

0.1% 메모리 비트 결함율에서 칩 면적 요구량이 적은  
영상 데이터 오류 복원 방법

이강<sup>0</sup> 옹환기, 정경훈\*

한동대학교 전산전자공학부, \*국민대학교 전자정보통신공학과  
{yk<sup>0</sup>, wyong}@handong.edu, \*khjung@kookmin.ac.kr

An Area-Efficient Error Recovery Method for Image Data Memory  
at 0.1% Memory Bit Error Rate

Kang Yi<sup>0</sup>, Whanki Yong, Kyeong Hoon Jung\*

School of Computer Science and Electrical Engineering, Handong Global University  
Department of Electrical Engineering, Kookmin University \*

요 약

멀티미디어 응용 환경에서 데이터 메모리에 오류가 있는 경우에 이를 복원시키는 간단하고 효율적인 영상 복원 방식을 제안한다. H.264 복호화기의 예를 이용하여 최대 BER=0.1%인 메모리 결함율의 환경에서 본 논문에서 개발한 방식의 효율성을 기존의 메모리 결함 복원 방식과 비교하였다. 실험 결과 제안된 방식에 의하여 복원된 영상의 질은 거의 원본과 같은 수준으로 뛰어나다. 본 방식을 구현하기 위하여 추가된 하드웨어가 소비하는 전력은 전체 시스템 전력량의 약 0.05%이지만, 면적 요구량은 기존 방식의 1/56 이하에 불과하다. 따라서, 멀티미디어 제품 설계에 제안된 방식을 사용하면, 결함이 존재하는 메모리를 재활용할 수 있게 됨으로써 관련 제품의 질을 유지하면서도 단가를 낮출 수 있게 될 것으로 기대된다.

1. 서론 및 연구 배경

최근들어 가장 수요가 늘어나는 디지털 기기류 중의 하나가 멀티미디어 응용 분야이다. MPEG2, H.264 등의 동 영상 압축 및 재생 기능이 많은 제품의 수요가 급격히 늘고 있다. 그런데, 이들 응용은 데이터 메모리의 요구량이 많은 것이 특징이다[1]. HDTV의 예를 들면, 영상의 크기가 1920 x 1080 이고 영상이 4:2:0 모드로 인코딩되고 2개의 프레임버퍼가 사용된다고 가정하면, 필요한 메모리 최소 데이터 크기가  $1920 \times 1080 \times 1.5 \times 2 \times 8 = 48.6$  Mbits 이다. [2]와 [3]에 따르면, 현재 90 nm 공정에서 SRAM 메모리의 한 셀의 크기가  $1.33 \mu\text{m}^2$  이고 주변 인터페이스 회로로 칩면적이 60% 정도 더 늘어나므로 48.6 Mbits의 메모리는 그 면적이  $100 \text{mm}^2$ 가 넘는다. 따라서, 앞으로 이러한 멀티미디어 응용을 위한 시스템을 단일 칩 시스템으로 구현하고자 할 경우 내장 메모리가 칩에서 차지하는 면적의 비율이 급격히 늘어나게 될 것이다 [1]. 칩 내의 메모리 양이 늘어나면서 코어 로직과 메모리를 한 칩으로 집적하는데 따르는 어려움은, 칩의 집적도를 높이기 위해서 나노 공정을 사용하여 칩을 제조할 경우 내장 메모리의 제조상의 결함 확률이 급격히 높아지기 때문에 칩의 수율이 떨어진다는 것이다. 그 이유는, 메모리 설계에서는 면적을 줄이는 것이 중요하기 때문에

일반 논리 회로에 비해서 더 공격적인 설계 규칙을 적용하기 때문이다. 제조사의 메모리 블록에 발생한 결함 때문에 전체 시스템을 쓰지 못하게 될 경우가 많이 발생할 것이고 이는 제품의 단가의 상승으로 이어질 것이다.

따라서, 장래의 멀티미디어용 단일 칩 시스템 설계의 고민은 어떻게 메모리의 결함을 적절히 보상할 수 있는지가 될 것이다. 생산시에 발생하는 메모리 블록의 결함을 복원시키는 기존의 방법들은 여분의 메모리 블록을 이용하는 방식이다. 기존 방식들은 메모리 설계 시에 여분의 행/열을 추가하고 생산 후에 테스트를 거쳐서 결함 위치를 찾아내고 레이저 빔 등을 이용하여 퓨즈를 단락시키거나 참조표(Lookup Table)를 사용해서 결함이 발생한 메모리의 셀이 있는 지역의 전체를 행이나 열 단위로 여분의 메모리 셀들로 대체하게 하는 것이다. 이 방식은 평균 메모리 비트의 결함 확률(BER)에 따라서 10%에서 70%까지의 메모리의 면적이 추가로 필요하게 되는 단점이 있다[4]. 한편, 현재의 반도체 공정상에서는 0.001% 정도의 오류만 대처할 수 있어도 문제없지만 가까운 장래의 나노 공정에서는 메모리 오류율이 0.1% 까지도 늘어날 것으로 예상된다[7]. 따라서, 기존 방식의 더욱 큰 문제는, 메모리 결함율이 0.001% 넘어서면 너무나 큰 면적 오버헤드 때문에 현실적으로 적용이 어렵게 된다는 것이다.

