

# 인접 가해 성분을 이용한 적응적 선형 축소 알고리즘의 성능 분석

곽노윤\*

\*천안대학교 정보통신학부

e-mail:nykwak@cheonan.ac.kr

## Performance Analysis of Adaptive FOD Algorithm Using Neighbor Intelligible Components

No-Yoon Kwak\*

\*Division of Information and Communication Engineering,  
Cheonan University

### 요 약

본 논문은 중심 화소의 FOD 성분값과 인접 가해 성분값의 평균으로 축소 성분값을 산출함으로써 FOD에 적응성을 부여한 디지털 영상 축소 알고리즘의 성능을 분석함에 그 목적이 있다. 제안된 방법은, 중심 화소의 우측 및 하측 인접 화소의 크기를 이용하여 산출한 각각의 국부 가해 가중치를 우측 및 하측 인접 화소값에 곱한 후에 그 결과를 합산함으로써 인접 가해 성분값을 구하고 FOD 성분값과 이 인접 가해 성분값을 평균하여 축소 성분값을 구하는 과정을 전체 영역에 반복적으로 수행함으로써 축소 영상을 얻을 수 있다. 제안된 축소 방법에 따르면, 적은 연산량을 요하면서도 평균적으로 우수한 결과를 제공하는 FOD 방식의 장점을 취하면서 인접 화소의 유효 가해 성분을 각각의 국부 가해 가중치에 따라 축소 성분값에 적응적으로 반영함으로써 FOD의 단점인 뭉뚱화 현상을 효과적으로 억제시킬 수 있는 바, 개선된 정보 보존성을 제공할 수 있는 이점이 있다. 본고에서는 주관적인 성능과 하드웨어 복잡도 측면에서 제안된 방법과 기존의 각 방식에 대한 성능을 분석·평가한다.

### 1. 서론

기본적으로 디지털 영상 축소는 축소율에 따라 결정된  $n \times n$  영역을 대표하는 하나의 화소값을 결정하고 이 화소값으로 해당 영역을 대체하는 과정을 전영역에 걸쳐 반복·적용함으로써 원영상의 크기를 줄이는 작업이다. 따라서, 비록 지금까지 알려진 방식들 중 가장 진보된 결과를 제공하는 알고리즘을 채택하더라도, 하나의 축소 성분값을 결정할 때에는  $n \times n$  영역을 모두 버리는 단계를 공히 포함하고 있기 때문에 정도의 차이는 있지만 이 과정에서 영상의 가해성에 영향을 미치는 유효 가해 정보가 함께 유실되거나 축소 성분값에 그 특성이 불충분하게 또는 다소 왜곡되게 반영되는 문제를 피할 수가 없다.

축소를 수행하는 필터의 설계 영역 측면에서 디지털 영상 축소 기술을 살펴보면, 크게 공간 영역 처리와 주파수 영역 처리로 나누어 살펴볼 수 있다. 공간 영역 처리 방식에는 저주파 필터링을 수행하는 컨벌루션 연산자(convolution operator)를 통해 영상을 원하는 크기로 줄이는 방법이 있으며, 컨벌루션 연산자를 이용한 방법의 한계를 극복하기 위해 반선형 필터 또는 적응형 필터의 특성을 이용하는 방법 등이 있다. 전자의 예로는 ZOD(Zero Order Decimation) 방법[1][2], 선형 축소(FOD: First Order Decimation) 방법[1][2], Cubic B-Spline[3]을 이용한 축소 방법 등이 있고, 후자의 예로는 미디안 필터(Median Filter)를 이용한 방법[1][2], Pseudomedian 필터를 이용한 방법[4], ELA(Edge based Line Average)를 이용한 축소 방법[5] 등이 대표적이다. 후자는 전자에 비해 상대적으로 많은 연산

량을 요구함에도 불구하고 뚜렷한 성능 향상 폭을 제공하지 못하는 것으로 평가되고 있다. 한편, 주파수 영역 처리 방식은 축소 함수에 대한 수학적 분석을 통해 푸리에 연산자(Fourier operator)에 기반한 축소 연산자를 정의하고 이를 통해 주파수 영역에서 영상을 축소하는 것을 일컫는다. 최근 들어, 이 방식의 일종으로, MPEG-1, MPEG-2, H.261, H.363 등과 같은 멀티미디어 코덱의 DCT(Discrete Cosine Transform) 영역에서 화면의 크기를 축소하는 기법이 다수 제안되고 있기도 하다[6][7][8].

