pattern과 對照되어 字素로서의 抽出可能性이 檢討된다. 그 결과, "ㄱ"은 독립된 字素로서 "ㅜ" 은 block로서 남는다. 둘째, 分枝點 Q_2에서 stroke의 進行方向은 "→→"과 같고, "ㅜ"은 "ㅡ"와 "ㅜ"으로 分離된다. 마찬가지로, 分枝點 Q_3에서 "ㅜ"은 "ㅣ"와 "ㄱ"으로 分離된다. 그러나 分離되어진 pattern "ㅡ", "ㅣ"는 標準 pattern과의 對照에 있어서, "ㅡ"는 字素로 抽出 可能하나 "ㅣ"는 位置情報가 부적당하여 字素로서 抽出 不可能하다. 따라서 Q_2에서는 stroke 分離가 不可能하고, Q_3에서 "ㅜ"와 "ㄱ"으로 分離되어 그림 8(c), (d) 와 같이 抽出된다. 그러나, pattern에 따라서는 가능한 조건속에서 반복된 處理·후에도 stroke 分離가 실패되는 경우가 있고, 그러한 block에 대해서는 字素抽出處理가 棄却된다.(그림 14 참조)

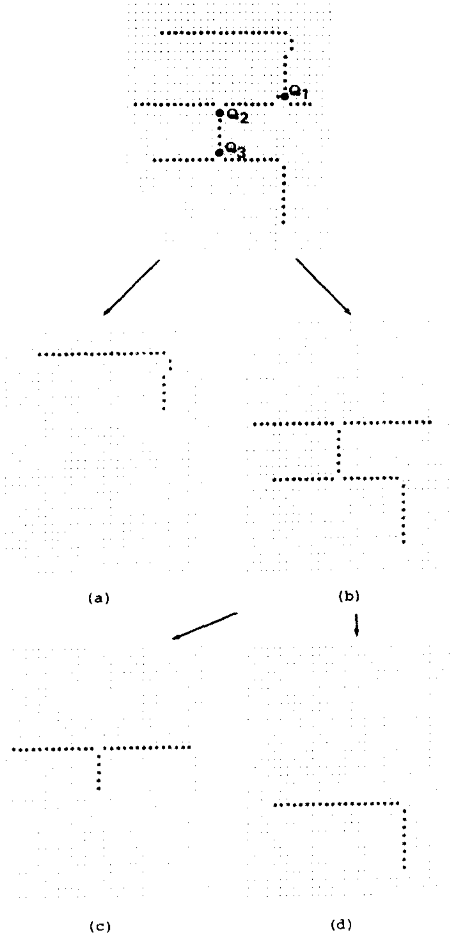


그림 8. 字素抽出의 例
Fig. 8. Extraction of phonemes.

(3) Stroke 合成

그림 1(c)의 "ㅎ"과 같은 字素는, 物理的으로 서로 독립된 stroke "—", "--", "ㅇ"로서 抽出되기 때문에 이들 stroke의 合成에 의하여 하나의 字素로서 抽出된다. 字素로서 合成可能한 stroke의 選定은 位置情報를 參照한 文字 組合 規則과 合成 可能한 stroke의 集合體와 標準 pattern과의 對照에 의하여 處理된다.

그림 9에서 C는 幾何學的으로 구해지는 文字의 中心 S_1, S_2, S_3은 각각 "—", "--", "ㅇ"의 中心을 表示한다. S_{12}, S_{23}은 각각 S_1과 S_2, S_2와 S_3으로부터 구해진다. 같은 방법으로 S_{12}와 S_{23}으로부터 구해진 P는 C에 대한 相對位置로 表現되고 "ㅎ"의 位置情報를 나타낸다.

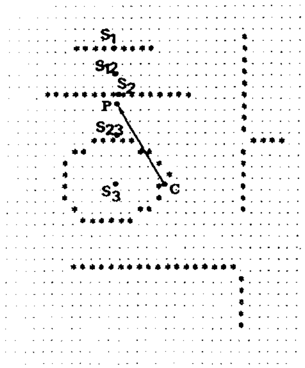


그림 9. 位置情報의 계산
Fig. 9. Caculation of positional information.

以上과 같이 位置情報를 구하고, 각 stroke의 一次元系列(後述)을 順次的으로 連結함으로서 合成可能하다. 合成되어진 pattern은 正規化된 후에, 標準 pattern과의 對照에 의하여 독립된 字素로서의 抽出 可能性이 檢討된다. 또한, "ㅊ"의 경우도 같은 방법으로 抽出된다.

