

논문 2004-41SP-6-3

# NTGST 병렬화를 이용한 고해상도 BLU 검사의 고속화

## (NTGST-Based Parallel Computer Vision Inspection for High Resolution BLU)

김복만\*, 서경석\*, 최흥문\*

(Bok-Man Kim, Kyung-Seok Seo, and Heung-Moon Choi)

### 요약

본 논문에서는 LCD (liquid crystal display) 생산라인에서 컴퓨터 비전에 의한 BLU (back light unit)의 고해상도 정밀검사를 원활하게 하기 위해 SIMD (single instruction stream and multiple data stream)형 병렬 구조의 다중 프로세서를 이용하여 계산 집약적인 NTGST (noise-tolerant generalized symmetry transform) 검사 알고리즘을 병렬구현 하였다. 먼저 알고리즘 자체의 속도향상을 위해 C 코드의 최적화를 거친 후, 순차형 프로그램을 N개의 데이터를 동시에 처리하는 SIMD형 언어로 변환하고, 검사영상 데이터를 SIMD형 다중프로세서에서 P개의 각 쓰레드에 분할 할당함으로써  $O(NP)$ 의 속도향상이 가능하도록 하였다. Dual Pentium III 프로세서를 사용하여 실험한 결과, 제안한 병렬시스템은 기존보다  $Sp=8$  배 이상 고속 처리가 가능하며, 다양한 크기의 BLU에 대한 고해상도 정밀검사장비에도 신속적으로 확장적용 가능성을 확인하였다.

### Abstract

A novel fast parallel NTGST is proposed for high resolution computer vision inspection of the BLUs in a LCD production line. The conventional computation-intensive NTGST algorithm is modified and its C codes are optimized into fast NTGST to be adapted to the SIMD parallel architecture. And then, the input inspection image is partitioned and allocated to each of the P processors in multi-threaded implementation, and the NTGST is executed on SIMD architecture of N data items simultaneously in each thread. Thus, the proposed inspection system can achieve the speedup of  $O(NP)$ . Experiments using Dual Pentium III processor with its MMX and extended MMX SIMD technology show that the proposed parallel NTGST is about  $Sp=8$  times faster than the conventional NTGST, which shows the scalability of the proposed system implementation for the fast, high resolution computer vision inspection of the various sized BLUs in LCD production lines.

**Keywords:** 고해상도 BLU 검사, NTGST, 병렬 고속화, SIMD

### I. 서론

최근, 영상 출력장치 LCD의 수요는 대형 고해상도 LCD TV, LCD 모니터, 노트북 컴퓨터, PDA (personal digital assistant) 및 기타 휴대장치 등의 생산 증가에 힘입어 대폭적인 증가 추세에 있다. BLU는 LCD 뒷면을 비춰주는 발광부품으로서 여러 층으로 이루어진 매우 정교한 핵심 부품이며, 이의 불량은 LCD의 불량률의 주요 원인이다. 이러한 BLU 불량을 효과적으로 검출하

기 위하여 잡음에 강건한 주목연산자인 NTGST 알고리즘을 이용한 검출방법이 활용되고 있다<sup>[1-4]</sup>.

기존의 검사 장비에서 15.2 인치 LCD의 BLU를 자동 검사할 때 약 2048 화소의 라인스캐너로 입력받은 영상을 사용하여  $200\mu\text{m}$  지름의 흑백 얼룩들을 20초에 한 장씩 검출할 수 있었다. 그러나 이런 정도 크기의 흑백 얼룩뿐 아니라  $10\mu\text{m}$  폭의 아주 미세한 흠집 (scratch)까지도 검출해야 하는 정밀 검사가 요구되고 있으며, 이를 만족시키기 위해서는 8192 화소의 고해상도 라인스캐너를 사용하여 입력받은 영상을 사용하여야만 검사가 가능하다. 이때 입력되는 영상의 크기는 약 50MB의 데이터 용량을 가지며, 기존의 NTGST를 이 50MB의 검사 영상 데이터에 적용하면 약 130초가 소요된다. 실제

\* 정회원, 경북대학교 전자전기컴퓨터학부  
(School of Electrical Engineering and Computer Science, Kyungpook National University)  
접수일자: 2002년8월26일, 수정완료일: 2004년10월29일