기존의 디지털 축소 방법들 중 FOD는 성능에 비해 하드웨어적인 복잡도가 낮고 연산량이 적을 뿐만 아니라 전체적으로 볼 때 우수한 정보 보존성을 제공하기 때문에 가장 보편화되어 있는 축소 기법이다. 다만, FOD 방법은 필터 설계의 특성상, 텍스처 영역이나 평탄 영역에서는 우수한 정보 보존성을 제공하는 반면, 화소값이 급변되는 스텝 에지 영역에서는 전경 화소와 배경 화소의 평균값을 축소 성분값으로 출력함에 따라 경계선을 시각적으로 뭉뚱화시키는 단점이 있다. 특히, 문서 영상과 같이 단일 화소 크기의 라인 에지 정보가 포함된 영역에서는 전경과 배경의 대비를 크게 저하시키는 문제가 있다.

제안된 디지털 영상 축소 방법은 이러한 문제를 개선하기 위해 연구된 것으로, 디지털 영상 축소에 있어서, 중심 화소의 FOD 성분값과 인접 가해 성분값의 평균으로 축소 성분값을 산출함으로써 FOD에 적응성을 부여함에 그 목적이 있다. 제안된 방법은, FOD 성분값을 축소 성분값의 주 성분으로 선택하고, 1차 미분 연산자를 이용하여 중심 화소

의 우측 및 하측 인접 화소의 기울기의 크기를 계산한다. 이후, 두 기울기의 크기값을 합산한 결과로 각 기울기의 크기값을 나누어 우측 및 하측 인접 화소 각각의 국부 가해 가중치(intelligible weight)를 구한다. 다음으로, 각각의 국부 가해 가중치를 우측 및 하측 인접 화소값에 곱한 후에 그 결과를 합산함으로써 인접 가해 성분값을 산출한다. 이렇게 구한 FOD 성분값과 인접 가해 성분값을 평균하여 축소 성분값을 구하는 과정을 전체 영역에 반복적으로 수행함으로써 축소 영상을 얻을 수 있다. 제안된 방법은 FOD 방식의 장점을 취하면서 인접 화소의 유효 가해 정보를 각각의 국부 가해 가중치에 따라 축소 성분값에 적용적으로 반영함으로써 FOD의 단점인 뭉뚱화 현상을 효과적으로 억제시킬 수 있는 바, 개선된 정보 보존성을 제공할 수 있는 장점이 있다.

2. 제안된 적응형 선형 축소 알고리즘

제안된 방법은 3x3 슬라이딩 윈도우(sliding window)를 사용한다. 수직 및 수평 방향에 대해 1/2배의 축소율을 적용하고자 할 경우, 화면 가장자리의 2화소 간격 내에 존재하는 화소들에는 3x3 슬라이딩 윈도우를 적용하는 것이 적절하지 않은 바, 이 영역은 FOD 방식으로 축소 성분값을 구한다. 3x3 슬라이딩 윈도우를 래스터 스캐닝(raster scanning) 방향을 따라 두 화소 간격 단위로 이동시키면서 슬라이딩 윈도우 영역에 포함된 화소값들을 이용하여 축소 성분값을 결정한다.

그림 1은 8x8 영상에 있어서, 제1윈도우 및 제2윈도우의 각 중심 화소의 위치와, I(x+1, y+1) 위치에서 단일 화소 폭으로 교차되는 수평 라인과 수직 라인을 보여주고 있다.

