

# PWM 구동 LED BLU LCD 보드상의 객체 기반 dimming 알고리즘 구현 연구

이병훈, 김영진  
 아주대학교 전자공학과  
 wphased@gmail.com, youngkim@ajou.ac.kr

## A study on implementation of an object-based dimming algorithm on a PWM driven LED BLU LCD board

Byung-Hoon Lee, Young-Jin Kim  
 Dept. of Electrical and Computer Engineering, Ajou University

### 요 약

사용자와의 인터페이스가 많은 모바일 기기에서 TFT LCD는 현재 가장 많이 사용되는 디스플레이로 백라이트(backlight)로 인한 전력 소모가 많아 배터리 수명이 짧아지는 것이 큰 문제이다. 전력 절감을 위해 사용되는 backlight dimming 기법은 명암비 조절에 따른 이미지 왜곡의 보완 기법과 함께 많은 연구가 진행되고 있다. 본 논문에서는 PWM 구동에 의한 LED backlight 제어가 가능한 TFT LCD 보드 구현을 소개하고 이 보드를 대상으로 framebuffer와 LED 제어용 리눅스 장치 드라이버를 사용한 객체 기반의 저전력 인간 시각 만족의 화질 보상 dimming 알고리즘의 구현 연구를 기술한다.

### 1. 서론

근래 스마트폰에서 TFT LCD는 현재 가장 많이 사용되는 디스플레이로 백라이트(backlight)로 인한 전력 소모가 많고 그로인해 배터리 수명이 줄어드는 등 문제가 있다 [1]. 본 논문에서는 PWM (Pulse Width Modulation) 구동에 의한 LED backlight 제어가 가능한 TFT LCD 보드 구현을 소개하고 이 보드를 대상으로 framebuffer와 LED 제어용 리눅스 장치 드라이버를 사용한 객체 기반의 dimming 알고리즘의 구현에 대해서 기술한다.

### 2. 배경

본 논문에서는 세부 영역(region)으로 나누어 각 region에 따른 다수 색깔(majority color)을 감지(detect)하여 local dimming을 적용하고 RGB 값을 보정(compensation)한 후 인간 시각 만족도를 충족하는지 확인하고자 한다. 이를 위한 객체 기반 local dimming 알고리즘은 다음과 같다. 한 이미지에서 각 BLU (Backlight Unit)에 따른 majority color를 구하는 과정을 통해 객체를 인식하고, 각 BLU마다 화질 보정(pixel compensation)을 진행한다[2]. backlight를 줄여가면서 pixel compensation을 통해 얻어진 majority color가 이미지의 최대값보다 작다면 해당 BLU를 왜곡된 지역으로써 분류한다. 미리 설정한 MSE 한계점(threshold)에 도달할 때까지 왜곡된 지역들의 backlight를 증가시킴으로써 이미지 강화가 이루어지며 그

림 1과 같이 처리된다. 다만, [2]에서는 시뮬레이터 상에서만 실험이 진행되었으므로 본 논문에서는 세부적인 처리 과정을 구현 보드에 맞게 처리할 것이다.

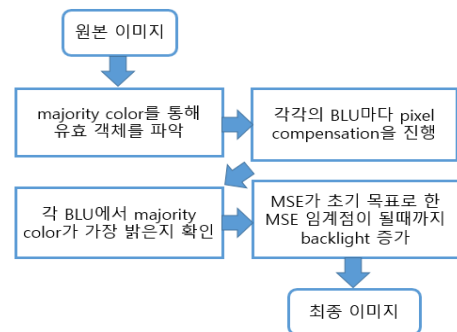


그림 1 . 객체 기반 dimming flowchart

YCbCr[3]은 영상 시스템에서 사용하는 색공간의 일종으로 밝기 성분과 색차 성분으로 이루어져 있다. 이러한 정보는 원본 RGB로부터 구하며 절대 색공간은 아니다. 따라서, YCbCr로 표현된 값은 표준 RGB 값을 사용했을 경우보다 예측이 용이해진다.

