

메쉬 구조형 SIMD 컴퓨터 상에서 신축적인 병렬 레이블링 알고리즘

박은진*, 이갑섭**, 성효경*, 최홍문*

*경북대학교 전자전기공학부, **구미전문대학 전자 계산과

A Scalable Parallel Labeling Algorithm on Mesh Connected SIMD Computers

Eun-Jin Park*, Gab-Seob Lee**, Hyo-Kyung Sung*, Heung-Moon Choi*

*The School of Electronics and Electrical Engineering, Kyungpook National University

**Department of Computer Science, Kumi College

Abstract

A scalable parallel algorithm is proposed for efficient image component labeling with local operators on a mesh connected SIMD computer. In contrast to the conventional parallel labeling algorithms, where a single pixel is assigned to each PE, the algorithm presented here is scalable and can assign $m \times m$ pixel set to each PE according to the input image size. The assigned pixel set is converted to a single pixel that has representative value, and the amount of the required memory and processing time can be highly reduced. For $N \times N$ image, if $m \times m$ pixel set is assigned to each PE of $P \times P$ mesh, where $P = N/m$, the time complexity due to the communication of each PE and the computation complexity are reduced to $O(P \log P)$ bit operations and $O(P)$ bit operations, respectively, which is $1/m$ of each of the conventional method. This method also diminishes the amount of memory in each PE to $O(P)$, and can decrease the number of PE to $O(P^2) = \Theta(N^2/m^2)$ as compared to $O(N^2)$ of conventional method. Because the proposed parallel labeling algorithm is scalable, we can adapt to the increase of image size without the hardware change of the given mesh connected SIMD computer.

1. 서론

영상 분석이나 컴퓨터 비전 등에서는 비슷한 성질을 갖는 영상내의 영역들을 독립적으로 추출하고, 단순 반복 처리에 의해 영상을 레이블링하는 연구가 많이 진행되고 있다. 그러나 영상 레이블링은 그 반복적인 처리 구조로 인하여 처리 시간이 비교적 많이 소요되는 것으

로 알려져 있다[1, 5]. 이러한 처리 시간을 줄이기 위하여 다양한 병렬 모델을 이용하여 레이블링을 병렬 처리하고 있다. 여러 가지 병렬 처리 모델 중에서 내부 연결 구조가 영상내의 화소들간의 공간적 관계를 잘 반영하는 규칙적 형태이며, 적은 비용으로도 구현이 용이한 메쉬(mesh) 구조의 SIMD(single instruction stream and multiple data stream)형 컴퓨터에 대한 연구가 많이 진행되어 왔다[2, 5].

Cypher 등[3]은 병렬 영상 레이블링 과정의 시간 복잡도(time complexity)에 있어서 곱셈 상수를 줄이기 위하여 $N \times N$ 크기의 영상에 대하여 Levaldi[6]가 제안한 축소방법(shrinking method)과 레이블 전파 방법을 사용하여 $O(\log N)$ 비트 연산의 시간 복잡도와 각 PE(processing element)당 $O(N)$ 비트의 메모리를 필요로 하는 알고리즘을 제안하였다. Alnuweiri 등[4]은 1에서 $\log(2N)$ 사이의 값을 갖는 임의의 정수 인수 k 에 따라 PE당 메모리 요구량이 $O(kN^{1/k})$ 비트이고 시간 복잡도가 $O(kN)$ 비트 연산인 병렬 레이블링 알고리즘을 제안하고, k 값을 조절함으로써 시간 복잡도와 PE당 메모리량 사이에 타협점을 설정할 수 있도록 하였다.

그러나 친숙한 병렬 레이블링 방법들은 $N \times N$ 크기의 영상에 대하여 $N \times N$ 배열의 PE에 화소를 일 대 일로 할당할 후 축소 알고리즘을 적용하기 때문에 레이블링할 영상의 크기에 따라 PE의 수와 각 PE에서 필요로 하는 메모리의 양을 미리 충분히 확보해야 한다. 그러나 경우에 따라서는 사용되지 않는 PE와 메모리의 낭비를 가져올 수 있으며, 이를 방지하기 위해서는 영상의 크기에 맞도록 시스템을 다시 구현해야 한다. 따라서 PE와 메모리의 낭비를 없앨 수 있도록 이미 구현된 메쉬형 컴퓨터 상에서 다양한 크기의 영상을 레이블링할 수 있는 신축적인 병렬처리 방안이 필요하다.