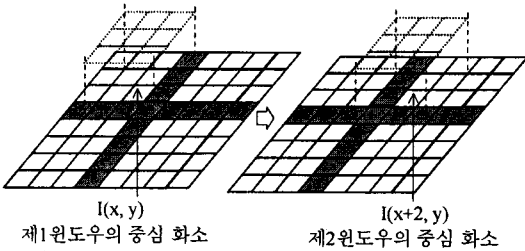


그림 1. 각 윈도우의 중심화소와 유효 가해 정보

이 8x8 영상에서 가장 중요한 가해 정보는 흑색으로 표시된 수직 라인과 수평 라인 상에 존재한다. 그러나, ZOD 방식을 적용할 경우, 제1윈도우에서도 백색이 선택되고 제2윈도우에서도 백색이 선택되는 바, 축소 영상에는 수직 라인의 정보가 나타나지 않게 된다. 물론, 중심 화소가 라인 위에 위치하도록 윈도우가 설정되면 좋은 결과를 얻을 수 있지만, 확률적으로 볼 때, 이러한 경우가 항상 보장될 수는 없다. 한편, FOD를 적용할 경우, 전경과 배경의 평균을 취하기 때문에 뭉뚱화를 피할 수가 없다. 즉, FOD는 인접 화소들에 존재하는 유효 가해 정보를 평균적으로 반영하기 때문에 유효 가해 정보와 다른 정보가 동일한 가중치로 반영되어 뭉뚱화 현상이 발생한다. 그러나, ZOD와 같이 이를 원천적으로 손실시키는 경우는 적어도 발생하지 않는다.

따라서, 제안된 방법에서는 FOD 성분값을 축소 성분값의 주성분으로 선택하고, FOD가 손실하는 인접 가해 성분을 축소 성분값에 적용적으로 반영하기 위해 중심 화소의 우측과 하측 인접 화소에 존재하는 유효 가해 정보의 양을 기울기의 크기값으로 정량화하고자 하는 것이다. 이를 위해 1차 미분 연산자∇1을 이용하여 중심 화소의 우측 인접 화소의 기울기의 크기값  $G_R$ 과 하측 인접 화소의 기울기의 크기값  $G_B$ 를 계산한다.

영상을 2개의 변수를 갖는 함수  $I(x, y)$ 로 가정할 때, 기울기(gradient)는 크기와 방향을 갖는 바, 식(1)과 같이 2차원 벡터로 나타낼 수 있다. 물론, 디지털 영상은 함수가 아니라 이산 데이터이기 때문에 연속신호처럼 미분을 취할 수 없다. 따라서, 인접 화소값과의 차분값으로 근사화시키는데, 가장 단순한 형태가 식(2)와 같은 1차 미분 연산자∇1이다[9]. 다양한 1차 미분 연산자가 있지만, 제안된 방법의 경우, 기울기를 통해 인접 화소의 국부 가해 가중치를 추정하는 것이기 때문에 이를 선정할 시에 최우선적으로 고려해야 하는 것은 국부성(localization)이다. 식(2)는 (x, y) 위치의 기울기를 구한 것이 아니라 (x-1/2, y-1/2) 위치의 기울기를 구한 것이지만 여타의 미분 연산자 비해 노이즈에 민감함에도 불구하고 국부성(localization)이 뛰어나 상대적으로 좋은 결과를 제공한다. 기울기의 크기값  $G_{mag}$ 은 식(3)과 같다.

$$\nabla I(x, y) = \left( \frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right) \tag{1}$$

$$\nabla_x I = \frac{\partial I}{\partial x} = I(x, y) - I(x-1, y) \tag{2}$$

$$\nabla_y I = \frac{\partial I}{\partial y} = I(x, y) - I(x, y-1) \tag{2}$$

$$G_{mag} = \sqrt{\left( \frac{\partial I}{\partial x} \right)^2 + \left( \frac{\partial I}{\partial y} \right)^2} \tag{3}$$

이때, 식(4)와 같이 절대값 연산에 의해 기울기의 크기값  $G_{max}$ 을 산출하더라도 무방하다. 이는 성능에 유의미한 변화를 주지 않으면서 연산량을 감축할 수 있다.

$$G_{mag} = \left| \frac{\partial I}{\partial x} \right| + \left| \frac{\partial I}{\partial y} \right| \tag{4}$$

이렇게 구한 두 기울기의 크기값이 각각 기설정된 임계치보다 작을 경우, 0으로 설정하고 그렇지 않으면 그대로 그 값을 이용한다. 본고에서는 식(3)을 통해 기울기의 크기값을 계산하는 경우에 준할 때, 기설정된 임계치로서 127을 사용하고 있다.

