

# 단색 영상장치를 위한 개선된 Floyd-Steinberg 디더링 기법 연구

진 영 근  
청양대학 전자계산학과  
전화 : 041-940-6731

## Study on the Refining the Floyd-Steinberg Dithering For Monochrome Display

Young-Goun Jin  
Dept. of Computer Science, ChongYang College  
E-mail : ygjin@chongyang.ac.kr

### Abstract

This paper has been studied a method that effectively displaying color image to monochromatic display such as PDA and movable-phone. Generally, the Floyd-Steinberg dithering algorithm has been used in this area and its' effectiveness were well known. But it shows some ugly patterns in white area and also shows some directionality in vertical and horizontal directions. To reduce those directionality, I suggest the error diffusion direction to be rotated randomly according to the bit value of the current position. This can also mitigate some ugly pattern in white area

### I. 서론

디더링(dithering)은 원색의 영상보다 표시 색상 또는 계조단계가 적거나 제한된 영상표시장치나 출력 장치에 알맞도록 원 영상을 렌더링(rendering)하는 과정을 의미하며 경우에 따라서는 디지털 해프토닝(digital halftoning), 칼라 줄이기(color reduction)로 불리기도 한다. 이 기술은 초창기 색상이 제한된 영상표

시장이나 출력장치등의 하드웨어 특성에 맞게 영상을 처리하기 위한 기술로 비교적 많은 연구가 되어 왔으며 그러한 하드웨어 기술이 발달한 현재에도 많은 분야에서 이 기술이 적용되어 사용되고 있다. 예를 들면 현재 대부분의 가정에서 사용하고 있는 레이저프린터나 잉크젯 프린터에서 영상을 인쇄시 여러 가지 디더링 방법을 선택 또는 기본사양으로 하여 출력하며 특히 웹브라우저에서 구현할 수 있는 색상의 제한으로 원색의 영상을 웹에서 구현 가능한 색상으로 줄여주는 디더링 과정이 필요하다. 또 이동통신의 발달로 휴대폰 또는 개인용 휴대 단말기(personal digital assistant)가 많이 보급되어 있으며 이러한 휴대 단말 장치는 현재 대부분 디스플레이가 모노 형식이므로 영상정보를 휴대폰 또는 PDA에 표시하기 위해서는 원영상이 칼라 또는 계조 영상인 경우 이진영상으로 하는 디더링이 필수적이다.

디더링은 크게 랜덤 디더링, 패턴 디더링, 순서상(ordered) 디더링, 오류분산(error dispersion) 디더링의 네가지 방식으로 나누는데 랜덤 디더링 방식으로 원영상을 이진영상으로 하는 경우는 원영상의 색 또는 계조 범위내의 임의의 수를 발생시켜 그 값이 현재위치의 픽셀의 값보다 크면 백 작으면 흑으로 하는데 속도는 네가지 방식중에 비교적 빠르나 고주파성의 잡음

현상이 발생할 수 있으며 출력물의 질은 다른 방식보다 낮은 편으로 잘 사용되지 않는다. 패턴 디더링은 픽셀의 제조 범위에 따라 일정한 이진 패턴을 대응시키는 방식으로 영상의 크기 또는 해상도보다 보다 도트 해상도가 좋은 흑백 레이저 프린터에서 많이 사용되는 방식으로 랜덤 디더링 보다 출력물의 질은 뛰어난 편이며 오류확산 디더링보다 처리속도가 빠르다. 순서상 디더링 방식은 패턴 디더링을 확장한 개념으로 이 방식의 대표적인 알고리즘으로는 베이어(bayer) 디더링<sup>[1]</sup> 방식이 있다. 이 베이어 방식은 비교적 많이 사용되며 그 출력물의 질도 뛰어나나 크로스해치 패턴 형태의 아티팩트(artifacts)가 발생할 수 있는 단점이 있다. 오류 분산 방식은 현재 영상의 화소의 값과 할당된 값의 차이를 오류로 하여 그 오류를 다음 스캔할 화소 및 다음 줄의 인접 화소들에 분산시키는 방식으로 대표적인 알고리즘으로는 Floyd-Steinberg 방식이 있다.<sup>[2]</sup> 오류분산 디더링은 네가지의 디더링 방식중에 우수한 화질을 보여주며 일반적으로 많이 사용된다. 그러나 출력되는 영상이 방향 강조 현상이 발생할 수 있으며 이러한 방향성을 제거하기 위하여 약간의 잡음을 영상에 삽입하던가 또는 섭동(perturbation)을 주게 된다.