LCD 생산라인에서 사용하고 있는 자동검사 장비에서는 BLU 한 장당 대략 20초 이내의 검사시간을 요구하고 있으므로 기존 순차형 NTGST 알고리즘으로는 이를 해결할 수가 없다. 최근 고해상도 LCD TV를 위한 30~60 인치의 대형 LCD가 생산됨에 따라 이런 검사 속도 문제들은 더욱 크게 대두되고 있다. 따라서 고속화에 대한 연구가 필요하며, 순차알고리즘 자체의 고속화만으로는 소프트웨어 및 하드웨어 장비 가격이 너무 고가이어서 다중 병렬 프로세서를 이용한 병렬고속화에 관한 연구가 절실히 요구되고 있다.

본 논문에서는 BLU 검사에 사용되는 NTGST 알고리즘을 SIMD 구조의 다중프로세서 환경에 맞게 병렬 고속화 하였다. 일반적으로 PC에 사용되는 범용 프로세서들은 최근 SIMD 구조를 선택하고 있고, Intel Pentium III 프로세서 이상에서는 MMX<sup>[5]</sup> 및 SSE<sup>[6]</sup> (streaming SIMD extensions)를 지원하므로 이를 이용하여  $N$ 개의 데이터를 한 번에 병렬처리 할 수 있고, 또한 NTGST 연산에서 화소나 영역 간에는 서로 독립인 점을 이용하여 처리대상 영상을 공간 분할하여  $P$ 개의 다중프로세서에서 다중쓰레드 (multi-thread)로 처리하도록 함으로써  $O(NP)$ 의 속도향상을 얻을 수 있도록 하였다. 실제의 BLU 검사 실험에 Dual Pentium III 1GHz 프로세서를 사용하여 평균적으로  $S_p = 8$ 배 이상의 고속 처리가 가능하였으며, BLU의 크기에 따라 다중프로세서를 사용하여 신속적으로 병렬화하면 다양한 크기 BLU의 고해상도 검사 장비의 요구 사양도 맞춰나갈 수 있음을 확인하였다.

II. 순차 NTGST 알고리즘

NTGST는 두 화소의 거리가 가까울수록, 명도변화 방향이 대칭일수록, 명도 변화 크기가 클수록 큰 대칭 기여도를 나타내며, 기존 GST<sup>[7]</sup>에서와는 달리 두 화소의 명도변화 방향이 수렴하면 음극성을 가지나 발산하면 양극성을 가진다. 이를 이용하여 복잡한 배경을 가지는 객체 영상에 적용하는 경우 복잡 배경 및 불규칙 잡음에 대한 대칭기여도는 서로 반대 극성으로 상쇄되고 대칭성이 강한 객체에 의한 대칭기여도 만을 누적하여 복잡 배경에서도 객체만을 검출 할 수 있다<sup>[3]</sup>.

그림 1에서와 같은 순차 NTGST 알고리즘에서는 입력 영상의 마스크내의 모든 화소들에 대해서 S.1에서 S.15까지의 15단계를 거쳐서 대칭도 맵을 얻는다.

Step	NTGST Algorithm
	for $k:=0$ to $N_{max}$ do for $l:=0$ to $N_{max}$ do for $m:=N_{max}$ to $0$ do for $n:=N_{max}$ to $0$ do begin
S. 1	$p_i := \hat{A}(x_k, y_l);$
S. 2	$p_j := \hat{A}(x_m, y_n);$
S. 3	find $g_x$ (*apply gradient operator*)
S. 4	find $g_y$
S. 5	$g := \text{magnitude}[g_x, g_y];$ (*look-up magnitude*)
S. 6	$\theta_{eta_x} := \arctan(g_y/g_x);$
S. 7	$\theta_{eta_y} := \arctan(g_y/g_x);$
S. 8	$\alpha_{pha_y} := \arctan\left(\frac{y_n - y_l}{x_m - x_k}\right);$
S. 9	$DWF_{sigm}(i, j) = \frac{1}{\sqrt{2 * pi * sigma}} \exp\left(-\frac{\ p_i - p_j\ }{2 * sigma}\right);$ (*DWF: Distance Weight Function*)
S. 10	$PWF(i, j) = \sin\left(\frac{\theta_{eta_x} + \theta_{eta_y}}{2} - \alpha_{pha_y}\right) * \sin\left(\frac{\theta_{eta_x} - \theta_{eta_y}}{2}\right);$ (*PWF: Phase Weight Function*)
S. 11	$\gamma_i := \hat{g}(x_k, y_l);$
S. 12	$\gamma_j := \hat{g}(x_m, y_n);$
S. 13	$SC(i, j) = DWF_{sigm}(i, j) * PWF(i, j) * \gamma_i * \gamma_j;$ (*SC: Symmetry Contribution*)
S. 14	$p = \frac{p_i + p_j}{2};$
S. 15	$SM_{sigm}(p) = \sum_{(i,j) \in T(p)} SC(i, j);$ (*SM: Symmetry Magnitude*)
	end

