

論文93-30B-8-5

## OCR 시스템을 위한 화상 정렬 알고리즘과 고속 하드웨어 구현

## (A Image Alignment Algorithm for an OCR System and its Hardware Implementation)

崔完洙\*, 崔震虎\*\*, 鄭淵九\*\*\*, 金壽遠\*\*

(Wan Soo Choi, Jin Ho Choi, Yun Koo Chung and Soo Won Kim)

## 要約

본 논문에서는 OCR 시스템에서 기울어진 문서화상을 정렬하는데 사용할 수 있는 하드웨어를 제시한다. 이 화상정렬 하드웨어는 한번의 위치이동에 의해 기울어진 문서 화상을 정렬할 수 있는 새로운 알고리즘에 기본을 두고, VHDL로 시뮬레이션 한 결과 480×380 크기의 화상을 정렬하는데 소요된 시간은 약 0.065초로 측정되었다. 또한 제시된 화상정렬 알고리즘을 기존의 화상 회전 알고리즘들과 처리시간 및 특성을 비교함으로써 이 방법이 문서인식 시스템에서 유용하다는 것을 보여줄 것이다.

## Abstract

This paper presents a hardware for image alignment based on proposed new algorithm which can align a small misaligned document image simply by one transformation with a parallel shifting of pixels. This hardware is simulated with VHDL and estimated to be about 65 ms to align an image made up of 380 by 480 pixels. Also, we will demonstrate the effectiveness of the proposed image alignment algorithm in OCR system by comparing its characteristics with those of the existing image rotation algorithms.

## 1. 서론

현재와 같은 정보화시대에는 정보량이 폭발적으로 급증하고 정보의 이동속도 또한 급격히 빨라지고 있으므로 서류등을 통한 기존의 정보전달 수단은 점차

그 매력을 잃어가고 있다. 반면에 컴퓨터 기술의 발달로 전기적 신호에 의한 정보저장 및 전송이 경제성을 갖게 되었고 그 실용가치가 높아졌다. 그 일례로 자기테이프, 광학디스크, 하드디스크, CD-ROM등의 저장매체를 들 수 있다. 그러나 이러한 저장매체에 방대한 양의 문서정보를 저장하고자 할 때 사람의 수작업으로 문서를 입력한다면 막대한 시간을 낭비할 뿐만 아니라 경제적 손실을 가져오게 된다. 그래서 OCR(Optical Character Recognition) 시스템은 빠르고 효율적인 문서 입력장치로서 그 중요성이 증가하고 있다.<sup>[1,2]</sup> 그러나 문서가 스캐너를 통해 입력될 때 기울어진 문서 화상이 입력되어 기능상의 문제가 종종 일어나게 된다. 이러한 기울어짐은 문서인식

\*正會員, 大林工業專門大學 電子科

(Dept. of Elec. Eng., Daelim College)

\*\*正會員, 高麗大學校 電子工學科

(Dept. of Elec. Eng., Korea Univ)

\*\*\*正會員, 韓國科學技術研究院 시스템工學研究所

(KIST / SERI)

接受日字: 1992年 12月 26日

의 전처리 과정중에서 문자의 자모를 분류하고 특징을 추출하는데 에러를 발생시키고 결국 문자 인식을 저하시키는 요인이 된다. 따라서 기울어져 입력된 화상을 정확하게 빨리 정렬시키는 것은 OCR 시스템에서 중요한 문제이며 요구사항이다.

본 논문에서는 제안된 알고리즘에 기본을 둔 고속 화상정렬 하드웨어를 제시한다. 새로운 알고리즘은 각 화소들에 대하여 한번의 위치이동에 의해 정렬이 이루어지고 화상을 형성하는 '1'을 가진 화소의 수가 원 화상과 비교할 때 정렬된 후에도 변하지 않는다. 또한 기울어진 화상을 회전 방법에 의해 정렬시킬 수 있는 몇 가지 영상 처리에서 사용되는 화상 회전 알고리즘<sup>[3,4,5]</sup>들과 비교 분석하였다. 문서인식 시스템에서 기울어진 문서화상을 정렬하기 위해 사용되는 알고리즘들이 고려해야할 중요사항들은 문서화상을 형성하는 화소 수의 변환유무 및 처리시간이다. 영상 처리에서 사용되는 화상 회전방식들의 가장 큰 단점은 처리시간이 오래 걸린다는 것이다.