소형 단색 영상표시 장치를 가진 휴대단말기 또는 PDA에 칼라 또는 제조 영상을 효율적으로 표시하기 위해서는 단순한 이진화가 아닌 디더링이 필요하다. 원 영상을 디더링할 수 있는 많은 소프트웨어들이 나와 있거나 오프라인용으로 데이터 또는 영상서버에서 원영상을 PDA 또는 휴대전화 단말기로 전송할 경우 자동적으로 흑백 디더링을 해결 필요성이 있다. 본 연구에서는 서버에서 정지 또는 동영상정보를 보내기 위한 디더링 방식으로, 개선된 Floyd-Steinberg 디더링 알고리즘을 제시하며 그 유효성을 실험으로 보인다.

## II. 개선된 Floyd-Steinberg 디더링

### 2.1 Floyd-Steinberg 방법

오류 분산 디더링 알고리즘인 Floyd-Steinberg 방식은 현재 화소의 오류를 다음 스캔하는 인접 화소들에 전파하는 방식으로 전체 분배되는 오류의 합은 1이 되도록 하고 있다. 영상을 왼쪽에서 오른쪽으로 또 위에서 아래로 순차적으로 스캔한다면 오류전파는 그림 1과 같다. 그림에서 Error는 다음 식과 같다.

$$\text{Error} = I(x,y) - I_d(x,y) \quad ; \quad I(x,y) \text{는 현재 화소 값, } I_d(x,y) \text{는 결정된 화소값}$$

즉 현재화소의 값과 현재화소에 가장 가까운 디더링색의 값과의 차이를 오류로 하여 일정 비율로 분배를 하고 있다. 분배 비율은 같은 라인의 오른쪽 인접 화소에  $7/16 * \text{Error}$ 를 다음 라인의 같은 위치 즉 바로 아래쪽의 화소는  $5/16 * \text{Error}$ 의 값을 분배하므로 수평 수직 방향의 오류 전파 강조성을 띠게된다. 일반적으로 각각의 방향으로 오류를 전파하는 비율 또는 필터값 그리고 오류의 영향을 받는 인접영역의 크기에 따라 단 순필터, Floyd-Steinberg필터, Burki 필터, Stucki필터

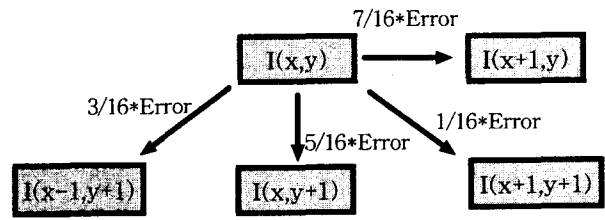


그림1 Floyd-Steinberg의 오류분배도

등으로 나누어지며 일반적으로 Floyd-Steinberg 필터가 실제 응용에서 많이 적용되고 있다. 그러나 오류전파의 수평, 수직 방향 강조성 및 넓게 퍼진 밝은 영역에서 바람직하지 않는 패턴 발생 가능성을 가지고 있다. 이러한 수평, 수직 방향 강조성을 줄이기 위해 지그재그 스캔을 사용하며 또 원치 않는 패턴의 발생을 억제하기 위하여 임의로 섭동을 가하기도 한다.

### 2.2 개선된 Floyd-Steinberg 방법

본 연구에서 제시하는 방법은 이러한 방향 강조성을 개선하기 위하여 현재 위치에서 디더링색의 값이 흑인 경우에는 분배하는 오류 비율을 시계방향으로 회전하게 하며 디더링색의 값이 백인 경우에는 반시계 방향으로 회전시킨다. 이렇게 하므로 임의의 섭동을 주는 것과 지그재그 스캔을 하는 효과 두가지를 동시에 얻을수 있다. 그림 2는 현재 화소의 값이 흑으로 결정된 경우에 그림 1에서 시계방향으로 회전된 오류 분배

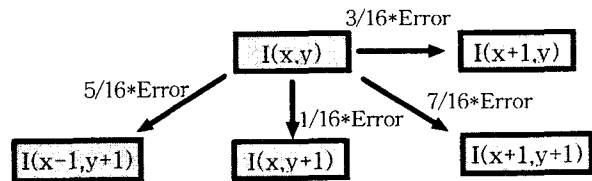


그림 2 현재화소의 디더링색의 값이 흑인 경우 전환된 분배비율

비율을 보여주며 그림 3은 현재 화소가 백으로 결정된 경우에 반시계 방향으로 회전한 오류 분배비율을 보여 준다.