그림 1. 순차 NTGST 알고리즘  
Fig. 1. Sequential NTGST algorithm.

NTGST는 마스크 안의 모든 화소 쌍에 대하여 중간한 점에 미치는 대칭 기여도를 누적할 때 각 화소쌍마다 명도변화의 크기 계산과정 S.3-S.5와 방향 계산과정 S.6, S.7, 위상 가중 함수 및 거리 가중 함수를 곱하는 과정 S.13 등을 반복해야 하는 계산 집약적 알고리즘이다. 그 순차적인 연산처리 시간  $t_s$ 는

$$t_s = (4t_{add} + 4t_{mul} + 2t_{look-up})M_{pixel-pair}W_I H_I \quad (1)$$

로 나타낼 수 있고, 따라서 그 계산복잡도  $C_s$ 는

$$C_s = O(M_{pixel-pair}W_I H_I) \quad (2)$$

연산 시간,  $t_{look-up}$ 은 조건표 (look-up table)를 참조하는 시간,  $M_{pixel-pair}$ 는 마스크내의 모든 화소 쌍의 수이다. 입력영상의 가로 화소 수  $W_I$ 와 세로 화소 수  $H_I$ 는 BLU의 물리적 크기를  $W \times H$ 라고 할 때 각각

$$W_I = W/R_w \quad (3)$$

$$H_I = H/R_H \quad (4)$$

이다. 여기서  $R_W$ 와  $R_H$ 는 입력영상의 가로와 세로의 해상도를 나타낸다.

특히 위상 가중 함수를 구하기 위해서 S.10과 같이 정현함수를 구하고 각도 계산을 해야 할 뿐만 아니라 이들 모두가 부동 소수점 연산으로 이루어지므로 영상의 크기가 커지거나 마스크의 영역이 넓어질 경우 처리 시간이 많이 소요된다. 따라서 고해상도 영상 처리나 실시간 처리를 해야 하는 등의 특정 응용분야를 위해서는 고속화 연산에 대한 연구가 필요하다.

### III. NTGST의 병렬고속화

본 논문에서는 그림 2와 같이 NTGST 알고리즘을 SIMD 구조의 다중프로세서에서 고속처리 할 수 있도록 병렬화 하였다. 먼저 NTGST 알고리즘을 분석하여 처리시간이 많이 소요되는 부동소수점 연산을 소정의 오차 범위 내에서 정수 연산으로 바꾼 다음, SIMD 구조를 적용할 수 있는 부분들을 분리하여 각각을 SIMD 명령어로 구현하였다. SIMD 구조로 구현된 부분을 멀티 쓰레드로 구현하여 단일 프로세서뿐 아니라 다중 프로세서 시스템에서도 적용 가능하게 하였다.

#### 3.1 알고리즘의 수정

우선, NTGST 알고리즘의 처리 시 각 화소에서의 명도변화 크기와 방향 및 각 대칭화소에 대한 거리가중함수는 루프 내에서 처리되지 않아도 되는 과정들이므로

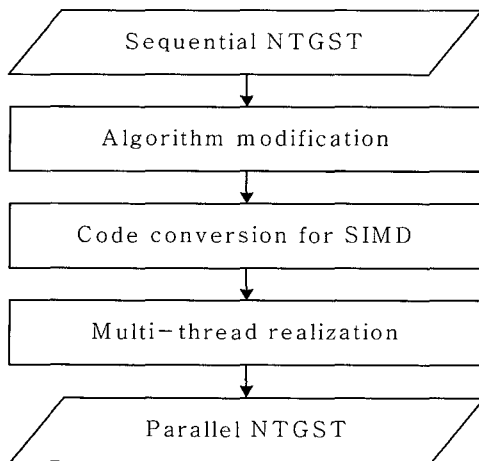


그림 2. NTGST의 병렬화 절차  
Fig. 2. Parallelization steps of NTGST algorithm.