인간 시각 만족도를 충족시키는지 확인하기 위하여 SSIM (Structural SIMilarity)[4]을 이용하고자 한다. SSIM은 단순히 두 이미지 간의 pixel별 차이를 계산하는 MSE와 달리, 명암, 대비, 구조적인 이미지 차이를 규명하기 위해 설계되었다. 이것은 인간 시각 만족도를 얼마나

잘 지켜지고 있는지 평가하는 도구이나, 계산 부하가 너무 크고 하드웨어적으로 backlight의 효과를 포함시킬 수 없어 영상 처리 보드의 알고리즘엔 탑재하지 않았다. SSIM 값으로부터 이미지 품질을 판단하는 기준은 다음 표와 같다.

표 1. SSIM 값에 따른 quality 기준[5]

Value	Quality
0.98 ~ 1.00	High
0.96 ~ 0.98	Medium
0.94 ~ 0.96	Low
~ 0.94	Unacceptable

### 3. 연구 수행 내용

#### 3.1 하드웨어 부분

##### 3.1.1 보드 개발 환경

본 논문의 알고리즘이 진행되는 타겟 보드는 Hybus사와 협력하여 만든, 객체 기반 dimming이 가능한 임베디드 보드(그림 2)를 사용한다. 이 타겟 보드는 리눅스 환경에서 동작하며 800×480 TFT LCD의 크기를 가지고 있다.



그림 2. 영상 시험 보드

그림 2의 타겟 보드는 minicom을 통해 PC와 보드 사이에서 통신을 하게 되며, 실험 PC의 리눅스 버전은 우분투의 12.04버전이다. 영상 시험 보드 display의 BLU는 총 32개로 나뉘어져 있으며 각각의 LED가 dimming 되는 구조를 띠고 있다. minicom으로 통신 연결된 PC에서 밝기를 전체 화면이나 각각의 구역을 0에서 255범위 내에서 dimming 하게 된다.

##### 3.1.2 알고리즘 구현을 위한 PWM LED BLU

타겟 보드에서의 PWM 구동 LED 제어 방식은 그림 3과 같다. PCA9532는 16bit I2C 프로토콜을 사용하는 LED dimmer IC로 CPU와 연결되어 있다. 타겟 보드의 CPU가 2개의 IC로부터 16개의 LED BLU를 각각 제어하며 256 step으로 backlight를 조절한다. 이 IC는 내부적으로 PWM의 frequency와 duty cycle을 입력된 backlight에 따라 적절하게 programming됨으로써 동작한다.

#### 3.2 소프트웨어 부분

##### 3.2.1 backlight 제어

dimmer 기능은 보드 상에서 LED용 PWM 드라이버를 사용하여 구현된다. 이 기능은 32개의 각 BLU에 해당하

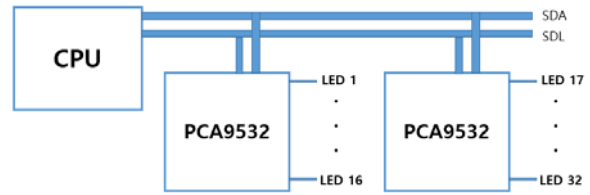


그림 3. 타겟 보드에서의 PWM 방식 LED 제어 방식

는 LED를 사용자가 지정하거나 이미 지정해 놓은 밝기로서 표현된다. RGB의 범위처럼 0에서 255로서 밝기를 제어하며 숫자가 작을수록 점점 밝아진다. 그러나 보통 숫자가 커질수록 밝아진다고 생각하기에 출력화면은 이를 고려하여 설계하였다.

##### 3.2.2 framebuffer의 제어

공개된 소프트웨어인 fbv (frame buffer viewer)[6]의 함수인 fb\_display를 사용하여 이미지를 framebuffer에 저장한 후 보드의 display에 표현한다.