3-3. 字素의 正規化

字素抽出에 있어서 stroke의 方向이 重要한 情報源으로 됨은 前述한 바와 같다. 한편, 각 stroke의 方向은 chain·code[3]에 의하여 "q" level로 量子化된 Symbol의 一次元 系列로 記述可能하다. 따라서, 複數의 stroke로 構成되는 字素는 stroke가 連結된 形態로 任意의 길이를 갖는 q元 symbol의 一次元 系列로 표현된다. 여기서, "0"와 "q-1"이 서로 인접하고 q元 要素의 길이가 동일하다고 가정하면, 一次元 系列間의 차이 즉 字素와 字素間의 거리의 大小는 각 symbol 間의 Lee-distance[14]에 對應한다. 따라서 未知 pattern과 標準 pattern과의 對照에 있어서 pattern 間의 類似度는 Lee-distance의 計算으로 구할 수 있다.

本 方式에서는 一次元 系列의 表現에 있어서 그림 10과 같이 q=8로 하고, 分枝點에 있어서 stroke의 進行方向의 優先順位는 0, 6, 5, 7, 1, 3, 2, 4의 方向으로 한다. 이 順位는 人間이 文字를 쓸 때의 書順을 경험적으로 반영시킨 것이다. 그림 7(a)를 symbol의 一次元 系列로 表記한 예를 ex.1)에 보인다. 여기서 #는 stroke 사이의 구별을 表示한다.

ex1) 00000000000000000000#766666#5
44444444444444444444#3222221

이와 같이 表現된 symbol 系列의 길이는 字素의

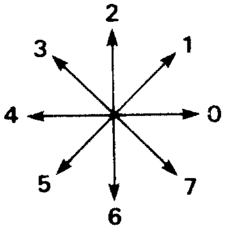


그림 10. Symbol의 方向
Fig. 10. Direction of symbols.

種類와 大小, stroke 數에 따라서 相異하다. 따라서, 標準 pattern과의 對照에 있어서 pattern 間의 類似度를 計算하기 위해서는 對照對象이 되는 symbol 系列長이 일정하게 正規化되어야 한다. 本 方式에서는 字素內의 각 stroke의 系列을 正規化한다. 處理 Algorithm을 그림 11에 보이고 記述에 있어서는 前記한 바와 같이 並列處理를 想定하였다. 正規化 處理 후의 stroke는 원래의 stroke의 方向이 보존된 채로 일정한 系列長으로 規格化된다. 上記 ex.1)의 正規化 處理 예를 그림 12에 보인다. 여기서 stroke 數는 4이고 正規化의 stroke 長은 4이다.

M is input data length and N is normalizing data length. d_L denotes a Lee-distance, and "*" indicates the removal of the symbol originally located there.

(A) when N < M,

(1) if $d_L(q_{k-1}, q_k) + d_L(q_k, q_{K+1}) \leq 1$
then $q_{k-1}q_kq_{k+1} = q_{k-1}$"*"$q_{k+1}$

(2) if $d_L(q_{k-1}, q_k) \times d_L(q_k, q_{k+1}) = 1$
then $q_{k-1}q_kq_{k+1} = q_{k-1}$"*"$q_{k+1}$

(3) if $d_L(q_{k-1}, q_k) \times d_L(q_k, q_{k+1}) = 4$ and $q_{k-1} = q_{k+1}$
then $q_{k-1}q_kq_{k+1} = \frac{q_{k-1}+q_k}{2}$"*"$\frac{q_k+q_{k+1}}{2}$ (when $q_k - q_{k-1} = 2$)
$= \left\{\frac{q_{k-1}+q_k}{2}+4\right\}$ MOD 8"*"$\left\{\frac{q_k+q_{k+1}}{2}+4\right\}$ MOD 8 (when $q_k - q_{k-1} \neq 2$)

These equations are executed with k=2n and k=2n+1 alternately. The lengths of the resulted sequences are set to M'.

(B) when N < M',
M←M' and execute (A).

(C) when N > M',
M' is segmented into blocks of length m(=2,3,...) so that each block contain at least one "*". If the number of "*" contained in a block exceeds one, one of them where original symbol kept the shortest distance concerning its adjacent symbols is replaced by its original.

(D) when N=M',
Execution is terminated.

그림 11. 正規化의 앨고리즘
Fig. 11. Algorithm of normalization.

```
DATA     ; 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0#7 6 6 6 6 6#5
           4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4#3 2 2 2 2 2 1

1st step ; 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0#7 - 6 - 6 6#5
           - 4 - 4 - 4 - 4 - 4 - 4 - 4 - 4 - 4 4#3 - 2 - 2 - 1

2nd step ; 0 - 0 - 0 - 0 - 0 0#7 6 6 6#5 - 4 - 4 - 4 - 4 - 4#3
           2 2 1

3rd step ; 0 - 0 - 0 0#7 6 6 6#5 - 4 - 4 4#3 2 2 1

END      ; 0 0 0 0#7 6 6 6#5 4 4 4#3 2 2 1
```

그림 12. 正規化의 例
Fig. 12. Example of normalization.

