

황재호*, 이대영

(The Study of multi-function processor implementation for binary
image processing)

요 약

본 논문에서는 2진 영상의 평활화(smoothing), 윤곽선 추출(contouring), 세선화(thinning), 특징추출(feature extraction) 등과 같은 처리를 할 수 있는 다기능 프로세서를 설계하였다. 하드웨어 구현에 용이한 3x3 마스크(mask)연산 알고리즘을 룩-업 테이블(look-up table)에 적용하여 256 * 256 까지의 2 차 영상에 대해 고속의 처리를 가능하게 하였다.

1. 서 론

영상 처리는 영상 분석(image analysis), 패턴 인식(pattern recognition, 컴퓨터 비전(computer vision)) 등에서 널리 사용된다. 일반적으로 영상은 A/D 변환기를 통하여 카메라나 스캐너로부터 받아 들여져 영상 메모리에 저장된다. 이러한 영상 메모리에 있는 데이터를 처리하는데 있어서 소프트웨어를 이용하여 처리하고 분석할 수 있다.

소프트웨어적 처리는 프로그램 수정이 간단하고 쉽게 평가할 수 있는 장점이 있다. 그러나 알고리즘의 복잡도와 컴퓨터의 수행속도에 따라 굉장히 많은 시간이 소요되는 단점이 있다. 더구나 영상은 화소의 정보를 포함하는 매트릭스 형태로 구성되기 때문에 굉장히 많은 양의 데이터를 처리해야 한다. 그리고 실시간 처리에 있어서 소프트웨어를 이용한 처리로는 단점이 있

기 때문에 하드웨어의 구현이 절실히 요구된다. 예를 들어 물체 인식을 하는데 있어서 필터링이나 세선화와 같은 전처리 작업은 프로그램을 실행시키는데 상당히 많은 시간이 소요된다. 이러한 처리시간을 단축시키기 위하여 개인용 컴퓨터를 이용하여 전처리를 고속으로 하는 프로세서의 개발이 필요하다.

본 시스템은 2 차 영상에 있어서 평활화(smoothing), 세선화(thinning), 끝점의 추출(end points extracting), 끝점의 라벨링(end points labeling), 분기점 추출(fork points extracting), 분기점 라벨링(fork points labeling) 등 최대 16가지의 전처리 과정을 수행할 수 있다. 64 * 64, 128 * 128, 256 * 256 해상도의 2 차 영상 처리를 가능하게 한다.

II. 3 * 3 마스크 연산

3 * 3 마스크는 문자, 지문 등에 있어서 영상 데이터의 골격 패턴을 추출하는 세선화와 같은 전처리에 사용되는데 3 * 3 윈도우가 그림 1에 나타난다.

P9 (i-1, j-1)	P2 (i-1, j)	P3 (i-1, j+1)
P8 (i, j-1)	P1 (i, j)	P4 (i, j+1)
P7 (i+1, j-1)	P6 (i+1, j)	P5 (i+1, j+1)

그림 1. 3 * 3 윈도우

III. 시스템 설계

그림 2는 인터페이스부(interface unit), 제어부(control unit), 프로세서부(processing unit)로 구성된 전체 시스템을 나타낸다.

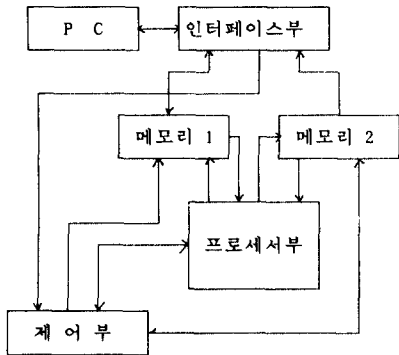


그림 2. 전체 시스템 구성

1. Interface

일반적으로 데이터 통신은 RS-232와 같은 protocol을 이용하여 컴퓨터와 interface한다. 하지만 본 시스템에서는 속도를 향상시키기 위하여 통신 protocol을 필요로 하지 않는 IO device를 구현함으로써 processor에 관련된 모든 메모리의 정보를 직접 access 함으로써 컴퓨터의 main memory와 같이 이러한 메모리를 사용할 수 있게 하기 위함이다. 그림 3은 Memory/IO Interface를 나타낸다.