이것은 기울기의 크기값이 임계치보다 작을 경우에는 이를 국부 평탄 영역으로 간주함을 의미한다. 제안된 방법은 기울기의 크기를 통해 시각적으로 명확하게 드러나는 유효 가해 정보의 양만을 추정하여 반영하고자 하는 것이다. 만일, 모든 기울기의 크기에 대해 국부 가해 가중치를 계산할 경우, 노이즈에 의한 악영향을 배제시키기 어렵고, 불명확하고 모호한 유효 가해 정보로 인해 FOD의 단점인 뭉뚱화 효과를 동반하는 문제를 야기시킬 수 있다. 이후, 두 기울기의 크기값을 합산하여 그 결과값  $G_{sum}$ 이 0이면, 평탄 영역으로 간주하여 중심 화소값을 축소 성분값으로 결정하고 그렇지 않을 경우, 식(5)와 같이 합산한 결과로 각 기울기의 크기값을 나누어 우측 인접 화소의 국부 가해 가중치  $W_R$ 와 하측 인접 화소의 국부 가해 가중치  $W_B$ (즉,  $W_B = 1 - W_R$ )를 각각 정의한다.

$$W_R = \frac{G_R}{G_R + G_B}, \quad W_B = \frac{G_B}{G_R + G_B} \tag{5}$$

이후, 식(6)과 같이 각각의 국부 가해 가중치를 우측 및 하측 인접 화소값에 곱한 후에 그 결과를 합산함으로써 인접 가해 성분값  $I_m(x, y)$ 을 산출한다.

$$I_m(x, y) = W_R \times I(x+1, y) + W_B \times I(x, y+1) \tag{6}$$

이렇게 산출한 인접 가해 성분값은 우측 및 하측 인접 화소들의 국부 가해성(local intelligibility)의 기여도를 반영

하는 값이다. 다음으로, FOD 성분값  $I_{FOD}(x, y)$ 을 식(7)을 이용하여 구한다.

$$I_{FOD}(x, y) = \frac{I(x, y) + I(x+1, y) + I(x, y+1) + I(x+1, y+1)}{4} \quad (7)$$

이후, 식(8)과 같이, FOD 성분값  $I_{FOD}(x, y)$ 과 인접 가해 성분값  $I_m(x, y)$ 을 평균하여 축소 성분값  $I_D(x, y)$ 을 구하는 과정을 전체 영역에 반복적으로 수행함으로써 축소된 영상을 얻을 수 있다.

$$I_D(x, y) = \frac{I_{FOD}(x, y) + I_m(x, y)}{2} \quad (8)$$

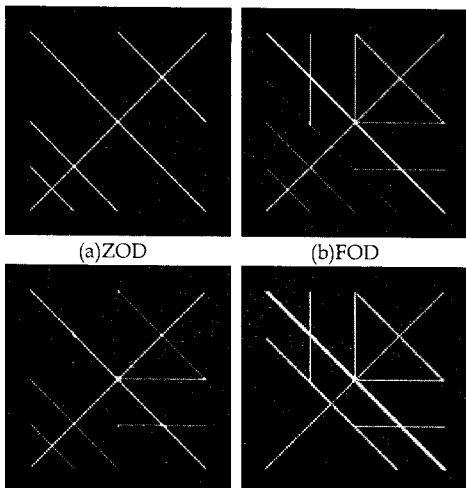
### 3. 시뮬레이션 결과 및 성능 분석

본 논문에서는 제안된 방법의 성능과 적용의 보편성을 평가하기 위해 기존의 방법과 제안된 방법에 대한 컴퓨터 시뮬레이션을 수행하였다. 시험 영상으로, 512×512의 LENA, 1687×2346의 CHART, 256×256의 CROSS 영상과 308×232의 KORCHAR (한글 문서 영상) 등을 이용하였다.