## II. 화상 정렬 알고리즘

문서인식 시스템에서 전 처리과정으로 기울어진 문서화상을 정렬하는 것은 시간이 오래 걸리는 작업이고 인식시스템의 속도를 향상시키는 장애요인이 되고 있다. 따라서 문자의 형태를 변환시키지 않고 고속으로 교정하는 방법이 필요하다. 우선 기존의 화상처리에서 사용되던 화상회전 알고리즘들을 살펴보고 본 논문에서 제안된 고속 화상 정렬 알고리즘과 비교하도록 한다.

### 1. 전통적인 회전 알고리즘

화상을  $\theta$  만큼 회전시키기 위해 오래전부터 사용되던 방법<sup>[3]</sup>은 식 (1)의 행렬식을 사용하여 정렬된 위치를 계산하는 방법이다.

$$\begin{bmatrix} X'_i \\ Y'_i \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \end{bmatrix} = T \begin{bmatrix} X_i \\ Y_i \end{bmatrix} \quad (1)$$

그러나 이 방법은 한 화소를 이동시킬 위치를 계산하기 위해 네번의 곱셈과 두번의 덧셈이 요구되기 때문에 많은 화소들로 이루어진 화상을 이동시키기 위해서는 계산시간이 오래 걸리는 단점이 있다. 또한 위치를 계산할 때 반 올림(Rounding Off)을 하기 때문에 화상이 깨지는 에러가 발생한다.

### 2. Pseudo 회전 알고리즘

Pseudo 회전 알고리즘은 전통적인 회전 알고리즘

이 한 화소를 옮기기 위해서 필요로 하는 산술적인 계산 대신에 평행이동 방법이 사용된다. 따라서 화소를 회전된 위치로 옮길 때 회전된 후의 화소 좌표를 계산에 의해서가 아니라 기준 화소에서 덧셈에 의해 이동되는 화소의 X좌표가 결정되어 이동된 후 또 한번의 덧셈에 의해 이동되는 화소의 Y좌표가 결정된다.<sup>[5]</sup> 이러한 이동들에 의해 얻어진 새로운 좌표 ( $X'$ ,  $Y'$ )를 수식으로 표현하면 다음과 같다.

$$\begin{bmatrix} X'_i \\ Y'_i \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\tan\theta & \sec^2\theta \end{bmatrix} \begin{bmatrix} 1 & \tan\theta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \end{bmatrix} = \begin{bmatrix} 1 & \tan\theta \\ -\tan\theta & 1 \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \end{bmatrix} \quad (2)$$

따라서 Pseudo 회전 알고리즘은 회전된 후의 화상 모양은 변하지 않으나, 식 (1)과 (2)로부터 알 수 있듯이 크기가 X, Y 축으로  $(1/\cos\theta - 1)$  만큼 커지게 된다.

### 3. 두 비스듬 이동 알고리즘(Twice Skew Transformations)

두 비스듬 이동 알고리즘은 Pseudo 회전 알고리즘에서 제기된 크기의 변화를 없애기 위해 제시된 알고리즘이다.<sup>[4]</sup> 기본적인 방법에 있어서는 화소가 X, Y축으로 두번의 평행이동에 의해 회전된다는 것은 크게 차이가 없고 단지 X 축 방향으로 화소들을 평행 이동할 때  $\tan^{-1}(\sin\theta)$  만큼 화상의 크기를 줄인 다음 다시 Y축으로 평행이동 시킨다는 점이 다르다. 이와 같이 하면 원 화상과 크기는 같아지나 크기를 줄이기 위해 계산되는 반 올림에 의해 회전된 화상이 깨지게 된다.

### 4. 고속 화상 정렬 알고리즘

문서인식 시스템에서 기울어진 문서화상을 정렬할 때 중요한 요소는 문자의 크기는 변해도 문자를 구성하는 자소의 형태는 변하지 않아야만 한다. 이러한 특성에 적합한 알고리즘은 Pseudo 회전 알고리즘인데 Pseudo 회전 알고리즘의 경우 두번의 평행이동이 요구되기 때문에 단지 한번의 이동에 의해 회전시킬 수 있는 고속 화상정렬 알고리즘을 제안하고자 한다.