본 논문에서는 메쉬형 SIMD 컴퓨터 상에서 국부 연산자(local operators)를 이용하여 영상의 크기에 따라

신속적인 병렬 레이블링 알고리즘(scalable parallel labeling algorithm)을 제안하였다. 각 PE당 하나의 화소를 할당하여 처리하던 기존의 병렬 레이블링 알고리즘과는 달리 이 방법에서는 영상의 크기에 따라 $m \times m$ 개의 화소 집합을 각 PE에 할당하고, 각 PE는 이를 $1/m^2$ 로 축소시켜 하나의 화소로 대표하여 처리함으로써 처리 시간 및 메모리 요구량을 대폭 줄일 수 있도록 하였다. $N \times N$ 크기의 영상에 대하여 P^2 개의 PE로 구성된 $P \times P$ 메쉬 컴퓨터 상에서 제안된 알고리즘의 각 PE간의 통신으로 인한 통신 복잡도 및 계산 복잡도 측면에서 실험하고 고찰하였다.

II. 국부 연산자를 이용한 레이블링

영상의 레이블링을 위해 본 논문에서는 그림 1과 같은 $N \times N$ 메쉬형의 SIMD 컴퓨터를 가정하고, 각 통신 링크(links)는 1비트 폭으로 가정하였다. 메쉬상에서 i 번째 행, j 번째 열에 있는 프로세서를 PE_{ij} 로 나타내고, 메쉬의 가장 좌상단에 있는 PE를 $PE_{1,1}$ 로 한다. 먼저 각 (i, j) 화소를 각 PE_{ij} 에 할당하여 주어진 $N \times N$ 영상이 $N \times N$ 메쉬에 저장된다고 가정한다. 메쉬상에서 영상을 레이블링하기 위한 국부 연산으로는 병렬 축소와 레이블 전파를 사용하였다.

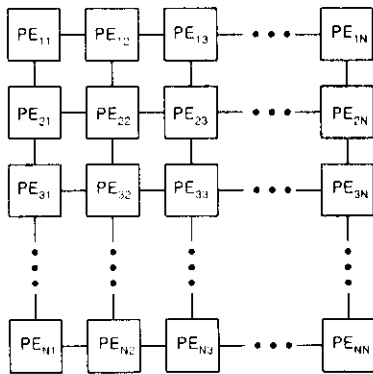


그림 1. 메쉬형 SIMD 컴퓨터
Fig. 1. Mesh connected SIMD computer.

1. 병렬 축소 알고리즘

Leviadi가 고안한 이진 형태학적 연산(binary morphological operation)을 반복적으로 적용하는 병렬 축소 알고리즘에서는 연산동안 (i, j) 화소의 값은 (i, j) , $(i, j-1)$, $(i-1, j)$, $(i-1, j-1)$ 화소들의 이진 값에 의해 정해진다. 즉, 각 화소의 값은 그 화소의 2×2 이웃 화소에 의해 결정되며, b_{ij} 를 화소 (i, j) 의 이진값이라 두면, 화소 (i, j) 의 새로운 값 V_n 은

$$V_n = h(b_{ij} + b_{i,j-1} + b_{i-1,j} - 1) + (h(b_{ij} + b_{i-1,j-1} - 1)) \quad (1)$$

에 의해 정의된다. 여기서, $h(t)$ 는 $t \leq 0$ 에 대해 $h(t)=0$, $t > 0$ 에 대해서는 $h(t)=1$ 로 정의되는 Heaviside 함수이다. (i, j) 화소의 값이 '1'일 때, 그 화소와 이웃하는 세 화소 중 하나 이상의 화소값이 '1'이면 축소연산 후의 (i, j) 화소의 값은 '1'이 된다. 그리고 (i, j) 화소의 값이 '0'일 때는 (i, j) 화소의 왼쪽 화소와 위쪽 화소가 둘 다 '1'의 값을 가질 경우에만 축소 연산 후의 (i, j) 화소 값이 '1'이 된다.