이미지가 처리되는 과정은 다음과 같다. 먼저, 표현하고자 하는 이미지의 규격과 파일 형식에 대한 정보를 구하고 보드의 규격에 맞게 조정한다. 이어서 비트 수에 따라 RGB 값을 ioctl 함수를 이용해서 구한 후, alpha(투과도) 값에 따라 User level에서 framebuffer data를 LCD driver로 보내고, driver의 메모리 할당을 통해 LCD controller에 의해 LCD 상에 출력한다. 여기서의 framebuffer를 이용하여 본 논문에서 고안한 함수를 적용시킬 것이다.

### 4. local dimming을 위한 함수 고안

위 3.2에서 언급하였듯이, PWM 제어를 통한 backlight dimming 기법과 fb\_display 함수의 framebuffer에 접근하는 기능을 결합하여 만든 함수가 seldim 함수이다. 이 함수는 먼저 이미지가 화면에 출력되기 직전의 framebuffer에 접근한다. 이로부터 각 BLU 구간의 majority color를 detect하는데 이 부분이 객체를 인식하는 단계이다.

이어서, 객체를 기반으로 한 majority color로부터 YCbCr의 밝기 성분을 구한 후, 이에 따라 LED를 제어하여 local dimming을 수행한다. backlight의 증가율에 따라 반비례 형태로 RGB 값을 보정하는 과정을 거친다. 그 이유는 원본에 비해 backlight 조절에 따른 이미지 변형 과정을 거쳤으므로 인간 시각 만족을 위해 RGB를 보정하기 위함이다.

아래의 그림 4는 seldim 함수의 처리 과정을 순서도로써 표현한 것이다. 이 중에서 pixel value를 보정하는데 이때, 각 pixel의 변화폭은 원본의 RGB 값의 2%를 넘지 않도록 조절하였다. 여기서의 2%의 기준은 아래의 실험 결과로부터 도출하였다.

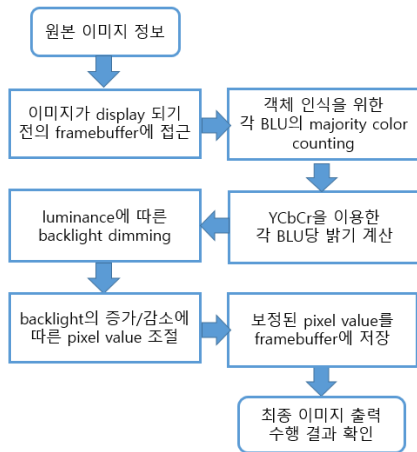


그림 4. seldim 함수의 순서도



그림 5. 원본 c4.jpg에 객체 기반 dimming 적용 전 캡춰



그림 6. c4.jpg에 객체 기반 dimming 적용 후 캡춰 (RGB 20% 보상)



그림 7. c4.jpg에 객체 기반 dimming 적용 후 캡춰 (RGB 10% 보상)



그림 8. c4.jpg에 객체 기반 dimming 적용 후 캡춰 (RGB 5% 보상)



그림 9. c4.jpg에 객체 기반 dimming 적용 후 캡춰 (RGB 2% 보상)

그림 5는 실험에 사용된 원본이미지이며 그림 6,7,8,9는 그림 4의 객체 기반 dimming 알고리즘에 따라 처리한 후 RGB 보정을 각각 20%, 10%, 5%, 2%를 수행한 결과이다. 2절에서 언급한 인간 시각 만족도 측정 도구인 SSIM을 이용하여 원본 이미지(그림 5)와 보정된 이미지(그림 6,7,8,9)를 비교해보니 각각 0.9885, 0.9961, 0.9970, 0.9992

의 값을 나타냈다. 이것은 모두 0.98 이상으로써 고품질이라는 뜻이긴 하나, 원본 이미지와 비교했을 때 블로킹 현상(blocking artifact)이 화질 보정을 크게 할수록 증가한다. 본 논문에서는 인간 시각 만족에 있어 좀 더 보수적인 결과를 얻기 위해 blocking artifact가 가장 적은 그림 9의 결과를 토대로 진행할 것이다. 구체적으로 아래 그림 10과 같이 밝기가 최대, 최소일 때를 기준으로 RGB의 변화가 2%의 범위 내에서 선형적인 식을 도출하였다.