제안된 화상 정렬 알고리즘에서 입력 문서화상의 기울어진 각도를 알아내기 위해서는 아래와 같은 방법을 문서화상 전체에 적용한다. 각 열에서 가장 왼쪽에 위치한 화소와 가장 오른쪽에 위치한 화소가 검색된다. 만약 현재열의 가장 왼쪽에 있는 화소의 X 좌표가 전 열의 가장 왼쪽에 위치한 화소의 X 좌표보다 작으면 (가장 오른쪽에 있는 화소의 경우 X좌표가

전 열의 가장 오른쪽에 위치한 화소의 X 좌표보다 크면) 현재 열의 가장 왼쪽에 위치한 화소(가장 오른쪽에 위치한 화소)가 선택된다. 문서인식 시스템에서 입력되는 문서화상이 보통 10도 이내로 기울어지기 때문에 첫 번째 열에서 X-SIZE × tan10° 만큼의 열까지 각 열에서 위의 조건이 만족되는 화소들이 선택된다. 만약 가장 왼쪽에 위치한 m개의 화소들이 선택됐다면 (3)번식에 의해 만큼 기울어진 각도가 구해지고 또한 가장 오른쪽에 위치한 m개의 화소들이 선택됐다면 같은 방법에 의해 시계방향으로 θ 만큼 기울어진 각도가 구해진다.

$$\theta = \frac{\sum_{i=1}^{m-1} \sum_{j=i+1}^m \tan^{-1} \frac{Y_j - Y_i}{X_j - X_i}}{\sum_{k=1}^{m-1} k} \quad (3)$$

$X_i, X_j$ : 순서도에서 구해진 m개의 화소중 i, j번째 X 좌표값  
 $Y_i, Y_j$ : 순서도에서 구해진 m개의 화소중 i, j번째 Y 좌표값

위와 같이 기울어진 각도 θ 가 얻어진 후 다음과 같은 방법으로 기울어진 문서 화상이 정렬된다. 그림 1에서 보여주는 θ 만큼 기울어진 문서화상을 생각하자.

- 단계 1 : 그림 1(a)에서 θ 만큼 기울어진 선분 AB에 있는 모든 화소들의 (X, Y) 좌표점들을 구한다.
- 단계 2 : 화소가 이동되는 시작 X 좌표들 (그림 1(b)에서 선분 A'D'에 있는 X 좌표들) 이 식 (4)에 의해 구해진다.

$$S_t = (Y - SIZE - t) \times \tan \theta, \text{단 } t = 0, 1, 2, \dots, (X - SIZE \cdot \tan \theta + Y - SIZE) \quad (4)$$

- 단계 3 : t가 0일 때를 시작으로 하여 단계 1에서 구한 (X, Y-t) 좌표점에 맞는 화소 값들이 단계 2에서 구한 시작 점으로 해서 그림 1처럼 이동된다.
- 단계 4 : t 값을 1씩 증가시키고 단계 2 - 3의 과정을 반복한다.
- 단계 5 : 단계 2 - 3의 과정을 그림 1(a)의 전 화상에 걸쳐 수행하면 그림 1(b)의 정렬된 화상을 얻는다.

위의 알고리즘에서 이동된 화소의 좌표는 변환 행렬 T'에 의해 이동된 것과 같으므로 다음과 같은 관계식으로 표시 할 수 있다.

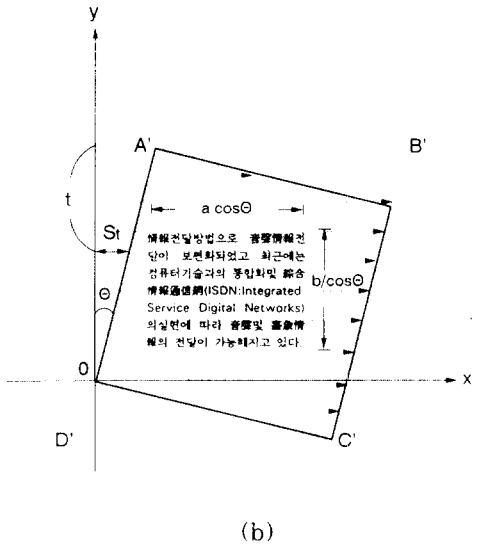
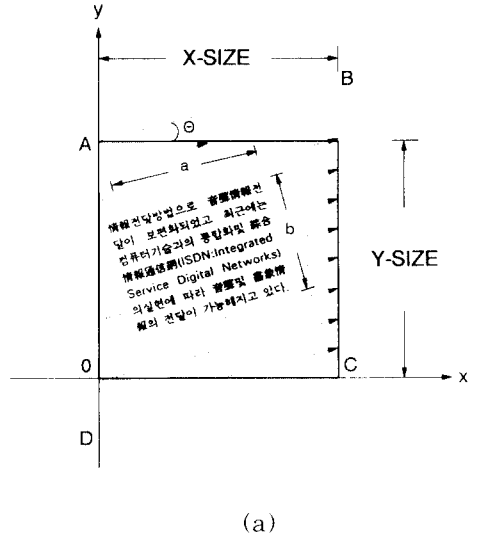