Heaviside 함수로 정의되는 국부 연산자는 영상내의 각 영역에 대해 하나의 화소가 남을 때까지 그 영역들을 각각 둘러싸는 가장 작은 사각형의 오른쪽 아래 부분으로 압축시키는 효과를 나타낸다. 영상내의 하나의 영역이 단지 하나의 화소만이 남을 때까지 축소시키는데 요구되는 축소 연산의 횟수는 그 영역을 둘러싸는 가장 작은 사각형의 오른쪽 아래 화소로부터 그 영역에 속하는 화소들 중 가장 먼 화소까지의 Manhattan 거리와 같다. 그러므로 최대 $2N$ 번의 축소 연산 후에는 모든 연결 성분들이 사라지게 된다. 축소 알고리즘은 영상내의 '1'값을 가진 화소가 모두 사라질 때까지 Leviadi의 축소 연산을 반복적으로 적용하는데, 축소 연산을 할 때마다 새로운 영상이 얻어지고 이 결과들을 PE의 메모리에 저장시킨다.

2. 레이블 전파

원 영상에 축소 연산을 y 번 적용한 후의 결과를 부분 결과 y 라고 한다면 부분 결과 y 의 값은 PE의 메모리 위치 y 에 저장되어 있으므로 레이블 전파 과정에서는 마지막 축소 연산의 결과값에서부터 역순으로 부분 결과값들에 따라 각 화소에 레이블을 할당한다. 축소 연산을 수행하는 중에 여러 개의 영역들이 같은 위치의 한 화소로 축소될 수도 있는데, 이는 같은 단계에서 일어날 수 없으므로 부분 결과 y 를 포함한 (i, j, y) 값을 그 영역의 레이블로 함으로써 같은 화소로 축소된 다른 영역들도 각각 다른 레이블을 갖게 된다. $2N$ 번째 얻어진 영상에서 첫 번째 영상으로 진행되는 레이블 전파의 y 단계에서 각 PE_{ij} 는 자신들의 $(y-1)$ 번째 메모리 위치의 값이 '1'이라면 PE_{ij} , $PE_{i,j+1}$, $PE_{i+1,j}$, $PE_{i+1,j+1}$ 중에서 그들의 y 번째 메모리 위치값이 '1'인 PE로부터 레이블을 전송 받는다. 만약 $(y-1)$ 번째 메모리 위치값이 '1'인 PE_{ij} 로 어떤 레이블도 전송되지 않는다면 해당 PE_{ij} 는 자신의 (i, j, y) 값인 새로운 레이블을 가지게 된다. 이렇게 y 를 $2N$ 에서 1까지 역순으로 레이블을 전파함으로써 각 연결 성분들은 유일한 레이블을 갖게 된다.

국부 연산자를 이용한 레이블링 알고리즘의 축소 연산과 레이블 전파 과정 동안의 통신 연산의 수는 통신 단계의 수 c 와 각 단계의 쉬프트(shift)수 s , 그리고 각 쉬프트당 전송되는 비트 수 b 의 곱과 같다. 축소 연산과 레이블 전파의 두 과정 모두 $c=2N$ 이고 $s=3$ 으로 같지만, 축소 연산에서는 $b=1$ 이고 레이블 전파 과정에서

는 레이블이 (i, j, y) 로 이루어지므로

$$b = 3\log N + 1 \quad (2)$$

이 된다. 그러므로 전체 통신 연산의 수는

$$T_{comm} = 18M\log N + 12N \quad (3)$$

이 되며, 각 PE가 필요로 하는 메모리의 양 S 는 최대 $2N$ 번의 축소 연산의 부분 결과들과 i, j, y 값을 나타낼 수 있어야 하므로

$$S = 2N + 3\log N + 1 \quad (4)$$

비트가 된다.

III. 신속적인 병렬 레이블링 알고리즘의 제안

제안한 방법은 매쉬 내의 각 PE당 한 개의 화소를 할당하여 처리하던 기존의 방법과는 달리 각 PE에 $m \times m$ 개의 화소 집합을 할당함으로써 더 큰 입력 영상에 대해서도 신속적으로 레이블링이 가능하도록 하였다. 그림 2에서 보는 바와 같이, 입력 영상의 크기에 따라 각 PE에 $m \times m$ 개의 화소 집합을 할당하면, 각 PE는 자신에게 할당된 화소들에 대해 논리곱 연산을 수행함과 동시에 플래그(flag) 처리 연산을 한다. 논리곱 연산은 할당된 $m \times m$ 개의 화소가 모두 '1'일 경우에만 '1'이 되고 그 외의 경우는 모두 '0'이 되므로 결국 각 PE는 하나의 화소값만을 가지게 되며 이 연산은 일종의 수리형 태학적 부식(erosion)과정으로 볼 수 있다.