4. 實驗과 考察

上述한 字素抽出의 Algorithm 全搬에 대하여 Computer · Simulatin을 통해 實驗했다. 入力裝置로서 ITV 카메라와 Image · Digtizer, 中央處理裝置로 HP-9845A를 使用했다. program 言語는 Basic을 利用하였다.

實驗 對象으로서는 日常 生活에서 使用되는 고딕체의 印刷文字로서 使用度數 累積分布率 99% 까지의 文字 677字[5]를 利用하였다. 實驗에 使用된 文字중에서 자모가 서로 떨어져 있는 文字가 약 70%, 어떠한 형태로 든지 붙어 있는 文字가 약 30%였다. 實驗 結果를 Table 1, Table 2에 보인다. Table 1은 文字數와 그에 대한 抽出率을, Table 2는 使用度數를 고려한 抽出率을 나타낸다. 實驗에 있어서는 "ㄱㄱ", "ㅅㅅ", "ㅈㅈ", "ㅔ", "ㅐ"를 基本字素로 취급하여 標準 pattern化 했다. 이들 字素는 變形되어 使用되는 경우가 많고 字素의 分離 및 抽出에 있어서 難解한 경우가 많다. 한편, 이들 字素를 基本字素로 가정하여도 標準 pattern의 量이 약간 증가할 뿐, 位置情報를 參照한 文字組合規則에 어긋나는 경우가 거의 없다. 따라서, 이들 字素를 基本字素로 취급함으로써 處理를 簡單히 하고 抽出率을 높일 수가 있었다.

표 1. 抽 出 率
Table 1. Extraction rate.

Extraction rate / Characters	Number	Percentage
Correctly Extracted	651	96.3
Rejected	10	1.4
Incorrectly Extracted	16	2.3

표 2. 抽 出 率 II
Table 2. Extraction rate II.

Characters \ Extraction rate	Percentage
Correctly Extracted	98. 1
Rejected	0. 4
Incorrectly Extracted	1. 5

또한, 字素抽出을 文獻 〔6〕의 form별로 分類하여 抽出率을 구하는 방법도 있다. 그러나, 本 硏究의 實驗對象은 국형을 제외하고는 그림 1과 같이 "학"형의 문자는 모음과 바침이 서로 떨어져 있다고 가정한 것이다. Form 別로 분류하여 본 결과, form이 복잡한 文字는 使用頻度가 적었고,

cc vv, cc vv, c vv, c vv, c vv, ccv cc vv ccvv
v , v , v , v , v , cc , c , c c,
cc c cc c

의 8개의 form의 文字는 Data에 포함되지 않았다. 따라서, algorithm의 正當性을 立證하기 위하여, 現在 使用하지 않는 文字의 form "cc vv / v cc"를 제외한 7개의 form의 文字에 대하여 별도로 실험한 결과, 대개 비슷한 抽出率을 보였다.

實驗結果, 誤抽出과 棄却의 原因은 다음과 같았다. (1) 變形된 入力 pattern에 의한 error, (2) 細線化處理에 있어서의 error, (3) Stroke 分離에 있어서의 error, 以上과 같은 誤處理의 主原因은 變形된 入力 pattern이 細線化處理에 미치는 영향에 의한 error로 특히 分枝點에서 심하다. 그들은 그림 13에 보인 바와 같이 分枝點에서의 stroke의 손실로 나타난다. Top-Down에 의한 本 方法의 stroke 分離는 그림 14와 같이 stroke가 서로 연결되는 交叉 점에서 error가 發生하고 stroke의 進行方向의 判別不能에 의한 分離不能으로 나타난다. 따라서, 交叉點에서의 stroke 分離 algorithm이 今後의 課題로 남는다.

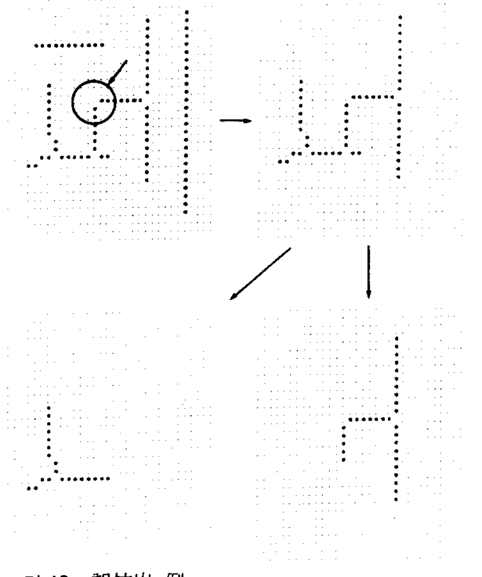


그림 13. 誤抽出 例
Fig. 13. Example of incorrected pattern.