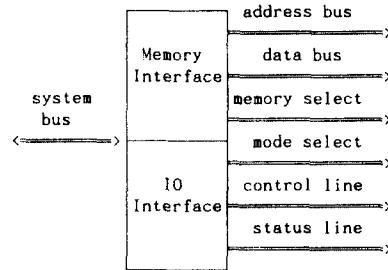
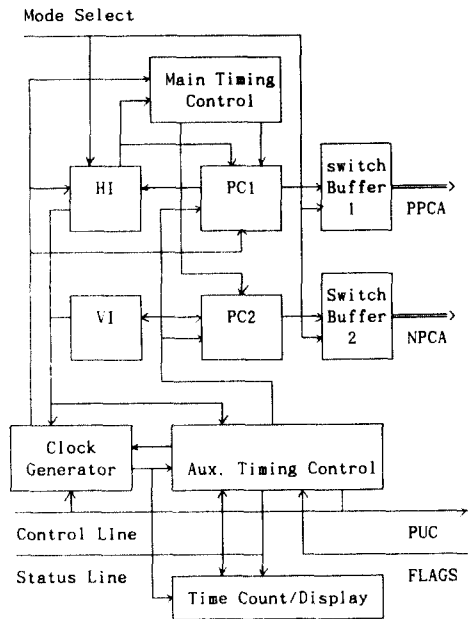


그림 3. M/IO interface

2. 제어부 (Control unit)

이 제어부는 프로세서를 제어하는 중요한 부분이다. 그림 4는 제어부의 구성도를 나타낸다.



- PPCA : Processed Pixel Counter Address
 - NPCA : Neighboring Pixel Counter Address
 - CBAC : Cross Boundary Arbitral Control
 - PUC : Processing Unit Control
- 그림 4. 제어부의 구성도

가) Processed Pixel Counter (Pixel Counter 1)

수평축을 따라 왼쪽에서 오른쪽으로 카운터하는 Xp 카운터와 하나의 수평축이 끝났을 때 위에서 아래로 카운트하는 Yp가 있다. 영상 메모리에서 처리되는

화소의 위치를 번지 지정하는데 사용되는 PPCA(Processed pixel Counter Address)는 SB1(Switching Buffer 1)을 통해 X_p 와 Y_p 로 부터 매핑된다.

나) Neighboring pixel counter (Pixel Counter 2)

처리되는 화소의 이웃들의 위치를 카운터하는데 사용되고 X_n 카운터와 Y_n 카운터가 있다. 영상 메모리에서 처리되는 화소의 이웃들의 위치를 번지 지정하는데 사용되는 NPCA(Neighboring Pixel Counter Address)는 SB2(Switching Buffer 2)을 통하여 X_n 과 Y_n 으로부터 매핑된다.

다) Horizontal Initial (HI)

line scan의 끝에서 X_p 의 값이 0 인지 아닌지를 감지한다.

라) Vertical Initial (VI)

영상의 끝에서 Y_p 의 값이 0 인지 아닌지를 감지한다. Vertical Initial 이 일어날때 Aux Timing Control 이 시작된다.

마) Main Timing Control

영상 메모리에 있는 화소의 access를 제어하는 부분이고 3×3 mask 에 기초하기 때문에 각 화소의 결과는 9 개의 화소가 shift register 에 저장된 후에 결정된다.

바) Aux Timing Control

Vertical Initial(VI)이 영상 끝(부반복의 끝)에서 일어날 때 Aux Timing Control 이 시작된다. flag 가 clear 된 상태로 남아 있으면 세선화 처리가 끝났음을 host 컴퓨터에게 알린다.

3. 프로세서부 (Processing Unit)

그림 5 에는 프로세서부의 구성도를 나타낸다.

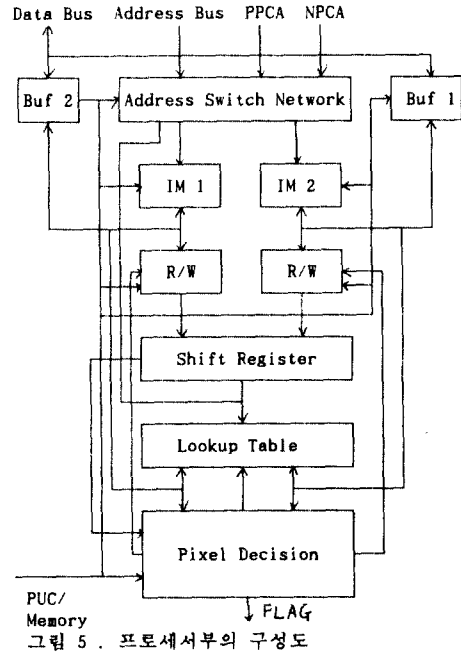


그림 5. 프로세서부의 구성도

가) Address Switching Network(ASN)

Address Switching Network 는 어떤 번지가 메모리(IM 1 또는 IM 2) 와 lookup table에 연결되는지를 결정하는데 그림 4 에 있는 PUC/Memory Select 에 의해 동작한다.