그림 1. 제안된 알고리즘에 의한 화상 정렬 방법  
 Fig. 1. Image alignment by only one transformation.

$$\begin{bmatrix} X'_i \\ Y'_i \end{bmatrix} = \begin{bmatrix} \cos^2 \theta & \cos \theta \sin \theta \\ -\tan \theta & 1 \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \end{bmatrix} = T' \begin{bmatrix} X_i \\ Y_i \end{bmatrix} \quad (5)$$

이 알고리즘은 단 한번의 위치이동에 의해 정렬되기 때문에 처리시간이 상당히 줄어든다. 그러나 원래의 문자화상과 제안된 알고리즘에 의해 정렬된 문자화상의 크기가 아래와 같이 달라짐을 알 수 있었다.

제안된 고속 화상 정렬 알고리즘에 의해 이동된 화소들의 좌표는 위의 행렬식 (5)에서 보여주는 변환 행렬  $T'$ 에 의해 나타내진다. 여기서 행렬식 (1)에 있는 변환 행렬  $T$ 는 방정식 (5)에 있는 변환 행렬  $T'$ 와 다음과 같은 관계를 갖는다.

$$T \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} = \begin{bmatrix} 1/\cos \theta & 0 \\ 0 & \cos \theta \end{bmatrix} \begin{bmatrix} \cos^2 \theta & \cos \theta \sin \theta \\ -\tan \theta & 1 \end{bmatrix} = S \cdot T' \quad (6)$$

따라서 행렬식 (6)으로부터 문자의 가로와 세로의 길이가 크기 조정 행렬  $S$ 만큼 변화한다는 것을 알 수 있다. 따라서 정렬된 후 문자의 가로 길이는 원래 문자 길이에서  $(1-\cos \theta)$ 만큼 변화하고 문자의 세로 길이는 원 글자 길이의  $(1/\cos \theta - 1)$ 만큼 변화한다. 10도로 기울어진  $64 \times 64$  크기의 문자인 경우 정렬된 후 일어나는 문자 크기의 변화는 약 한 화소 크기만큼 바뀐다. 실제로 문자 크기 그 자체는 인식을 하는데 있어서 고려되는 요소는 아니다. 왜냐하면 활자체에 따라 크기가 달라지기 때문에 문자인식에 있어서 가장 중요한 요소는 문자를 구성하는 자소의 형태이다. 정렬과정 후에도 자소의 원래 형태를 유지하기 위해서는 다음 조건들이 만족되어야 한다 : (1) 문자를 구성하는 화소 즉 이진값 1을 갖는 화소의 수가 정렬된 후에도 문자를 구성하는 화소의 수와 같아야 한다. (2) 인접 화소는 정렬된 후에도 인접 화소이어야 한다.

제안된 고속 화상 정렬 알고리즘에 의해서 기준 화소의 이동되는 위치가 결정된 후 주위 화소들이 순차적으로 이동되기 때문에 위의 조건이 만족된다. 따라서 제안된 알고리즘은 원문자를 구성하는 화소의 형태를 변화시키지 않으면서 효율적으로 빠르게 기울어진 화상을 정렬한다.

### III. 고속 화상 정렬 하드웨어

제안된 알고리즘에 기본을 둔 고속 화상 정렬 하드웨어는 그림 2와 같이 제어부, 주소발생부, 메모리부 등으로 크게 나뉘어진다.