루프 밖에서 별도 처리하도록 함으로써 일단 루프 연산 속도를 높였다.

한편, NTGST의 위상가중함수

$$P(i, j) = \sin\left(\frac{\theta_j + \theta_i}{2} - \alpha_{ij}\right) \times \sin\left(\frac{\theta_j - \theta_i}{2}\right) \quad (5)$$

에서는 두 개의 정현 함수값을 구해서 곱하는 계산뿐만 아니라 정현함수를 구하기 위한 각도 계산까지 부동소수점 연산으로 처리해야만 하므로 처리시간이 많이 소요된다. 이중 각도 계산과 정현 함수값의 계산은 소정의 오차범위 내에서 정수연산으로 변환하여 처리 속도를 향상시킬 수 있다. 또한 현재 MMX를 이용하는 프로세서의 하드웨어 구조상 정수형 SIMD가 부동 소수형 SIMD (SIMD-FP)를 사용하는 경우보다 연산속도가 빠를 뿐만 아니라 한 번에 묶어서(packing) 병렬 처리하는 데이터 수도 더 많다. 따라서 정현함수내의 각도 계산을 정수형으로 처리하고, 이를 조건표를 이용하여 부동 소수점형으로 변환하여 처리할 수 있게 알고리즘을 수정하였다. 또한, 기존의 순차 NTGST 알고리즘의 데이터 처리에 있어서 SIMD로 구현 가능한 부분만을 분리하고, 다시 분기문 제거 및 루프 펼침 (unrolling) 등의 작업을 수행하였다.

#### 3.2 SIMD 구조를 이용한 병렬화

하나의 명령으로 여러 개의 데이터를 병렬 처리할 수 있는 SIMD 구조는 반복적이고 병렬성이 강한 루틴에 적합하며, 멀티미디어의 처리나 그래픽, 음성통신 등 많은 부분의 데이터 처리가 이러한 성향을 가지므로 SIMD를 이용한 고속화가 바람직하다<sup>[8]</sup>.

현재 많이 사용되고 있는 Pentium 프로세서는 정수형 SIMD인 MMX 기술을 사용하고 있으며, 수식 (5)의 위상 가중 함수에서 정현함수 내부 인자 계산은 정수형으로 바꾸어 MMX 명령어로 처리하였으며, Pentium III 이후의 프로세서는 SSE에서 병렬부동소수점 연산인 SIMD-FP를 지원하므로 이를 이용하여 위상 가중 함수의 정현함수 곱은 병렬 SIMD-FP로 연산하였다.

NTGST의 대칭도 맵을 구하는데 사용되는 수식 (5)의 위상 가중 함수와

$$D_\sigma(i, j) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{|p_i - p_j|}{2\sigma}\right) \quad (6)$$

로 표시되는 거리 가중 함수, 그리고 각 대칭점의 명도 변화 크기를 곱하여 대칭기여도를 구하는 연산

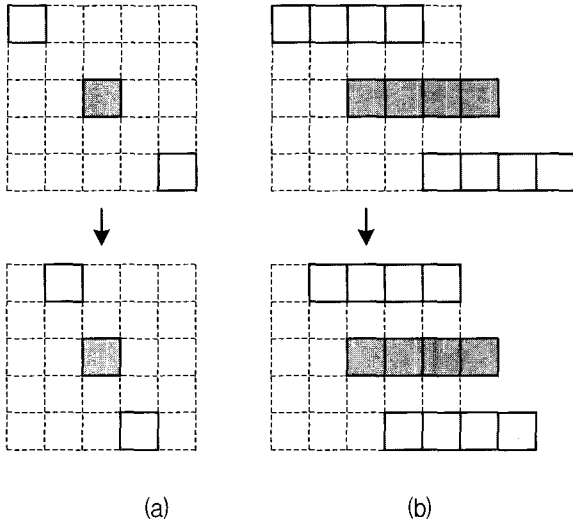


그림 3. (a) 순차 알고리즘 (b) 병렬 알고리즘  
 Fig. 3. (a) Sequential algorithm and (b) parallel algorithm.