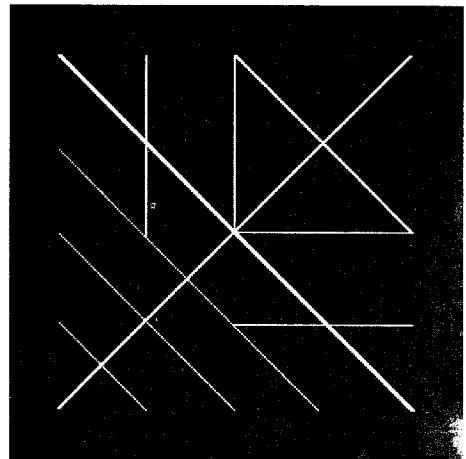
#### 3.1 주관적인 성능 분석

컴퓨터 시뮬레이션을 통해 LENA와 같은 자연 영상을 대상으로 각 방식을 적용할 경우, 공간적인 상관성에 기인하여 각 방식의 축소 영상들 간에는 주관적 성능차가 현저하지 않음을 확인할 수 있었다. 그림 2와 그림 3는 각각 CROSS 영상과 KORCHAR 영상을 대상으로 각 방법을 적용한 결과 영상을 나타낸 것이다.

CROSS 영상은 수직 라인과 수평 라인 및 대각선 라인을 인위적으로 생성시켜 만든 시험 영상이다. ZOD와 H-shaped PMED의 경우, 유효 가해 정보가 많이 손실당하고 있다. 이에 반해 FOD와 제안된 방법은 완벽하지는 않지만, 원영상에 포함된 라인 에지 정보를 대부분 보존하고 있는 것을 알 수 있다. KORCHAR 영상은 인터넷 신문의 기사를 소프트웨어적으로 캡처한 시험 영상이다. ZOD와 H-shaped PMED는 해당 기사의 내용을 읽기 힘들 정도 저급한 성능을 제공한다. 특히, ZOD는 그 열악함의 정도가 더 심각하다. 그림 3(b)는 FOD의 장점과 단점을 동시에 보여주고 있다. 유효 가해 정보를 원천적으로 잃어버리지는 않지만, 전경 화소와 배경 화소의 대비를 다소 떨어뜨리고 있음을 알 수 있다. 그러나, FOD는 연산량에 비해 매우 우수한 성능을 제공하는 것으로 평가할 수 있다.