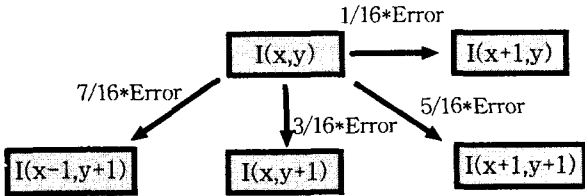


그림 3 현재화소의 디터링 색의 값이 백인 경우 반시계방향으로 전환된 분배비율

### III. 실험

제시된 방법의 유효성을 검증하기 위하여 영상에 적용시켜보았다. 첫 번째 영상은 흑에서 백으로 점진적으로 변하는 계조영상을 사용하였다. 그림4는 원 계조 영상으로 보기에 흑백계조이지만 24bit 컬러 영상이다. 그림 5는 Floyd-Steinberg 방법을 사용한 디터링 결과이고 그림 6은 순서상 디터링 중에 Bayer 방식을 적용시켜 보았으며 그림 7은 본 논문에서 제시하는 개선된 Floyd-Steinberg 디터링에 의한 결과이다.

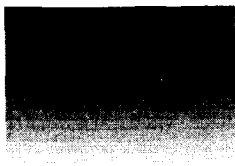


그림 4 흑백계조 원 영상

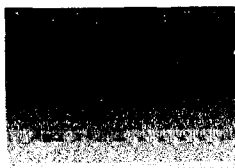


그림 5 Floyd-Steinberg 디터링 영상

현재 화소의 결정된 디터링의 값에 따라 분배비율을 임의의 방향으로 전환시킨 경우에 비전환시보다 방향 강조성이 줄어들었으며 처리된 영상의 효율성이 증가하는 것을 볼수 있으며 순서상 디터링에 비하여 규칙적

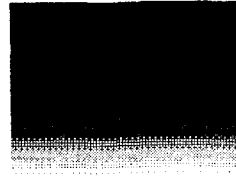


그림 6 Bayer 디터링 영상



그림 8 개선된 Floyd-Steinberg 영상

인 패턴 형성이 적은 것을 볼수 있다.

두 번째 영상은 일반적으로 영상처리 표준영상으로 사용되고 있는 lena영상으로 그림 8은 원영상을 나타



그림 8 lena 영상



그림 9 Floyd-Steinberg 디터링 영상

내며 그림 9는 Floyd-Steinberg 디터링을 그림 10은 Bayer 디터링을 그림 11은 개선된 Floyd-Steinberg 방



그림 10 Bayer 디터링 영상



그림 11 개선된 Floyd-Steinberg 영상

식을 적용한 경우이다. 그림에서 보면 확실히 순서상 디터링보다 오류확산 디터링 방식이 자연스럽게 원영상을 표현할 수 있는 것을 알수 있으며 또 오류확산 디터링에서도 인공적인 문양이 발생하는 것은 피할 수 없으며 이러한 인공적인 문양은 오류를 더욱 넓은 범위에 걸쳐 전파하므로 줄일 수 있으며 본 논문에서 제시하는 개선된 Floyd-Steinberg 방법을 사용하여 줄일 수 있으나 임의의 영상에서도 완전히 제거할 수는 없는 것으로 보인다.

#### IV. 결론

PDA나 휴대전화기와 같은 단색 디스플레이를 쓰는 휴대장비에서 원색영상을 표현하기 위해서는 원색영상을 단색 영상으로 변화시켜주는 디터링 과정이 필요하며 일반적으로 Floyd-Steinberg 디터링 알고리즘이 주로 많이 사용되고 있다. 그러나 이 알고리즘은 수평수직 방향 강조성 및 밝은 영역에서의 인공문양등이 발생할 수 있다. 본 논문에서 제시하는 개선된 Floyd-Steinberg 알고리즘은 그러한 단점들을 완화시킬 수 있는 것으로 나타났으며 그 유효성을 실험으로

보였다. 앞으로의 연구는 임의성을 증가시켜 가능한 인공문양을 더 제거할수 있고 또 원 영상 표현의 효율성을 높이도록 해야 한다.

#### 참고문헌(또는 Reference)

- [1] B. E. Bayer, "An Optimum method for Two-Level Rendition of Continuous-Tone pictures", Proc. IEEE Int. Conf. Commun., pp. (26-11)-(26-15), 1973
- [2] R. W. Floyd and L. Steinberg, "An adaptive algorithm for spatial grayscale", proc. SID, Vol.17/2, pp.75-77, 1976.
- [3] R. Ulichney, "The void-Cluster method for Generating Dither Arrays", "IS&T/SPIE Symposium on Electronic Imaging Science & Technology, San Jose, CA, vol. 5, no. 2, pp. 9-18, 1993.
- [4] Daniel L. Lau, "Green-Noise Digital halftoning", Proc. IEEE, vol. 86, no. 12, pp. 2424-2442, Dec 1998.
- [5] Daniel L. Lau, "Robust halftoning with Green noise", 1999 Image processing, Image Quality, Image Capture system Conference, Savannah, Ga. USA, April 26-28, 1999.