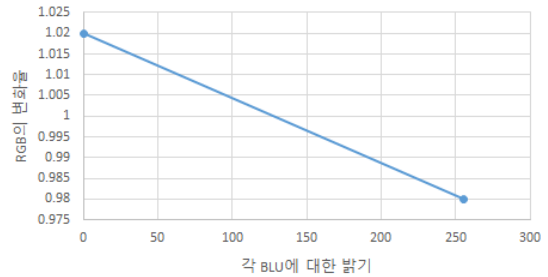


그림 10. 각 BLU의 밝기에 따른 RGB의 변화율

이 결과를 토대로 그림 4의 객체 기반 dimming 알고리즘을 임베디드 보드 상에 구현하기 위해 다음의 두 사진을 이용하였다.



그림 11. 원본 r1.jpg의 객체 기반 dimming 적용 전 캡춰



그림 12. 원본 r3.jpg의 객체 기반 dimming 적용 전 캡춰

## 5. 실험 결과 및 논의

### 5.1 실험 결과

backlight의 차이를 보여야 하므로, 그림 11,12에서의 객체 기반 dimming 알고리즘 적용 전 원본 이미지와 그림 13,14에서의 알고리즘 적용 후의 이미지는 보드 상에 출력된 화면을 카메라를 이용하여 촬영하였다. 아래 그림 15,16은 그림 11,12를 본 논문의 객체 기반 dimming 알고리즘을 수행한 후 보드의 각 BLU당 backlight 밝기 크기를 나타낸 결과 화면이다.

보통의 모니터나 TV의 경우, 밝기를 사용자가 임의로 제어하거나 global dimming으로 제어한다. 이것은 어두운 이미지까지 밝기를 필요이상으로 낭비하거나 또는 환한 부분까지 낮은 밝기로 표현함으로써 인간 시각 만족도를 떨어뜨리게 된다. 하지만 그림 15와 16에서 보듯이 각각의 BLU마다의 객체를 인식하고 그에 따라 각각 dimming하여 전력 소비 감소와 인간 시각 만족도를 개선시킨다.