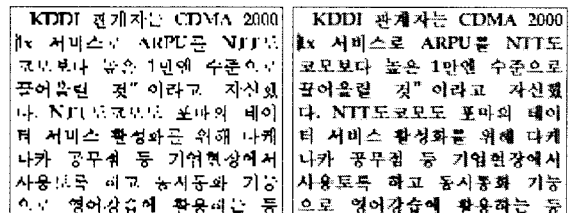


(c)H-shaped PMED (d)제안된 방법



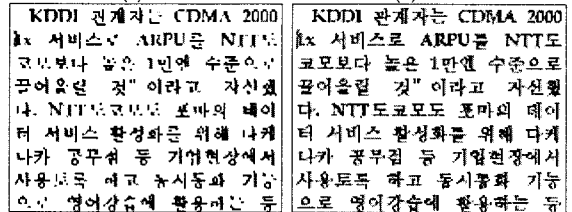
(e)원영상

그림 2. CROSS 영상을 대상으로 각 방법을 적용한 결과 영상



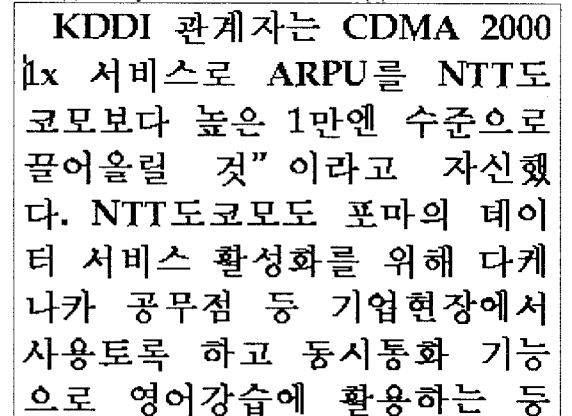
(a)ZOD

(b)FOD



(c)H-shaped PMED

(d)제안된 방법



(c)원영상

그림 3. KORCHAR 영상을 대상으로 각 방법을 적용한 결과 영상

그림 3(d)의 제안된 방법의 결과 영상은 FOD에 비해 대 비도가 높고 유효 가해 정보에 대한 보존력이 우수하지만, FOD에 근거한 것인 바 뒤통화 현상을 피할 수가 없다. 그러나, 기존의 방법들에 비해 시각적인 거부감이 적고 대 비도가 양호함을 확인할 수 있다. 특히, 그림 3(d)에서 살펴볼 수 있듯이, 모음 '키', '히', '기' 등에서 기존의 방식에 비해 획이 좀 더 명료함을 발견할 수 있다.

3.2 연산량 비교

표 1은 각 방법을 적용하여 하나의 화소를 축소할 시에 요구되는 연산량을 비교한 것이다.

표 1. 각 방법의 연산량 비교

축소 방식	연산	가산	감산	승산	제산	비교 연산	자승 연산	제곱근 연산
ZOD	-	-	-	-	-	-	-	-
FOD	3	-	-	1	-	-	-	-
CUBIC B-SPLINE	24	-	25	1	-	-	-	-
ELA	1	3	1	-	2	3	-	-
H-shaped PMED	1	-	1	-	11	-	-	-
Proposed Method	$G_{sum}=0$	6	4	-	1	3	4	2
	$G_{sum}!=0$	8	5	3	2	3	4	2

구성 및 처리 방법 간에 다소 차이가 있기 때문에 각 방식을 완벽하게 동일한 조건에서 비교할 수는 없지만, 표 1을 통해 사칙 연산량과 비교 연산량 및 절대값 연산량을 비교해 봄으로써 통상적인 하드웨어 설계 시, 시스템의 복잡성과 실시간 구현성 정도를 대략적으로 살펴볼 수 있다. 이때, 연산을 수행할 시에 요구되는 임시 저장 수단인 버퍼의 수(또는, 레지스터수)는 비교 대상에서 제외하기로 한다. 제안된 방법은 두 인접 화소의 기울기 합산값  $G_{sum}$ 이 0인 경우, 6회 가산, 4회 감산, 1회 제산, 3회 비교 연산, 4회 자승 연산 및 2회 제곱근 연산이 소요된다. 한편,  $G_{sum}$ 이 0이 아닌 경우, 인접 가해 성분값을 계산한 후에 중심 화소값과 인접 가해 성분값을 평균하기 위해, 8회 가산, 5회 감산, 3회 승산, 2회 제산, 3회 비교 연산, 4회 자승 연산 및 2회 제곱근 연산이 추가로 소요된다. 각 경우의 연산량에 대한 산출 방식의 기준을 변경하면 다소 다른 결과가 나올 수 있다. 제안된 방법이 있어서, 각 경우의 발생 빈도는 각 시험 영상에 따라 다를 수 있지만, 평균적인 연산량을 통해 연산량 평가를 수행하더라도 큰 무리는 없을 것으로 판단된다. 그림 4는 제안된 방법의 하드웨어 구성도를 예시한 것이다.

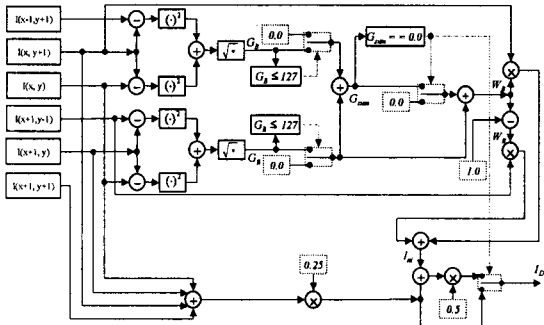


그림 4. 제안된 방법의 하드웨어 구성도

설계 방법에 따라 다양한 구현이 가능하겠으나, 그림 4의 하드웨어 구성도는 실제 동작 시의 타이밍을 고려하지 않은

것이며 최적화보다는 제안된 알고리즘의 전체적인 동작 흐름에 대한 이해를 돕기 위한 것이다.

