빛 확산 함수를 이용한 이차원 디밍 알고리즘

정회원 정 혜 동*, 종신회원 강 성 진**°

Two-Dimensional Dimming Algorithm using Light Spread Function

Hye-Dong Jung* ***Regular Member***, Sung-Jin Kang**° ***Lifelong Member***

요 약

본 논문에서는 LCD TV의 전력 절감을 위해 빛 확산 함수(Light Spread Function)을 이용한 LED BLU(Backlight Unit)의 디밍 제어 알고리즘을 제안하고, 효율적으로 구현할 수 있는 방법을 제시한다. 기존의 이차원 디밍 제어 알고리즘은 BLU의 LED를 MxN개의 블록으로 분할하고, 각 분할 영역에 해당하는 입력 영상에 따라 백라이트의 밝기를 계산하여 디밍 제어를 수행한다. 그러나, 각 블록의 LED 빛은 확산되어 인접 블록에 영향을 주기 때문에 각 블록의 최종 휘도는 디밍 제어 값보다 크게 된다. 제안된 알고리즘은 빛 확산 함수를 이용하여 각 블록이 인접 블록으로부터 영향 받는 양 만큼 줄인 디밍 제어 값을 사용함으로써, 기존의 알고리즘보다 더 많은 전력 절감이 가능해진다.

Key Words : Dimming, LCD TV, Backlight, LED, Spread Function

ABSTRACT

In this paper, in order to reduce the power consumption of LCD TV, we present a two-dimensional dimming algorithm for LED BLU(Backlight Unit) using light spread function and a pragmatic implementation method. In conventional two dimensional dimming algorithms, LED BLU is divided into MxN blocks and proper backlight luminance for each block is computed according to input image and used for dimming control. But, LED light of each block is diffused and affects the neighboring blocks so that luminance of each block becomes larger than that of intent. In the proposed algorithm, dimming control values are reduced by the amount of quantity affected by the neighboring blocks using light spread function, resulting in additional reduction of power consumption.

Ⅰ. 서 론

Liquid Crystal Display(LCD)디스플레이 장치는 기존의 Cathode Ray Tube(CRT) 디스플레이 장치에 비하여 얇고, 선명하다는 장점을 가지고 있다. 따라서, 경량화, 박형화에 대한 소비자의 욕구가 증대됨에 따라 현재 디스플레이 시장에서 LCD 디스플레이는 대부분을 차지하고 있다.

Backlight Unit(BLU)는 LCD 디스플레이 장치에서 광원의 역할을 수행하는 요소로서 전체 시스템에서 소비되는 전력의 60% 이상을 차지하고 있는 실정이다. 따라서 BLU의 효율적인 제어를 통한 소비 전력의 절감[1-6]은 LCD 디스플레이 장치를 사용하는 사용자 각자의 에너지 절약 효과 뿐 아니라 국가적인 에너지 절약 차원에서도 매우 중요하다.

LCD TV에서 BLU의 광원으로 대부분 Cold

* 전자부품연구원 RFID·USN융합연구센터(hudson@keti.re.kr),
** 한국기술교육대학교 정보기술공학부(sjkang@kut.ac.kr), (° : 교신저자)
논문번호 : KICS2010-07-304, 접수일자 : 2010년 7월 14일, 최종논문접수일자 : 2010년 7월 29일

Cathode Fluorescent Lamps(CCFL)을 사용했기 때문에, 0차원 디밍(global dimming) 또는 1차원 분할 디밍(one dimensional local dimming) 제어를 수행했지만[1-3], 최근에는 Light Emitting Diode(LED)를 사용하는 방식으로 변화됨에 따라, 디밍 제어 연구도 LED BLU에 적합한 기술이 연구되고 있다. LED BLU는 CCFL BLU와는 다르게 2차원으로 분할하여 LED를 배치할 수 있기 때문에, 전체 화면을 M x N개의 블록으로 분할하여 2차원 디밍 제어(two dimensional local dimming)가 가능하다[4-6].