나) Pixel Decision

이 회로는 테이블이 참조된 후에 처리되는 화소가 1 에서 0 으로 바뀐지 아닌지를 결정한다. 화소가 바뀐다면 flag 가 set 되고 결과 0 이 PPCA에 의해 결정된 영상에 의해 저장된다. 그리고 부반복이 끝날 때 프로세서의 flag는 제어부로 message를 보낸다. 그리고 나서 제어부는 flag 의 값에 따라 프로세서가 다음 부반복을 수행 할지 아닌지를 결정한다.

V. 모의 실험 및 고찰

모의 실험은 핸드 스케너를 통하여 획득한 128×1

28 지문, 64 * 64 문자 데이터에 대하여 IBM AT 상에서 어셈블러로 수행하였다. 그림 6 은 본 논문에서 사용한 세선화 알고리즘의 첫번째 부분복에서의 특연 테이블을 나타낸다.

		Lower address bits: A ₃ A ₂ A ₁ A ₀															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		1	1	1	0	1	0	0	0	1	1	1	1	0	0	0	0
1		1	1	1	0	1	0	1	0	1	1	1	0	1	0	1	
2		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
3		0	1	1	0	1	0	1	0	1	1	1	0	1	0	1	
4		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
5		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
6		0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
7		0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
8		1	0	1	0	1	0	1	0	1	1	1	1	0	1	0	
9		1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	
A		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
B		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
C		0	0	1	0	1	1	0	1	1	1	1	1	1	1	0	
D		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
E		0	0	1	0	1	1	0	1	1	1	1	1	1	1	0	
F		0	0	1	0	1	1	1	0	0	1	0	1	1	1	1	

Higher address bits: A₇A₆A₅A₄

그림 6. 세선화 첫번째 부분복의 특연 테이블

VI. 결 론

본 시스템에는 평활화, 윤곽 추출, 세선화, 특징 추출 등과 같은 처리에 있어 어떠한 3 * 3 mask 알고리즘을 사용할 수 있다. 그리고 일반적으로 프로세서와 host 컴퓨터와의 데이터 통신에 있어서 RS-232 와 같은 protocol을 사용하지만 본 시스템에서는 이러한 프로토콜을 필요로 하지 않기 때문에 host 컴퓨터의 CPU 는 직접 프로세서와 관련된 모든 메모리를 access 할 수 있다.

본 시스템에 multi-processor 를 사용하고 pipeline 개념을 도입하여 processing cycle 의 time clock 를 줄일 수 있고 현 VLSI 기술을 도입하여 clock rate 를 10 MHz 이상으로 높여 성능을 향상시킬 수 있다. 더 나아가서 이러한 multi-unit 프로세서를 array 로 구성하여 설계하면 훨씬 더 우수한 결과를 얻을 것으로 기대한다.

참 고 문 헌

- [1] Zhang, T.Y. and Suen, C.Y. A fast Parallel Thinning Algorithm.
- [2] Chen & Hsu, A Modified Thinning Algorithm.
- [3] Yung-Sueng Chen & Wen-Hsing Hsu, A Multi-function Parallel Processor For Binary Image Processing.
- [4] 허 운석, 영상의 고속 세선화 장치 구현에 관한 연구.
- [5] Lewis C. Eggebrecht, Interfacing to The IBM Personal Computer.
- [6] Walter A. Triebel, The 8088 Microprocessor.
- [7] Kal Hwang, Computer Architecture and Parallel Processing.
- [8] Harry Garland, Introduction to Microprocessor System Design.
- [9] Gonzalles, Digital Processing.
- [10] Stanley R. Sternberg, Pipeline Architectures for Image Processing.
- [11] Siegel L.J., Siegel H.J., Swain P.H., Parallel Algorithm Performance Measures.
- [12] Jagadish H.V. et al, A survey of Pipelining in Computing Arrays.
- [13] Holt C.M. et al, An improved Parallel thinning Algorithm.
- [14] Chu Y.K., Suen C.Y., An alternate Smoothing and Stripping Algorithm for Thinning Digital Patterns.
- [15] Rosenfield, A., Kak, A.C., Digital Picture processing.