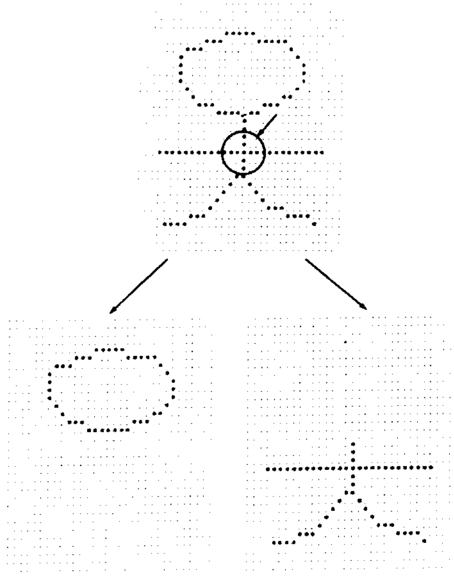


그림 14. 棄 却 例.
Fig. 14. Example of rejected pattern.

5. 結　　論

本 論文은 文字 構造 分析에 의한 한글 認識에 있어서 字素抽出 方法에 대하여 論한 것이다. 2節에서는 stroke의 方向과 字素의 位置情報를 利用하여 Top-Down 的으로 한글의 構造를 分析하였다. 3節에서는 分析結果를 이용한 3段階 處理에 의하여 字素가 順次的으로 抽出되는 過程을 보였다. 4節에서는 Algorithm 全搬에 대한 實驗에 대하여 論했다. 實驗結果, 本 字素抽出 方式은 各 文字의 form의 複雜, 簡單에 관계없이 各 文字가 갖는 고유의 特性과 前處理된 pattern에 영향을 받음을 알았고 높은 抽出率에 의해서 本 方式의 有効性을 확인할 수 있었다. 또한 algorithm을 약간 수정함으로써 표준형(국정교과서체)의 文字에 대하여도 높은 抽出率이 기대된다.

本 方式의 最大의 利點은 한글의 構成原理에 立脚한 文字 構造의 分析에 의하여 字素가 段階的으로 抽

出되므로 文字 認識에 있어서 處理가 dynamic 하게 進行될 수 있다는 점이라 하겠다. 筆者들은 現在, 한글 處理에 적당한 細線化 algorithm의 開發과 本 方式을 적용한 한글 認識 system의 實用化에 대하여 研究를 進行시키고 있다.

謝 辭

本 研究를 進行하는 데 있어서 有益한 助言과 많은 격려를 하여 주신 仁荷大學 李柱根 敎授님과 慶應義塾大學 相磯秀夫 敎授님께 感謝드립니다.

參 考 文 獻

1. 李柱根 ; 한글 文字의 認識에 관한 연구, 電子工學會誌 Vol. 9, No. 4, Sep., 1972.
2. 李柱根, 崔興文 ; 韓國語 音節의 Entropy에 관한 연구, 電子工學會誌 Vol. 11, No. 3, June, 1974.
3. 安秀桔 ; 한글문자 모아쓰기 Display의 한 방안, 電子工學會誌, Vol. 12, No. 1, Feb. 1975.
4. 이주근, 김흥기 ; 위상회전에 의한 필기체 한글의 자동인식, 電子工學會誌 Vol. 13, No. 1, March, 1976.
5. 한글 기계화 연구소 ; 한글 기계화 연구, 1975.
6. J. K. Lee ; "Korean Character Display and Recognition by Variable Combination Method", Keio Engineering Reports, Vol. 26, No. 10, 1973.
7. B. U. Choi, T. Ichikawa and H. Fujita ; "A Hangeul Recognizer and Basic Experiments", Technical Report PRL 79-111, IECE.
8. C. J. Hilditch ; "Linear Skeleton from Squre Cupboard", Machine Intelligence IV, 1969.
9. S. Yamamoto, M. Yasuda, Y. Miyamoto and M. Tsutsumi ; "Design of Hand-written Numeral Recognizer", J-IECE, Vol. 53-C, No. 10, Oct, 1970.
10. Y. Shirai ; "Analysis Intensity Arrays Using Knowledge About Scenes", The Psychology of Computer Vision, p 96, Winston P. H. ed., Mc-Graw-Hill, 1975.
11. W. Metzger ; "Gesetze des Sehens", Verlag von Waldmar Kramer, 1953.
12. Y. Uesaka and K. Tajima ; "An Interpretative Model of Figure Segregation", J-IECE, Vol. 59-D, No. 1, Jan., 1976.
13. H. Freeman ; "Computer Processing of Line-Drawing Images", Computer Surveys, Vol. 6, No. 1, March, 1974.
14. H. Miyakawa, Y. Iwatare and H. Imai ; "Coding Theory", Shokodo, p. 15-18, p. 389-391.