기존의 이차원 디밍 제어 방식은 각 분할 영역마다 화면의 최대 밝기 정보를 추출하여 BLU의 밝기를 적절히 줄임으로써 LCD TV의 소비 전력을 절감하고 있다.[4-6] 하지만 원 영상과의 차이를 줄이기 위해 BLU의 밝기가 줄어든 양에 비례하여 영상 데이터의 밝기를 증가시켜 주어야한다. 영상 데이터를 보상할 때, BLU의 각 블록의 빛은 확산되어 인접 블록에 영향을 미치기 때문에, 빛 확산 함수(Light Spread Function)[7]을 이용하여 인접 블록의 영향까지 고려해야 한다[6]. 이는 각 블록의 최종 밝기는 인접 블록에서 빛이 확산되어 들어오기 때문에 실제 디밍 제어된 값보다 밝아지기 때문이다. 따라서, 각 블록이 특정 밝기를 가지기 위해서는 인접 블록의 영향을 반영하여 디밍 제어 값이 더 작아질 수 있음을 알 수 있다.

본 논문에서는 빛 확산 함수를 영상 데이터를 보상할 때에만 사용하는 것이 아니고, 디밍 제어 값을 계산할 때에도 이용하는 이차원 디밍 알고리즘을 제안한다. 제안된 이차원 디밍 알고리즘은 2단계로 디밍 제어 값을 계산하는데, 1단계에서는 [4-6] 등과 같이 각 블록의 화면 밝기 정보를 이용하여 디밍 제어 값을 계산한 후에, 2단계에서 빛 확산 함수를 이용하여 실제 제어할 디밍 값을 다시 계산한다. 2단계에서 계산된 디밍 제어 값은 1단계에서 계산된 값보다 작기 때문에, 추가적인 전력 절감을 기대할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 제안된 이차원 디밍 알고리즘에 대하여 설명하고 3장에서는 구현을 위해 간략화된 이차원 디밍 알고리즘을 설명한다. 4장에서는 실험결과를 보여주고, 5장에서 결론을 맺는다.

Ⅱ. 제안된 디밍 알고리즘

LED BLU를 이차원 디밍 제어 값을 계산하는 알고리즘은 많이 연구되어 왔지만, 본 논문에서는 설명의 편이상 식 (1)과 같은 간단한 형태의 알고리즘을 사용한다. 즉, 입력 영상을 $K = M \times N$개의 블록으로 분할하여, k번째 블록 내의 픽셀 데이터 중에서 가장 큰 밝기(Luminance)값을 찾은 후에, 해당하는 k번째 블록의 LED의 밝기를 제어한다.

$$L_k = \max\{L(x,y) \mid (x,y) \in k^{th}\ \text{block}\} \qquad (1)$$

여기에서 $0 \le k \le K-1$ 이고, L_k는 k번째 블록의 디밍 제어 값이고, $L(x,y)$는 (x,y)에 위치하는 픽셀 밝기이다.

그림 1은 한 블록의 밝기 값을 최대로 했을 때, 화면의 밝기 분포를 측정한 후에 정규화된 분포이다[7]. 본 논문에서는 이 빛 확산 함수를 $g(x,y)$로 정의한다.

그림 1로부터 한 블록의 광원은 해당 블록에만 영향을 미치는 것이 아니고, 인접한 주변 블록의 밝기에도 영향을 미치는 것을 알 수 있다. 또한, $g(x,y)$가 한 블록 내에서도 균일하지 않음을 확인 할 수 있다. 따라서, 각 픽셀에 대한 백라이트의 실제 밝기 값은 다음과 같이 됨을 알 수 있다.

$$L^*(x,y) = \sum_{k=0}^{K-1} B_k \cdot g(x - x_k,\ y - y_k) \qquad (2)$$

여기에서, (x_k, y_k)은 k번째 블록의 중심 좌표이다. 이상적으로는, 식 (2)를 계산한 후에 식 (1)과 같이 다시 계산하여 밝기 제어 값을 계산해야하지만, 식 (2)가 계산량이 너무 많고, LED BLU는 각 픽셀 단위로 제어를 하지 않고 블록단위로 제어하기 때문에, 식 (2)에서 각 블록의 중심 좌표에 대해서만 계산을 하면 다음 식과 같이 표현할 수 있다.