기본적인 하드웨어의 동작은 우선 호스트로부터 하드웨어 내 주소발생부의 메모리 내로 입력 화상의 기울어진 각도에 해당하는 선의 좌표점(그림 1의 AB, A'D')들과 메모리부의 소스 메모리 내로 입력 화상 데이터가 전송된다. 그리고 하드웨어는 독립적으로 소스메모리 내의 입력 화상을 정렬하여 타겟메모리로 옮기고 호스트에 인터럽트를 건다. 마지막으로 타겟메모리 내의 정렬된 화상 데이터는 호스트 내의 메모리로 옮겨진다.

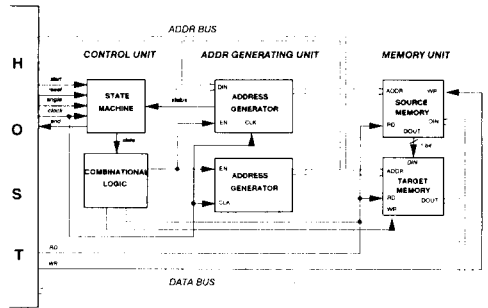


그림 2. 하드웨어 구성도  
Fig. 2. The structure of hardware.

#### 1. 각 부분의 구성과 동작

제어부(Control Unit)의 동작은 크게 두가지로 나눌 수 있다. 첫번째는 호스트와 하드웨어사이의 인터페이스를 제어하고 두번째는 주소발생부에서 입력되는 표 1의 네가지 상태(status)조건에 따라 State Machine은 그림 3과 같이 13개의 state를 천이하면서 각 state마다의 제어신호를 Combinational Logic에서 출력한다.

표 1. 상태 조건의 종류

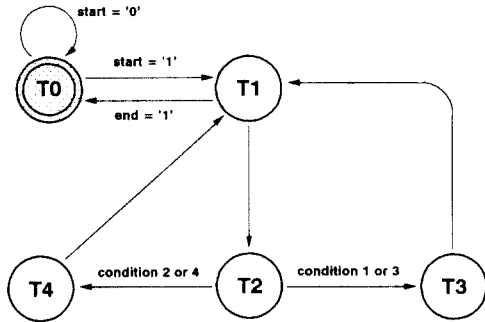
Table 1. Type of status condition.

상태조건의 종류	상태조건의 발생 원인
status condition 1	스캔의 범위가 입력된 화상의 가장 상측의 Y좌표보다 작을 때 발생한다.
status condition 2	스캔의 범위가 입력된 화상의 가장 우측의 X좌표보다 클 때 발생한다.
status condition 3	스캔의 범위가 입력된 화상의 가장 하측의 Y좌표보다 클 때 발생한다.
status condition 4	위의 경우에 어디에도 해당되지 않을 경우에 발생한다.

또한 State Machine에서 천이되는 총 state 수는 13개이나 각 상태(status)조건마다 13개의 state 모두를 거치는 것이 아니라 그림 3과 같이 단지 4개의 state만 천이하면 된다. 예를 들면 상태(status)조건이 condition 4인 경우에는 state 1→state 2→state 3→state 11의 순서로 천이한다. 그리고 제어부문의 모든 동작은 클럭에 동기하여 행하여 진다.

주소발생부(ADDR Generating Unit)은 램, 카운터, 덧셈기, 래치와 플립플롭으로 그림 4와 같이 구성되어 있다. 램에서는 그림 1의 AB와 A'D'의 선 데이터가 저장되어 있고 이 램의 데이터와 카운터의

출력이 메모리부문을 액세스하는 주소가 된다. 덧셈기와 플립플롭은 상태(status)신호를 발생하며 래치는 클럭에 동기하여 주소를 메모리부문에 넘겨주기 위해 필요하다.



- T0(state 0) : Initialize hardware.
- T1(state 1,2,4,5,6,7) : control AGU.
- T2(state 3,8) : Check the status condition.
- T3(state 9,10) : Initialize AGU.
- T4(state 11,12) : Control MU.

\* AGU : ADDR Generating Unit.  
MU : Memory Unit.

그림 3. 상태 전이도  
Fig. 3. State transition diagram.