그림 13. r1.jpg에 객체 기반 dimming 적용 후 캡춰



그림 14. r3.jpg에 객체 기반 dimming 적용 후 캡춰

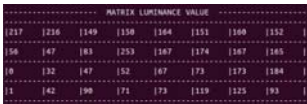


그림 15. r1.jpg에 객체 기반 dimming에 따른 BLU당 밝기

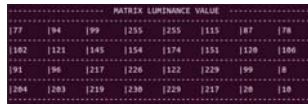


그림 16. r3.jpg에 객체 기반 dimming에 따른 BLU당 밝기

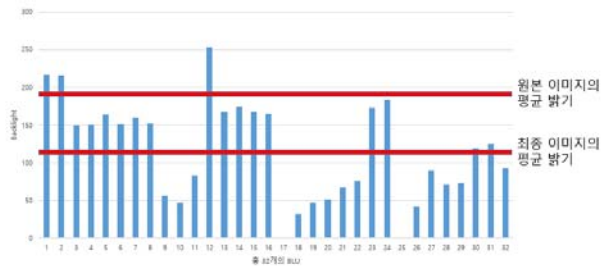


그림 17. r1.jpg를 화면에 출력 했을 때의 backlight 밝기 분포도

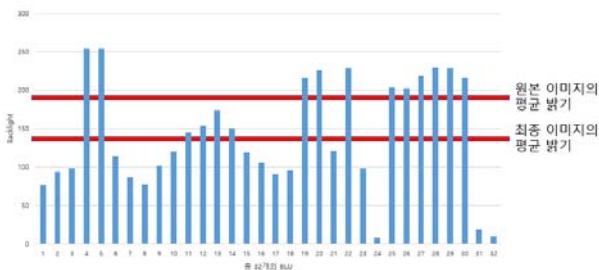


그림 18. r3.jpg를 화면에 출력 했을 때의 backlight 밝기 분포도

그림 10과 그림 11은 원본 이미지가 dimming 되고 난 후의 각 BLU마다의 밝기 크기를 히스토그램으로 나타내었다. 평균 LCD의 밝기가 최대 밝기의 75%정도로 설정하는 사실에 비추어볼 때, 최종 이미지에 비추는 backlight는 최대 밝기의 45%, 56%정도로써 약 30%, 26%의 전력 감소 효과가 있는 것으로 확인된다.

5.2 논의

그림 11과 그림 13의 0.9931의 SSIM 값을 나타내었으며 그림 12와 그림 14의 SSIM 값은 0.9959의 값을 나타내었다. 인간 시각 만족도를 위한 이미지 품질 측정하는 도구인 SSIM은 표 1에서 나타낸 것과 같이 두 이미지 사이의 값이 0.98 이상이면 고품질을 나타내는데 본 논문의

실험에서 쓰인 이미지의 SSIM 값은 모두 그 이상이므로 이미지 품질은 높다고 볼 수 있다. 그러나 화질 보정에 따른 blocking artifact가 선명하게 존재함에도 불구하고 고품질로 나온다는 부분은 보드 환경에 맞게 동적인 이미지 보상이 필요한 부분이다.

위의 결과에서 확인하였다시피 이미지의 majority color가 어두우면 어두울수록 전력 소모 절감이 크다. 한편, 본 논문의 객체 기반 dimming 알고리즘은 정지 사진(still picture)을 대상으로 하므로 수행 측면에서 동영상에서 이 기법을 적용하기 어렵기에 향후 개선이 필요하다.

6. 결론

본 연구에서는 객체 기반 저전력 및 인간 시각 만족의 화질 보정 dimming 알고리즘을 실제 임베디드 보드에서 PWM 구동에 의한 LED dimming 기법을 구현함으로써 전력 소모 절감을 도모하고자 하였다.

본 연구에서 실험한 이미지를 토대로 객체 기반 dimming 알고리즘을 수행한 결과가 그렇지 않은 것보다 30%, 26%정도의 전력 소모 절감 효과가 있다는 사실을 확인하였다. 이와 동시에 인간 시각 만족도를 측정하는 SSIM을 통해 원본과 비교하였을 때 화질 보정 이미지의 품질이 우수함을 확인하였다.

ACKNOWLEDGMENT

이 논문은 2013년도 정부(교육부)의 재원으로 한국과학창의재단(대학생 창의융합형 연구과제 지원사업)의 지원을 받아 수행된 연구임.

참고 문헌

[1] A. Carrol, H. Gernot, "An analysis of power consumption in a smartphone." Proceedings of the 2010 USENIX conference on USENIX annual technical conference. 2010.  
 [2] A. Anggorosesar, Y.-J.Kim, "Object-based Local Dimming for LCD Systems with LED BLUs", Proc. of the 2011 International Symposium on Low Power Electronics and Design (ISLPED), 2011.  
 [3] YCbCr <http://ko.wikipedia.org/wiki/YCbCr>  
 [4] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity", IEEE Trans. Image Processing, vol. 13, no. 4, pp. 600-612, Apr. 2004.  
 [5] A. Bartolini, M. Ruggiero, and L. Benini, "Visual quality analysis for dynamic backlight scaling in LCD systems", in Proc. of the 2009 Conference on Design, Automation, and Test in Europe (DATE), 2009  
 [6] fbv : the FrameBuffer Viewer, <http://www.eclis.ch/fbv/>