$$L_k^* = \sum_{l=0}^{K-1} L_l \cdot g(x_k - x_l,\ y_k - y_l) \qquad (3)$$

식 (1)과 (3)으로부터, 항상 $L_k^* \ge L_k$이 성립됨을

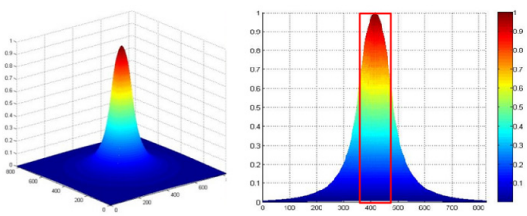

그림 1. 빛 확산 함수 $g(x,y)$

알 수 있다. 따라서, 아래 식 (4)를 만족하는 d_k가 존재한다면, 항상 $d_k \le L_k$이 성립한다. 따라서, 식 (1)의 L_k대신 식 (4)를 만족하는 d_k값에 따라 디밍 제어를 함으로써 추가로 소비 전력을 절감할 수 있다.

$$\begin{aligned} L_k &\approx \sum_{l=0}^{K-1} d_l \cdot g(x_k - x_l, y_k - y_l) \\ &\equiv \sum_{l=0}^{K-1} d_l \cdot g_{k,l} \end{aligned} \tag{4}$$

식 (4)에서 $g_{k,l} \equiv g_l(x_k - x_l, y_k - y_l)$로 나타내었다. 식 (4)를 만족하는 d_k를 얻기 위해서, 아래 식과 같은 행렬을 정의한다.

$$\mathbf{L} = \begin{bmatrix} L_0 & L_1 & \cdots & L_{K-1} \end{bmatrix}^T \tag{5}$$

$$\mathbf{D} = \begin{bmatrix} d_0 & d_1 & \cdots & d_{K-1} \end{bmatrix}^T \tag{6}$$

$$\begin{aligned} \mathbf{G} &= \begin{bmatrix} \mathbf{G}_0 & \mathbf{G}_1 & \cdots & \mathbf{G}_{K-1} \end{bmatrix}^T \\ &= \begin{bmatrix} g_{0,0} & g_{0,1} & \cdots & g_{0,K-1} \\ g_{1,0} & g_{1,1} & \cdots & g_{1,K-1} \\ \vdots & \vdots & & \vdots \\ g_{K-1,0} & g_{K-1,1} & \cdots & g_{K-1,K-1} \end{bmatrix} \end{aligned} \tag{7}$$

여기에서, T는 행렬의 전치(transpose)를 의미한다. 식 (5)~(7)을 이용하여, 식 (4)를 식 (8)와 같이 행렬 형태로 표현하면, 식 (9)와 같이 d_k를 계산할 수 있다.

$$\mathbf{L} \approx \mathbf{G} \cdot \mathbf{D} \tag{8}$$

$$\mathbf{D} \approx \mathbf{G}^{-1} \cdot \mathbf{L} \tag{9}$$

식 (9)에서 얻은 d_k는 L_k의 값의 분포에 따라 음의 값을 가지는 경우가 있으며, 이러한 음의 값에 대한 디밍 제어는 의미가 없으므로, 실제 디밍 제어에 사용되는 디밍 제어 값 d_k^*은 식 (10)과 같다.

$$d_k^* = \begin{cases} 0, & \text{if } d_k \le 0 \\ d_k, & otherwise \end{cases} \tag{10}$$

식 (10)의 d_k^*을 사용했을 때 각 블록의 밝기 $\widehat{L_k}$는 식 (11)과 같다.

$$\widehat{L_k} = \sum_{l=0}^{K-1} d_l^* \cdot g(x_k - x_l, y_k - y_l) \tag{11}$$

식 (9)에서 얻은 d_k를 사용하는 대신에, 식 (10)의 d_k^*를 사용했기 때문에, $L_k \le \widehat{L_k} \le L_k^*$인 관계가 성립함을 알 수 있다.