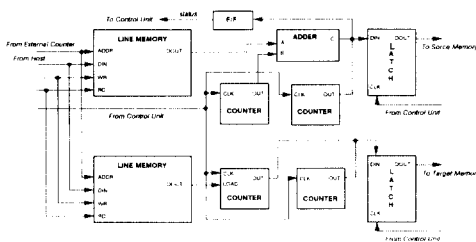


그림 4. 주소발생부문 구성도  
Fig. 4. The structure of address generating unit.

메모리부문(Memory Unit)은 램, 디코더, 멀티플렉서와 래치를 사용하여 그림 5와 같이 구성하였다. 제어부문의 write, chip select 신호와 주소발생부문에서 발생하는 주소에 따라 소스메모리에 저장된 화소 데이터가 타겟메모리 내의 정렬된 위치로 옮겨진다. 단 스캐너에서 읽어들이 입력화상은 메모리에 비트맵 방식으로 저장되어지고 소스, 타겟메모리사이의 데이터 액세스는 1 byte 단위로 행해지기 때문에

멀티플렉서와 디코더, 논리회로(AND, OR), 래치를 사용하여 1byte의 데이터중 실제 옮기고자 하는 1bit의 화소 데이터를 추출하여 정렬하도록 설계되었다.

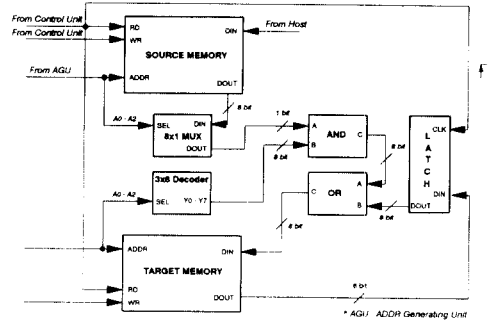


그림 5. 메모리부문 구성도  
Fig. 5. The structure of memory unit.

2. 시뮬레이션

제안된 하드웨어는 VHDL을 사용하여 각 소자별로 기술한 다음 시뮬레이션을 하였다. 또한 시뮬레이션된 소자를 사용하여 각 블록별로 다시 시뮬레이션 하였고 마지막으로 전체 시스템을 시뮬레이션 하였다. 시뮬레이션 수행 결과는 그림 6에 보여진다. 시뮬레이션에 사용된 클럭의 속도는 12MHz이고 소스메모리에는 임의의 화소데이터를 저장하였다. 또한 그림 6에서 보는 바와 같이 제안된 하드웨어는 범용의 마이크로 프로세서를 사용하지 않고 전용의 Control Unit를 사용하기 때문에 한 화소를 옮기기 위해 필요한 클럭 수를 줄일 수 있었다.

- \* Cursor 1 : Condition 4.
- Cursor 2 : Condition 1.
- Cursor 3 : Condition 2.
- Cursor 4 : Condition 3.

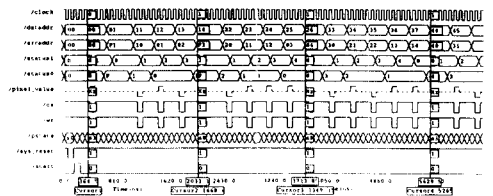


그림 6. 시뮬레이션 타이밍도  
Fig. 6. Simulation timing diagram.

시뮬레이션 수행 결과 상태(status)조건이 condition2, 4인 경우에 소스메모리 내의 화소 데이터가

타겨메모리 내로 정렬되고 condition1, 3인 경우에는 주소발생부문을 초기화만 시키고 직접적인 화소 데이터의 정렬은 발생하지 않는다. 또한 각 상태(status)조건이 발생할 경우의 수와 사용된 클럭의 속도가 정해지면 480×380 크기의 화상이 10도정도로 기울어져 있을 경우 정렬하는데 걸리는 시간은 아래와 같이 계산할 수 있다. 즉 상태(status)조건이 condition1인 경우 그림 1 (a)에서 X-SIZE의 크기만큼 condition2인 경우에 Y-SIZE의 크기만큼 condition3인 경우에 DOC의 크기만큼 condition4인 경우에는 X-SIZE×Y-SIZE 크기만큼 발생한다.

condition1 : 화상의 X-SIZE × 클럭 속도 × 4개의 상태수 = 480 × 80ns × 4 = 0.2ms

condition2 : 화상의 Y-SIZE × 클럭 속도 × 4개의 상태수 = 380 × 80ns × 4 = 0.13ms

condition3 : 화상의 X-SIZE<sup>2</sup> × tan 10° ÷ 2 × 클럭 속도 × 4개의 상태수 = 480<sup>2</sup> × tan 10° ÷ 2 × 80ns × 4 = 6.5ms

condition4 : 화상 크기 × 클럭 속도 × 4의 상태수 = 480 × 380 × 80ns × 4 = 58ms

따라서 총 시간은 0.2ms, 0.13ms, 6.5ms, 58ms를 더하면 약 65ms가 걸린다.