$$C(i, j) = D_{\sigma}(i, j) P(i, j) r_i r_j \quad (7)$$

과 대칭도를 누적하여 대칭도 맵을 구하는

$$M_{\sigma}(p) = \sum_{(i, j) \in \Gamma(p)} C(i, j) \quad (8)$$

연산 등 NTGST에서 누적 대칭도를 구하는 모든 과정을 순차 알고리즘에서는 그림 3과 같이 하나의 화소 단위로 계산하던 것을 SIMD-FP 구조를 이용하여 한 번에 네 개의 화소에 대해서 동시에 계산하도록 병렬알고리즘으로 수정하여 네 배의 연산속도 향상을 기대할 수 있도록 하였다.

### 3.3 다중쓰레드를 이용한 병렬화

한편, NTGST 알고리즘은 계산에 있어서 영상의 화소 및 영역간의 의존성이 없으므로 동일한 작업량이 되도록 영상을 동일 크기로 공간 분할하여 각각을 쓰레드로 구현하고, 그림 4에서와 같이 다중프로세서 시스템에서 멀티 쓰레드로 처리함으로써 프로세서 수에 비례한 속도 향상을 얻을 수 있도록 하였다.

이때 영상 메모리로부터 SIMD 레지스터로 데이터를 빠르게 읽기 위해서 검사영상을 가로로 분할하였다. 이때 그림 5에서와 같이 공유영역에 저장된 영상데이터는 시간과 공간적으로 두 개의 프로세서에서 동시에 접근하지 않도록 하고 경계면에서의 마스크 스캐닝은 겹쳐지도록 하여 분할 경계면의 처리 문제를 해결하였다.

NTGST 알고리즘을  $N$ 개의 데이터를 한 번에 처리

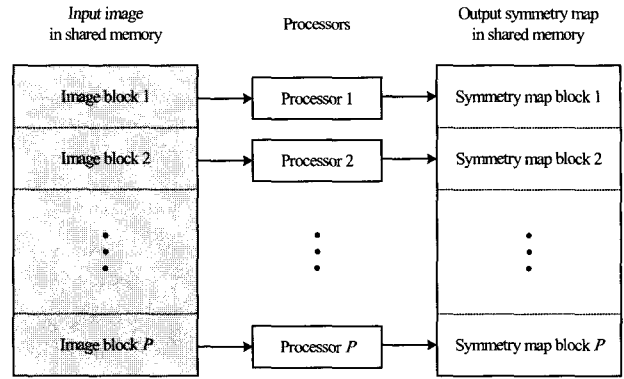


그림 4. 다중프로세서에서의 영상 분할 처리  
 Fig. 4. Image partition and processing onto multiprocessor.

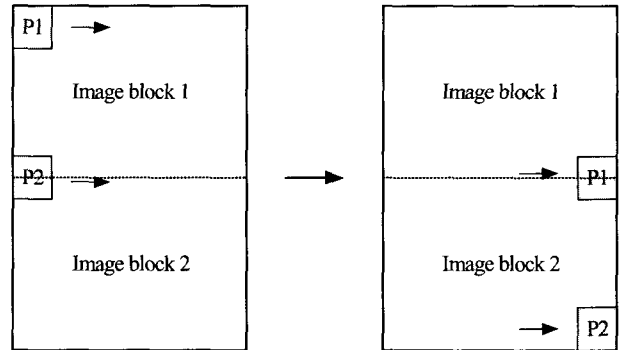


그림 5. 영상에서 각 할당된 프로세서의 처리과정  
 Fig. 5. Processing step for each of the allocated processes in partitioned image.

하는 SIMD 병렬 구조에 적합하도록 변환하고, 개인 컴퓨터에 일반적으로 사용되는 Pentium 프로세서의 SIMD 명령을 이용하여 병렬구현 하였다. NTGST 연산에서 공유영역에 있는 검사영상을 시공간적으로 중복되지 않게  $P$ 개의 프로세서에서 나누어 처리하도록 멀티 쓰레드로 구현함으로써 종합적으로 NTGST 알고리즘의 병렬 처리시간  $t_p$ 는

$$t_p = \frac{(4t_{add} + 4t_{pl} + 2t_{look-up})M_{pixel-pair}W_I H_I}{NP} \quad (9)$$