제안된 디밍 제어 알고리즘은 LCD TV 제조시에 빛 확산 함수 $g(x,y)$를 이론적 또는 실험적으로 계산하여 $\mathbf{G}$행렬을 만든 후에 $\mathbf{G}^{-1}$를 미리 계산하여 메모리 저장하여 이용한다. 따라서, LED BLU 제어 모듈에서는 식 (9)와 식 (10)을 이용하여 d_k^*을 계산하는 기능이 필요하다.

Ⅲ. 간략화된 디밍 알고리즘

현재 국내 L사에서 출시되고 있는 직하형 LED BLU를 사용하는 LCD TV의 경우, 42인치는 192블록, 47인치는 216블록, 55인치에는 240블록으로 분할하여 디밍 제어를 하는 것으로 알려져 있다. 따라서, 식 (9)의 행렬 계산은 전체 블록 수 K가 많아지면 계산량이 많아서 실제 구현에 적합하지 않을 수 있다. 따라서, 제안된 알고리즘의 성능을 크게 저하시키지 않는 간략화된 알고리즘이 필요하다.

그림 1의 빛 확산 함수는 중심에서 멀어질수록 급격하게 값이 감소하는 것을 알 수 있다. 따라서, 빛 확산 함수를 어느 기준 값 이하에서 '0'으로 근사화 한다면, 그림 2에서와 같이 $M \times N$개의 블록중에서 한 블록(검정색 블록)에 영향을 미치는 블록은 주변의 몇몇 블록으로 제한될 것으로 예측할 수 있다. 따라서, 행렬 $\mathbf{G}^{-1}$가 그림 2와 같은 분포를 갖는 다면 식 (9)의 계산량은 현저하게 작아질 수 있을 것으로 예측할 수 있다.

식 (7)의 $\mathbf{G}_k$행렬의 원소를 그림 2의 각 분할 영역과 대응되도록 아래 식과 같이 다시 정의를 한다.

$$\begin{aligned} \mathbf{G}_k &= \begin{bmatrix} g_{k,0} & g_{k,1} & \cdots & g_{k,K-1} \end{bmatrix} \\ &\equiv \begin{bmatrix} g_{k,0} & g_{k,0} & \cdots & g_{k,N-1} \\ g_{k,N} & g_{k,N+1} & \cdots & g_{k,2N-1} \\ \vdots & \vdots & & \vdots \\ g_{k,(M-1)N} & g_{k,(M-1)N+1} & \cdots & g_{k,K-1} \end{bmatrix} \end{aligned} \tag{12}$$

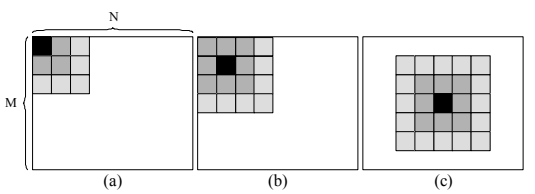

그림 2. 영향을 미치는 인접 블록 예

식 (9)의 $\mathbf{G}^{-1}$를 설명의 편의를 위해 식 (13)과 같이 재정의한다.

$$\mathbf{G}^{-1} = \Lambda = [\Lambda_0 \quad \Lambda_1 \quad \cdots \Lambda_{K-1}]^T \tag{13}$$

여기에서 Λ_k는 식 (12)와 유사하게 식 (14)와 같이 정의한다.

$$\begin{aligned}\Lambda_k &= [\lambda_{k,0} \quad \lambda_{k,1} \quad \cdots \quad \lambda_{k,K-1}] \\ &\equiv \begin{bmatrix} \lambda_{k,0} & \lambda_{k,0} & \cdots \lambda_{k,N-1} \\ \lambda_{k,N} & \lambda_{k,N+1} & \cdots \lambda_{k,2N-1} \\ \vdots & \vdots & \vdots \\ \lambda_{k,(M-1)N} & \lambda_{k,(M-1)N+1} & \cdots \lambda_{k,K-1} \end{bmatrix}\end{aligned} \tag{14}$$