IV. 실험결과 및 결론

제안된 새로운 화상 정렬 알고리즘과 다른 화상 회전 알고리즘들을 PC 386상에서 C 언어로 구현하였다. 각 알고리즘에 대해서 처리시간과 그들의 특징들을 비교하였고 여러가지 기울어진 각도에 대하여 원 화상과 정렬된 화상을 구성하는 이진값 '1'을 가진 화소 수의 변환 정도를 측정하였다. 이러한 결과는 그림 7에 고속 화상 정렬 하드웨어의 처리시간과 함께 나타내었고 또한 표 2에 각 알고리즘에 대한 변환 비율을 보여준다. 그림 8은 480×380 크기의 문서화상에 대해서 제안된 알고리즘과 다른 화상 회전 알고리즘들에 의해 정렬된 문서화상들을 보여준다. 두 비스듬 이동 알고리즘<sup>[4]</sup>과 전통적 회전알고리즘<sup>[3]</sup>에서는 화상수 의 변화(화상손실)현상이 나타났다. 전자는 크기 조정과정 때문이고 후자는 계산할 때의 반올림(Rounding off)때문이다. 반면에 Pseudo 회전

알고리즘<sup>[5]</sup>은 화상수에서 변화를 가져오지 않았으나 제안된 고속 정렬 알고리즘보다 약 두 배의 처리시간이 걸렸다. 우리가 제안한 알고리즘으로 처리할 때 소요된 시간은 1.04초였다. 이것은 다른 화상 회전 알고리즘들과 비교할 때 상당히 향상된 처리시간이다. 그리고 문자인식에서 문자형태를 유지하기위해 중요한 요소인 문자 화상을 형성하는 화소 수에 전혀 변화가 없음을 알 수가 있었다. 따라서 제안된 고속 화상 정렬 알고리즘이 OCR 시스템들에 적용될 때 기울어진 화상을 정렬하는데 매우 효과적이다.

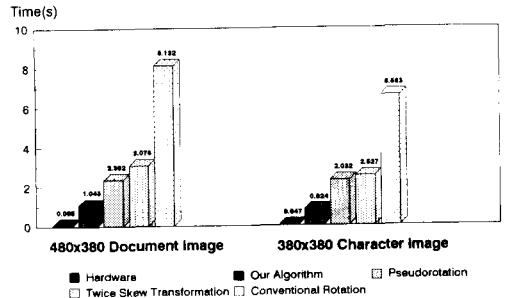


그림 7. 처리시간의 비교

Fig. 7. Compared with processing times.

표 2. 화소 수의 변환비율

Table 2. Changing rates in the number of data pixel.

IMAGES	METHODS	Our algorithm	Pseudo rotation[5]	Twice skew transformation algorithm[4]	Conventional rotation method[3]
480 × 380 sized document image	Changing rate in the # of data pixel	0 %	0 %	0.461 %	0.438 %
	5°	0 %	0 %	1.828 %	0.438 %
	15°	0 %	0 %	4.264 %	3.894 %
384 × 384 sized character image	Changing rate in the # of data pixel	0 %	0 %	0.713 %	0.694 %
	5°	0 %	0 %	3.107 %	2.416 %
	15°	0 %	0 %	7.149 %	5.123 %

情報 전달 방법으로 音聲情報 전달이 보편화되었고 최근에는 컴퓨터 기술과의 통합화 및 綜合情報通信網(ISDN: Integrated Service Digital Networks)의 실현에 따라 音聲 및 畫象情報의 전달이 가능해지고 있다.

(a) 원 화상

情報 전달 방법으로 音聲情報 전달이 보편화되었고 최근에는 컴퓨터 기술과의 통합화 및 綜合情報通信網(ISDN: Integrated Service Digital Networks)의 실현에 따라 音聲 및 畫象情報의 전달이 가능해지고 있다.