이되고, 계산복잡도  $C_p$ 는

$$C_p = O(M_{pixel-pair}W_I H_I / NP) \quad (10)$$

이 된다. 따라서 병렬 알고리즘의 수행속도는 기존의 순차 알고리즘에 비해 알고리즘의 변환에 의한 속도향상까지 포함하여 속도제고율  $S_p$ 는

$$S_p = t_s / t_p = O(NP) \quad (11)$$

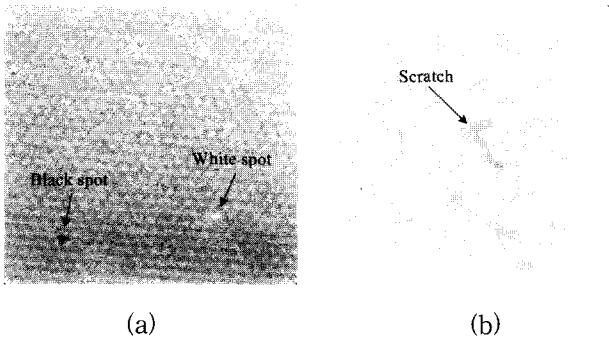


그림 6. 해상도별 BLU 검사영상 예: (a) 백점 및 흑점 (100×100 $\mu\text{m}^2$ /pixel)과 (b) 흠집 (20×20 $\mu\text{m}^2$ /pixel)  
 Fig. 6. BLU inspection images with (a) white, black spots in 100×100 $\mu\text{m}^2$ /pixel and (b) a scratch in 20×20 $\mu\text{m}^2$ /pixel resolution.

의 속도향상이 가능하도록 하였다. 속도제고율이 순수하게 SIMD 구조에서 한번에 처리할 수 있는 데이터 개수  $N$ 과 프로세서의 개수  $P$ 에만 비례하므로, 병렬 알고리즘의 수행속도는 SIMD 구조에 의한 확장은 고정되어 있지만 프로세서의 개수에 따라 선형적으로 증가하므로 제안한 알고리즘은 신축적인 확장성을 가진다.

#### IV. 고해상도 BLU 검사에 병렬 NTGST의 적용

그림 6에는 실제 BLU 검사 영상에서 격자형태의 음영과 배경에 대해 낮은 대비를 가지는 얼룩과 흠집 등을 각 해상도별로 예시하였다.

앞서 언급한 것처럼 15.2 인치 LCD의 BLU에서 10 $\mu\text{m}$  폭의 흠집까지 검사하려면 8192 화소의 고해상도 라인 스캐너를 사용하여 정밀검사를 해야 한다. 이 경우 BLU의 물리적 크기는 가로  $W$ 는 335mm이고 세로  $H$ 는 228mm이므로 입력영상 가로  $W_I$ 는 8192화소, 세로  $H_I$ 는 6000화소의 영상 크기를 가지며, 256 레벨의 밝기는 1 byte로 표기할 수 있으므로 49152000 bytes의 데이터 크기가 된다. 한편, 일반적으로 BLU 검사 장비에서 NTGST 수행에 요구되는 시간은 전술한 바와 같이 약 20초 정도인데 기존의 NTGST 방법으로 상기의 고해상도 영상을 검사하는 데는 약 130초가 소요되므로 이를 20초 이내로 줄이기 위해서는 최소한 일곱 배 이상의 속도 개선이 요구된다.

#### V. 실험 결과 및 고찰

제안한 알고리즘의 성능을 평가하기 위해서 개인용

컴퓨터에서 널리 쓰이는 MMX SIMD 병렬화 기술의 표 1. 15.2 인치 BLU의 해상도별 검사 영상에 대한 검사시간 비교

Table 1. Comparison of the inspection time for each resolution for 15.2 inch BLU.

Unit : [sec]

Image size: $W_I \times H_I$ [Pixels] (Resolution)	3000×2048 (112×111 $\mu\text{m}^2$ /pixel)	8192×6000 (41×38 $\mu\text{m}^2$ /pixel)
Methods		
Conventional NTGST	16	130
Proposed parallel NTGST	2	16

Dual Pentium III 1GHz 프로세서와 256MB의 SDRAM을 장착한 시스템을 사용하여 15.2 인치 BLU의 해상도별 여러 영상 크기에 대해서 실험하고 그 결과를 기존의 NTGST와 비교하였다. 표 1에는 검사 해상도를 증가시킴에 따라 화소 단위의 영상 크기의 변화와 그때마다 순차적 NTGST와 제안한 병렬 NTGST의 검사시간을 비교하였다. 표에서 보는 바와 같이 평균 여덟 배의 속도 향상을 얻을 수 있었으며, 따라서 15.2 인치 LCD의 고해상도 BLU 검사에서 한 장 검사하는데 요구되는 20초라는 실무 요구조건을 충분히 만족시킬 수 있음을 확인하였다.