$\mathbf{G}_0$와 Λ_0의 값의 분포는 그림 2(a)와 유사한 분포를 가지며, $\mathbf{G}_{N+1}$과 Λ_{N+1}은 그림 2(b)와 유사한 분포를 가진다. 아래 식 (15)와 (16)은 $\mathbf{G}_0$와 Λ_0에 대한 예를 보여준다. 식 (15)는 52인치 LCD TV에서 128분할한 경우에 빛 확산 함수를 측정한 후 정규화 및 근사화시켜 얻은 값이며, 식 (16)은 $\mathbf{G}^{-1}$를 계산하여 얻은 Λ_0에서 3블록이상 떨어진 블록의 밝기 값을 '0'을 근사화하여 얻은 값이다.

$$\mathbf{G}_0 = \begin{bmatrix} 1 & 0.1871 & 0.0396 & 0.0133 & 0 & \cdots & 0 \\ 0.1526 & 0.0810 & 0.0281 & 0.0110 & 0 & \cdots & 0 \\ 0.0297 & 0.0233 & 0.0133 & 0 & 0 & \cdots & 0 \\ 0.0094 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \tag{15}$$

$$\Lambda_0 = \begin{bmatrix} 1.0583 & -0.1815 & -0.0001 & -0.0046 & 0 & \cdots & 0 \\ -0.1403 & -0.0304 & -0.0026 & -0.0037 & 0 & \cdots & 0 \\ -0.0024 & -0.0028 & -0.0040 & 0.0063 & 0 & \cdots & 0 \\ -0.0061 & 0.0075 & 0.0021 & -0.0004 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \tag{16}$$

따라서, k번째 블록에 영향을 주는 인접한 n_l번째 블록의 총 개수를 L이라 하면 식 (9)의 d_k는 식 (17)과 같이 계산된다.

$$d_k = \sum_{l=0}^{L-1} \lambda_{k,n_l} L_{n_l} \tag{17}$$

식 (17)에 의해 d_k가 계산되면, 식 (10)에 의해 최종 디밍 제어 값 d_k^*를 얻을 수 있으며, 이 때 각 블록의 최종 밝기는 식 (11)과 같이 된다. 식 (19)와 (17)을 비교하면 연산량이 현저하게 감소될 수 있음을 알 수 있다.

Ⅳ. 성능평가

그림 3은 52인치 128분할한 경우에 빛 확산 함수를 측정한 후, 정규화 및 근사화시켜 얻은 값이다.

그림 4는 1080x1920 화면을 8x16=128분할했을 때, 식(1)에 의해 계산된 디밍 제어 값이다. 이 경우에 디밍 제어를 하지 않은 경우에 비해 약 26.85%의 전력 절감을 가진다. 그림 5는 식(17)에 의해 다시 계산된 디밍 제어 값이며, L_k를 사용하는 경우에 비해 추가로 약 41.10% 전력 절감을 할 수 있다. 그림 6은 d_k^*

Index	-3	-2	-1	0	1	2	3
-3	0	0	0	0.009437	0	0	0
-2	0	0.013282	0.023346	0.029738	0.023346	0.013282	0
-1	0.011042	0.028095	0.080965	0.1526	0.080965	0.028095	0.011042
0	0.013282	0.039604	0.187111	**1**	0.187111	0.039604	0.013282
1	0.011042	0.028095	0.080965	0.1526	0.080965	0.028095	0.011042
2	0	0.013282	0.023346	0.029738	0.023346	0.013282	
3	0	0	0	0.009437	0	0	0

그림 3. 실험에 사용된 빛 확산 함수 $g(x,y)$

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	82	97	106	111	170	173	201	218	206	192	185	185	153	155	149	125
1	82	98	128	162	192	228	248	249	253	239	206	189	184	154	148	128
2	87	119	157	178	210	255	255	255	255	255	253	211	201	172	147	136
3	88	115	158	193	232	255	255	255	255	255	255	236	203	177	158	153
4	97	126	164	199	243	255	255	255	255	255	255	235	205	158	113	104
5	98	125	149	200	232	255	255	255	255	255	253	227	201	160	143	132
6	92	140	177	211	220	242	255	255	255	249	231	189	162	136	106	123
7	140	176	162	178	203	197	210	224	219	205	182	187	125	105	88	79