(b) 제안된 고속 화상 정렬 알고리즘

情報 전달 방법으로 音聲情報 전달이 보편화되었고 최근에는 컴퓨터 기술과의 통합화 및 綜合情報通信網(ISDN: Integrated Service Digital Networks)의 실현에 따라 音聲 및 畫象情報의 전달이 가능해지고 있다.

(d) Pseudo 회전 알고리즘

情報 전달 방법으로 音聲情報 전달이 보편화되었고 최근에는 컴퓨터 기술과의 통합화 및 綜合情報通信網(ISDN: Integrated Service Digital Networks)의 실현에 따라 音聲 및 畫象情報의 전달이 가능해지고 있다.

(c) 두 비스듬 이동 알고리즘

情報 전달 방법으로 音聲情報 전달이 보편화되었고 최근에는 컴퓨터 기술과의 통합화 및 綜合情報通信網(ISDN: Integrated Service Digital Networks)의 실현에 따라 音聲 및 畫象情報의 전달이 가능해지고 있다.

(e) 전통적인 회전 알고리즘

그림 8. 여러 알고리즘에 의해 정렬된 문서 화상

Fig. 8. Document image aligned by algorithms.

제안된 알고리즘에 기본을 둔 고속 화상정렬 하드웨어로 처리할 때 480×380 크기의 기울어진 화상의 경우 약 65 ms가 소요된다. 소프트웨어로 처리할 때 처리시간이 사각화상의 옆 길이들의 자승에 비례하지만, 하드웨어 처리시간은 단지 사각화상의 옆 길이에 비례한다. 그러므로 논문에 제시한 하드웨어 가속기를 사용한다면 소프트웨어보다 약 16배 빨리 정렬할 수 있고 A4 크기의 화상을 정렬하는데 약 2초가 소요되므로 문서인식 시스템의 처리속도를 크게 향상시킬 수 있다.

參考文獻

[1] Y.S. Kim, W.S. Choi, S.W. Kim, "A high-speed thinning processor for character recognition system", *IEEE Transactions on Consumer Electronics*.

vol. 38, no. 4, pp 762-766, Nov. 1992.  
 [2] 김 민석, 손 한웅, 최 완수, 김 수원, "자소 추출 방법을 이용한 고속 한글 인식 시스템의 구현" 전자공학회 논문지 vol. 29, no. 6, pp. 418-424, 1992.  
 [3] R.C. Gonzalez, and P. Wintz: *Digital image processing*. Addison-Wesley Publishing Company, pp. 79-80, 1987.  
 [4] N. Tsuchida, Y. Yamada, and M. Ueda: "Hardware for image rotation by twice skew transformations", *IEEE Trans. on Acoustics, Speech, and Signal Processing*, pp. 527-531, April 1987.  
 [5] J.E. Volder, : "The CORDIC trigonometric computing technique", *IRE Trans. on Electron. Comput.*, EC-8, pp. 330-334, 1959.

著者紹介

崔完洙(正會員) 第29卷 B編 第6號 參照  
현재 대림 공업 전문대학 전임 강사



崔震虎(準會員)  
1968年 11月 16日生. 1992年 2月  
고려대학교 전자공학과 졸업.(공학사) 1992年 ~ 현재 고려대학교  
전자공학과 대학원 석사과정. 주  
관심분야는 화상 및 음성인식 등  
임.



鄭淵九(正會員)  
1951年 6月 1日生. 1979年 고려  
대학교 전자공학 학사. 1985年 미  
국 클리블랜드 주립대학 전산학  
석사. 컴퓨터비전 전공. 1991年  
미국 웨인주립대학 전산학 박사.  
컴퓨터비전 전공. 1991年 3月 ~  
1993年 현재 고려대학교 객원교수. 주관심분야 머신  
비전, 컴퓨터비전, 패턴인식, 공장자동화, 인공지능,  
데이타베이스. 주요관심분야는 자동시각 검사시스템  
개발, 머신비전을 이용한 양,부 판정시스템 개발, 머  
신비전시스템의 공장자동화 활용연구, 자동공장을 위  
한 인터페이스 및 인테그레이션 연구. 움직이는 물체  
의 트래킹.

金壽遠(正會員) 第27卷 第4號 參照  
현재 고려대학교 전자공학과 부교수