4. 결론

본 논문에서는 컴퓨터 시뮬레이션을 통해 제안된 방법과 기존의 방법들의 주관적 성능을 비교·분석 함으로써 제안된 방법의 우수성을 평가하였다.

본 연구 과정에서 얻은 결론은, 좋은 축소 결과를 얻기 위해서는 다음과 같은 두 가지 접근법에 유념해야 한다는 것이다. 첫 번째로, 슬라이딩 윈도우의 크기를 어느 정도로 설정하는 것이 바람직한가하는 것이다. 두 번째로, 축소 성분값을 결정하는 방법이다. 제2 영상상의 경우, 각종 방법론을 동원하여 슬라이딩 윈도우에 포함된 화소값들 중에서 해당 영역을 가장 잘 대표할 수 있는 하나의 화소값을 선정한다 할지라도 이것만으로는 만족스러운 결과를 얻을 수 없다. 그렇다고 FOD처럼 슬라이딩 윈도우에 포함된 모든 화소값을 선별 없이 동일한 가중치로 평균하는 것도 마찬가지다. 좋은 결과를 얻기 위해서는 유효 가해 정보에 대한 기여도에 따라 슬라이딩 윈도우에 들어오는 모든 화소들의 국부 가해 가중치를 명확하게 정의하고 이를 통해 각 화소값들의 가중 평균을 구하는 것이다. 제안된 방법은 적응형 알고리즘에 기초한 것이기 때문에 FOD에 비해 연산량과 하드웨어 복잡도가 높지만, 성능향상에 비해 복잡도가 소폭 증가하는 바, 하드웨어적 구현 측면에서도 실용성이 있다. 한편, 1차 미분 연산자  $\nabla 1$ 을 이용하는 방법 외에도 인접 화소의 국부 가해 가중치를 정의하는, 좀 더 개선된 방법이 있을 수 있음에 따라 이에 대한 부가적인 연구도 필요할 것으로 판단된다. 제안된 방법에는 아직 미진한 부분이 남아 있는 바, 향후, 축소 영역을 하나의 화소값으로 좀 더 효과적으로 대체함으로써, 시각적인 거부감의 초래를 최대한 억제할 수 있을 뿐만 아니라 원영상의 특성과 형상을 가능한 한 보존할 수 있는 좀 더 진보된 연구가 필요할 것으로 사료된다.

참고 문헌

- [1] Randy Crane, A Simplified Approach to Image Processing, Prentice Hall, pp.112-135, 1997.
- [2] D. Phillips, Image Processing in C, R&D Publications, Inc., pp.288-294, 1994.
- [3] M. A. Sid-Ahmed, Image Processing-Theory, Algorithms, and Architectures, McGraw-Hill, pp.167-177, 1995.
- [4] 채종석, 조화현, 최철호, 박진성, 권병현, 최명렬, "고해상도 FPD에서 Image Decimation을 위한 T-형 축소 알고리즘", 제13회 산학연 멀티미디어 산학기술 학술대회 논문집, 제13권, pp.36-39, Mar. 1999.
- [5] K. Oistamo, Y. Neuvo, "A Motion Intensive Method for Scan Rate Converter and Cross Error Cancellation", IEEE Trans. Consumer Electronics, vol.37, No.3, pp.296-301, Aug. 1991.
- [6] B. K. Natarajan and V. Bhaskaran, "A Fast Approximate Algorithm for Scaling Down Digital Images in the DCT Domain", Proc. ICIP 95, 1995.
- [7] N. Merhav and V. Bhaskaran, "Fast Algorithms for DCT-Domain Image Down Sampling and for Inverse Motion Compensation", IEEE Trans. on CSVT, vol.7, No. 3, Jun. 1997.
- [8] 서광덕, 이상희, 김재균, 고종석, "MPEG 영상을 위한 DCT 영역에서의 화면 축소 기법", JCCI '99 논문집, pp.3-7, 1999. 4.
- [9] J. R. Parker, Algorithms for Image Processing and Computer Vision, Wiley Computer Publishing John Wiley & Sons, Inc., pp.10-16, 1997.