그림 4. 식 (1)에 의한 디밍 제어 값 L_k

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	54	52	53	39	98	79	103	117	103	94	94	104	71	84	84	81
1	40	32	48	68	77	100	111	105	113	104	75	69	78	52	61	66
2	42	49	69	67	81	115	103	102	102	105	116	77	85	68	53	71
3	41	39	63	77	98	108	102	101	102	102	107	102	80	73	69	95
4	49	50	69	78	108	106	102	101	101	102	108	97	85	57	23	42
5	48	46	44	80	95	108	101	101	101	102	110	96	88	63	65	78
6	31	54	76	95	87	104	112	108	110	109	101	68	62	54	29	77
7	98	111	79	88	111	94	105	119	113	105	84	113	52	49	41	46

그림 5. 식 (17)과 (10)에 의한 디밍 제어 값 d_k^*

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	80	94	103	108	169	172	201	218	205	191	184	183	151	153	147	123
1	79	95	124	159	191	228	249	250	254	239	205	187	182	151	146	126
2	85	116	154	175	209	255	255	255	255	255	253	209	199	170	144	133
3	85	112	155	191	231	255	255	254	254	254	254	236	201	175	155	150
4	95	123	161	197	243	255	254	254	254	254	255	234	203	155	110	101
5	95	122	146	198	232	255	255	255	254	254	253	226	199	157	140	129
6	89	137	174	209	219	242	255	255	255	249	231	188	160	133	103	121
7	138	174	160	176	202	196	209	224	218	205	181	186	123	102	86	77

그림 6. 식 (11)에 의한 LED BLU의 최종 밝기

로 디밍 제어 했을 때, LED BLU의 최종 밝기 값이며, 식 (11)에 의해 계산된 값이다.

그림 4와 그림 6의 결과 값이 매우 유사하기 때문에, 본 논문에서 제안된 디밍 제어 알고리즘을 사용하면, 최종 LED BLU의 밝기를 동일하게 유지하면서 추가적으로 전력 절감할 수 있다.

표 1은 8x16=128분할한 경우에 한 블록에 영향을 주는 인접(주변을 둘러싸고 있는) 블록을 최대 3블록 이내로 제한했을 경우에 식 (9)와 (17)의 계산량을 비교한 것이다. 식 (17)의 간략화된 디밍 알고리즘을 사용하면, 연산에 필요한 곱셈, 덧셈의 횟수도 현저하게 줄어들었고, 행렬 A의 계수를 저장하기 위해 필요한 메모리도 작음을 확인 할 수 있다. 표 1은 프로세서 또는 DSP를 이용한 S/W시에 필요한 연산량이며, H/W로 병렬 및 파이프라인 처리를 통해 효과적으로 ASIC으로 구현이 가능하다.

표 2는 그림 7에 나타나 있는 영상에 대하여 기존의 2차원 디밍 알고리즘과 제안된 디밍 알고리즘의 평균 밝기 오차를 나타낸다. 표 2에서 APL은 Average Picture Level을 나타내며, Average Error는 요구되는 밝기와 실제 밝기의 차이를 최대 밝기 값에 대한 비율(%)로 표현하였다. 예를 들어, 최대 밝기가 255(8bit)이고 평균 밝기 차이가 25.5이면 Average Error값은 10%가 된다. 따라서, 요구되는 밝기와 실제 밝기의 차이가 클수록 Average Error값이 커지게 되고, 이는 전력소모도 커진다는 것을 의미한다. 표 2

표 1. 복잡도 비교

	식(9)	식(11)
Frames/sec	30	30
전체블록수	128	128
블록당 평균 인접 블록수	128	34.375
초당 곱셈 횟수	491,520	132,000
초당 덧셈 횟수	491,520	132,000
계수 저장에 필요한 메모리(word)	16,384	4,400