이렇게 하면 각 연결 성분의 가장자리의 화소를 할당 받은 PE들은 '0'을 대표값으로 가지게 되지만 이 화소들 중에는 연결 성분에 속하는 화소들이 포함되어 있으므로 이들 화소에도 레이블을 할당해 주어야 한다. 이는 해당 PE의 플래그 비트를 세트(set)시켜 줌으로써 해결할 수 있다. 이 플래그 비트는 각 연결 성분의 가장자리 화소들을 할당받은 PE들에서만 세트되므로 레이블 전파시 상·하·좌·우로 네 번의 추가적인 레이블 전파 연산을 수행해 주어야 한다.

논리곱 연산과 플래그 처리 연산이 끝나면 각 PE는 할당받은 $m \times m$ 개의 화소 집합을 저장하고, 연산의 결과값인 대표값을 가지고 축소 연산을 수행한다. 이 축소 연산은 각 PE당 하나의 대표값만 가지고 수행하므로 Levaldi가 제안한 방법과 동일하다. 축소 연산의 부분 결과 y 가 모두 '0'이 될 때까지 이 연산을 반복 수행하고, 축소 연산이 끝나면 전술한 방법과 같은 방법으로 레이블을 전파하는데 레이블 전파의 반복 횟수는 앞에서 제안한 방법보다 네 번 증가한다. 이는 플래그 비트가 '1'인 PE 즉, 각 연결 성분의 가장자리 화소를 할당받은 PE에게도 레이블을 전파해 주어야 하기 때문이다.

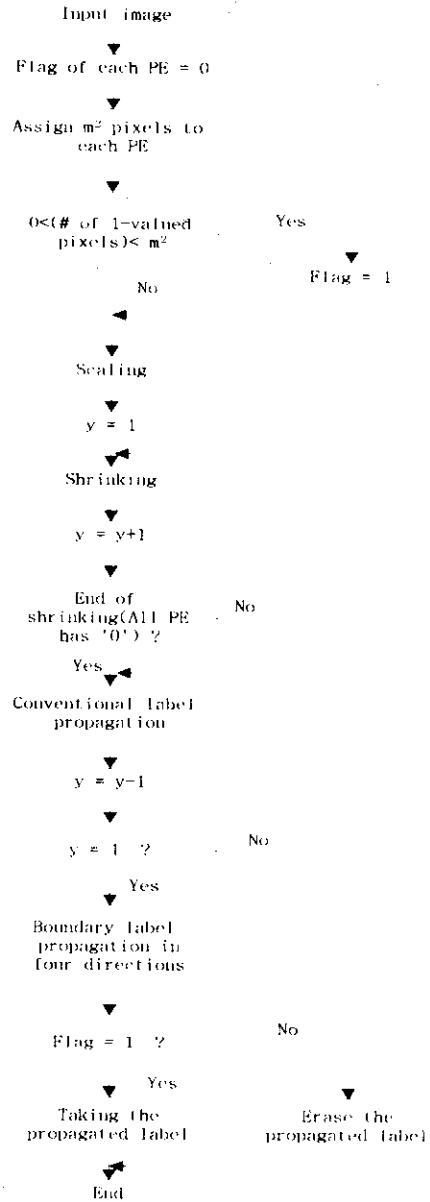


그림 2. 제안한 신속적인 병렬 레이블링 알고리즘의 흐름도

Fig. 2. Flow chart of the proposed scalable parallel labeling algorithm.

이 네 번의 레이블 전파 역시 수리형태학적 팽창(dilation) 과정으로 볼 수 있다. 이러한 레이블 전파 연산이 완전히 끝나면 각 연결 성분에 속하는 화소를 할당 받은 PE들은 모두 하나의 레이블을 가지게 된다. 그러므로 레이블을 갖고 있는 PE에 할당되었던 모든 화소

들은 그 PE가 가진 레이블을 자신의 레이블로 가지게 된다.

제한한 방법에서는 m 의 값을 조정함으로써 기존의 방법으로 구성된 메쉬 컴퓨터 상에서 레이블링할 수 있는 영상과 비교해 볼 때 m^2 배만큼 큰 영상까지 처리할 수 있다. 바꿔 말하면 같은 크기의 영상을 처리할 경우 제한한 방법은 기존 방법에 비해 영상의 크기를 $1/m^2$ 로 줄이는 것과 같은 신축성을 가지므로 처리시간 및 메모리 요구량을 모두 $1/m$ 로 줄일 수 있고 레이블링을 위해 실제로 사용되는 PE의 수도 $1/m^2$ 만큼 줄일 수 있다.