현재 시중에서 사용되고 있는 PC의 SIMD 연산에서  $N$ 은 정해져 있다. 따라서 검사대상 BLU의 크기가 증가하거나, 같은 크기라 해도 요구되는 검사 해상도가 높아지면 그에 따라 검사에 필요한 프로세서의 개수  $P$ 를 신축적으로 증가시켜서 요구조건을 만족시킬 수 있다.

#### VI. 결론

고해상도의 BLU 영상을 고속으로 검사하기 위하여 검사 알고리즘인 NTGST를 병렬 고속화하였다.  $N$ 개의 데이터를 동시에 처리할 수 있는 SIMD 구조의 프로세서를  $P$ 개 사용할 경우  $O(NP)$ 의 속도향상을 얻을 수 있도록 구현하였다. 제안한 병렬고속 NTGST 알고리즘을 Dual Pentium III 프로세서를 사용하여 병렬구현하고 15.2 인치 LCD 공정상에서 10 $\mu\text{m}$  폭의 미세 흠

집까지 검출할 수 있는 고해상도 BLU 검사 영상에 적용한 결과, 기존의 NTGST의 처리속도를 여덟 배 이상 고속화 할 수 있었다. 따라서 제안한 검사시스템은 PC 레벨에서 여러 가지 크기의 LCD의 BLU에 대한 컴퓨터 비전 검사에서 프로세서 개수를 가변하여 신속적으로 대응할 수 있을 뿐만 아니라, 그 외에도 컴퓨터 비전의 여러 응용분야에 걸쳐 신속성 있는 활용이 가능할 것으로 판단된다.

### 참 고 문 헌

- [1] J. Jang, S. Lim, M. Oh, "Technology Development and Production of Flat Panel Displays in Korea," Proceeding of the IEEE, Vol. 90, No. 4, Apr. 2002.
- [2] C. J. Park, S. H. Cho, and H. M. Choi, "An Implementation of Noise-Tolerant Context-free Attention Operator and its Application to Efficient Multi-Object Detection," IEEEK Transaction on Signal Processing, vol. 38SP, no. 1, pp. 89-96, Jan. 2001.
- [3] C. J. Park, "A Noise-Tolerant Context-free Attention Operator for Efficient Multi-Object Detection," Ph. D. Thesis, Kyungpook National University, Daegu, Korea, 2000.
- [4] C. J. Park, W. G. Oh, S. H. Cho, and H. M. Choi, "An efficient context-free attention operator for BLU inspection of LCD," Proc. of the IASTED SIP, pp. 251-256, Las Vegas, Nevada, USA, Nov. 2000.
- [5] A. Peleg and U. Weiser, "MMX Technology Extension to the Intel Architecture," IEEE Micro, vol. 16, no. 4, pp. 42-50, Aug. 1996.
- [6] S. K. Raman, V. Pentkovski, J. Keshava, "Implementing streaming SIMD extensions on the Pentium III processor," IEEE Micro, vol. 20, no. 4, pp. 47-57, July-Aug. 2000.
- [7] D. Reisfeld, H. Wolfson, and Y. Yeshurun, "Context-free attentional operators: The generalized symmetry transform," IJCV, vol. 14, pp. 119-130, Jan. 1995.
- [8] A. Peleg, S. Wilkie, and U. Weiser, "Intel MMX for Multimedia PCs," Communications of the ACM, vol. 40, no. 1, pp. 25-38, Jan. 1997.

---

### 저 자 소 개



김복만(정회원)  
1999년 경일대학교 전자공학과  
학사 졸업.  
2002년 경북대학교 전자공학과  
석사 졸업.  
<주관심분야: 영상처리, 병렬분산  
처리, 임베디드 프로세서>

서경석(정회원)  
제38권 SP편 3호 참조

최흥문(중신회원)  
제38권 SP편 3호 참조