표 2. 그림 7의 영상에 대한 APL 및 평균 밝기 오차

영상	APL(%)	Average Error(%)	
		Proposed	Conventional
(a)	61	33.89	74.00
(b)	41	33.82	69.25
(c)	39	30.71	60.10
(d)	34	15.72	44.03
(e)	27	28.35	47.04
(f)	20	36.34	53.08

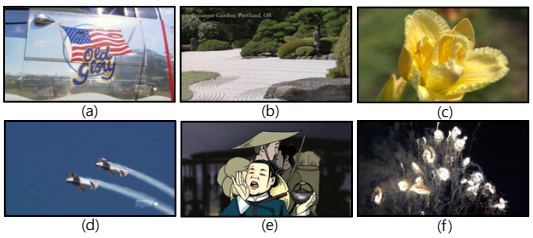

그림 7. 실험에 사용된 영상

로부터, 제안된 알고리즘을 사용했을 때 Average Error 값이 작게 나오기 때문에, 더 많은 전력 절감이 가능하다는 것을 알 수 있다.

V. 결 론

본 논문에서는 LED BLU를 사용하는 LCD TV에서 전력소모를 줄이기 위해 인접 블록의 영향을 고려한 이차원 디밍 제어 알고리즘을 제안하였다. 실험결과로부터 기존의 이차원 디밍 제어 알고리즘보다 더 많은 전력 절감이 가능함을 알 수 있다.

참 고 문 헌

〔1〕 Naehyuck Chang, Inseok Choi, Hojun Shim, “DLS: Dynamic Backlight Luminance Scaling of Liquid Crystal Display”, *IEEE Trans. on VLSI Systems*, Vol.12, No.8, pp.837-846, Aug., 2004

〔2〕 C. Lai, C. Tsai, “Backlight Power Reduction and Image Contrast Enhancement Using Adaptive Dimming for Global Backlight Applications,” *IEEE Trans. on Consumer Electronics*, pp.669-674, Vol.54, No.2, May 2008

〔3〕 J. Hong, S. Kim, W. Song, “A Clipping Reduction Algorithm Using Backlight Luminance Compensation for Local Dimming Liquid Crystal Displays,” *IEEE Trans. on Consumer Electronics*, pp.240-246, Vol.56, No.1, Feb. 2010

〔4〕 H.F.Chen, J. Sung, T. Ha, Y, Park, C, Hong, “Backlight Local Dimming Algorithm for High Contrast LCD-TV”, *Proceedings of ASID 2006*, pp.168-171, 2006

〔5〕 T. Shirai, S. Shimizukawa, T. Shiga, S. Mikoshiba, “RGB-LED backlight for LCD-TVs with 0D, 1D, and 2D adaptive dimming,” in SID’06 Dig., pp.1520-1522, 2006

〔6〕 W. Oh, D. Cho, K. Cho, G. Moon, B. Yang, T. Jang, “A Novel Two-Dimensional Adaptive Dimming Technique of X-Y Channel Drivers for LED Backlight System in LCD TVs,” *IEEE Journal of Display Technology*, pp.20-26, Vol.5, No.1, Jan. 2009.

〔7〕 Y. Cheng, Y. Lu, C. Tien, H. Shiech, “Design and Evaluation of Light Spread Function for Area-adaptive LCD System,” *IEEE Journal of Display Technology*, pp.66-71, Vol.5. No.2, Feb. 2009.

정 혜 동 (Hyedong Jung) 정회원

1998년 8월 경희대학교 전자공학과 학사
2002년 2월 경희대학교 전자공학과 석사
2007년 2월 경희대학교 전자공학과 박사 수료
2002년~현재 KETI RFID/USN 융합 연구센터 선임연구원

<관심분야> Rich Media, Universal Middleware, Local Dimming

강 성 진 (Sung-jin Kang) 종신회원

1998년 8월 연세대학교 전자공학과 공학박사
1998년 12월 ETRI 무선방송기술연구소 선임연구원
2000년 3월 (주)이노텔리텍
2002년 9월 KETI 통신네트워크연구센터 책임연구원
2007년 3월~현재 한국기술교육대학교 정보기술공학부 조교수

<관심분야> WPAN, WLAN, MODEM SoC, Local Dimming