IV. 시뮬레이션 결과 및 고찰

기존 방법과 제한한 방법의 성능을 비교하기 위해 64×64 크기의 메쉬 컴퓨터 상에서 입력 영상의 크기를 변경하고, 또 각 PE에 할당하는 화소의 수를 변화시키면서 실험하였다. 기존의 알고리즘을 실험한 후, 사용한 PE 구조에 논리곱 연산 및 풀랙 처리 연산부를 첨가하여 각기 다른 크기의 영상에 대해 m 의 값을 바꾸어 가면서 실험하고 그 결과를 표 1과 표 2에 나타내었다.

표 1. 영상의 크기 변화에 따른 통신 복잡도의 비교
Table 1. Communication complexity comparison for each image size. [Unit time]

Image size	Conventional algorithm		Proposed algorithm	
	m	1	2	4
64×64		7680	3268	1348
128×128		-	7684	3268
256×256		-	-	7684

표 2. 영상의 크기 변화에 따른 계산 복잡도의 비교
Table 2. Computation complexity comparison for each image size. [Unit time]

Image size	Conventional algorithm		Proposed algorithm	
	m	1	2	4
64×64		256	130	66
128×128		-	258	130
256×256		-	-	258

표 1 및 표 2에서 보는 바와 같이 64×64 메쉬 컴퓨터 상에서 기존 방법[3]으로는 128×128 및 256×256 크기의 영상을 레이블링 할 수 없으나 제한한 방법에서는 128×128 및 256×256 크기의 영상에 대해서도 신축적으로 레이블링을 할 수 있다. 또한 통신 복잡도와 계산 복잡도를 신축 인수 m 에 반비례하도록 줄일 수 있다.

V. 결론

본 논문에서는 메쉬형 SIMD 컴퓨터 상에서 국부 연산자를 이용하여 영상의 크기에 따라 신축적인 병렬 레이블링 알고리즘을 제안하였다. 제안한 알고리즘에서는 영상의 크기에 따라 $m \times m$ 개의 화소 집합을 각 PE에 할당하여 하나의 화소로 대표하여 처리함으로써 $1/m^2$ 의 영상 축소 효과를 얻을 수 있으므로 처리 시간과 각 PE가 필요로 하는 메모리의 양을 대폭 줄일 수 있도록 하였다. 시뮬레이션 결과 $N \times N$ 크기의 영상에 대하여 $P \times P$ 배열의 메쉬형 컴퓨터 상에서 각 PE당 $m \times m$ 개의 화소들을 할당했을 때 PE들간의 통신 복잡도와 PE내의 계산 복잡도는 각각 $O(P \log P)$ 및 $O(P)$ 로서 기존 방법들에 비해 $1/m$ 만큼 줄었으며, 각 PE당 필요한 메모리 역시 $O(P)$ 로 줄일 수 있었다. 그리고 실제 연산을 하기 위해 필요한 PE의 수도 기존의 $O(N^2)$ 에서 $O(P^2) = O(N^2/m^2)$ 로 줄일 수 있었다. 또한 입력 영상이 커질 경우 신축 인수 m 의 값을 증가시킴으로써 P 를 일정하게 유지시킬 수 있으므로 메모리의 양이나 PE 개수의 변화 없이 여러 크기의 영상을 신축적으로 레이블 할 수 있다.

참고문헌

- [1] S. M. Smith and J. M. Brady, "SUSAN - a new approach to low level processing," *International Journal of Computer Vision*, vol. 23, no. 1, pp. 45-78, May 1997.
- [2] H. Shi and G. X. Ritter, "A fast algorithm for image component labeling with local operators on mesh connected computers," *Journal of Parallel and Distributed Computing*, 23, pp. 455-461, 1994.
- [3] R. E. Cypher, J. L. C. Sanz, and L. Snyder, "Algorithms for image component labeling on SIMD mesh-connected computers," *IEEE Trans. Comput.*, vol. 39, no. 2, pp. 276-281, Feb. 1990.
- [4] H. M. Alnuweiri and V. K. Kumar, "Fast image labeling using local operators on mesh-connected computers," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, no. 2, pp. 202-207, Feb. 1991.
- [5] H. M. Alnuweiri and V. K. Kumar, "Parallel architectures and algorithms for image component labeling," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 14, no. 10, pp. 1014-1034, Oct. 1992.
- [6] S. Levialdi, "On shrinking binary picture patterns," *Commun. ACM*, vol. 15, no. 1, pp. 7-10, Jan. 1972.