본 논문에서는 결함 메모리 셀을 다른 정상 셀로 대체하여 완벽하게 고치려는 기존 방식과는 달리 영상 데이터 자체의 특성을 이용하여 결함이 있는 메모리 블록을 그대로 사용하면서도 좋은 화질로 복원하는 방식을 제안한다. 제안된 방식은, 결함 셀이 있는 메모리를 그대로 시스템 구성의 한 부품으로 사용하되, 메모리의 회로 단계가 아닌 응용 단계에서 데이터의 오류를 복원시키고자 한다. 즉, 영상 데이터에 포함된 중첩된 정보(예컨대, 공간적 시간적 유사성)를 이용한 간단한 영상 필터를 적용하여 결함이 발생한 화소의 값을 추정하는 방식이다.

다음 2장에서는 제안된 방법을 적용하기 위한 제안된 시스템의 구성도와 개념도를 보이고, 3장에서 이에 맞는 필터를 설계하고 4장에서는 이를 H.264 복호화기를 대상으로 실험하여 그 성능 및 비용을 분석한다. 끝으로 5장에서는 요약 및 향후 연구 과제를 제시하고 논문을 맺는다.

2. 문제 정의 및 새로운 방식의 개념

본 논문에서 제안된 방식은 메모리의 결함 위치의 주소를 Defectmap이라는 별도의 외부 메모리에 저장하여 둔다고 가정한다. Defectmap 메모리의 정보 기록은 처음 메모리 칩 제조 후 또는 시스템의 (재)부팅 시에 메모리 시험 과정을 통해서 결함이 존재하는 메모리 셀을 찾아내고 그 위치를 Deectmap에 기록한다고 가정한다. 이러한 부가적인 작업은 기존 메모리 결함 복원 방식에서도 사용되므로 본 방식 고유의 추가 비용이라고 보기는 어렵다. 매 영상 화소 데이터 읽기 동작 시에 이 Defectmap을 참조하여 결함 셀인지 여부를 판단하고 결함 셀일 경우에는 영상 필터 실행 등의 별도의 과정을 통해서 바른 화소값을 추정한다.

다음 그림 1은 제안된 방식의 개념도이다. 그림1에서 Data memory 블록은 이전의 복호화된 프레임 영상 정보가 YUV 값으로 저장되는 메모리이다.

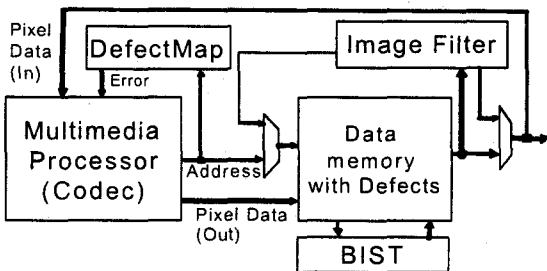


그림 1. 제안된 영상 메모리 오류 복원 방식의 개념도

H.264 복호화기는 입력된 압축 영상 스트림 데이터를 역양자화와 IDCT를 거치고 Motion Compensation과정을 통해서 원래 영상 프레임값을 계산한다. 복호화 과정의 Motion Compensation 시에 이전 프레임의 화소 값을 프레임 버퍼에서 읽을 필요가 있다. 이 메모리 읽기 동작에서 화소 데이터가 저장된 메모리 셀이 결함이 있는 자리임이 DefectMap에 의해서 알려지면 Image Filter 블록에 의해서 계산된 값을 화소값으로 사용한다. 영상 필터의 계산을 위해서 오류가 난 주변 화소들의 값을 사용할 필요가 있기 때문에 image Filter블록에서 데이터 메모리를 직접 읽을 수 있도록 되어 있다. BIST 블록은 메모리의 결함 위치를 시스템 부팅시에 찾아내는 기능을 한다.

DefectMap은 CAM (Content Addressable Memory)를 이용해서 구현된다고 가정한다. 그림 2는 Defectmap의 구성을 상세히 보여준다. CAM 메모리인 Defectmap에는 결함이 존재하는 메모리의 주소가 저장되어 있고 메모리 참조 때마다 이 DefectMap을 동시에 참조하여 주어진 주소와 일치하는 값이 저장되어 있으면 match 신호가 '1'이 되어서 결함 위치임을 Codec 블록에 알려주게 된다. Match 신호가 '1'이 아닌 경우는 데이터 메모리에 저장된 값이 화소값으로 사용된다.

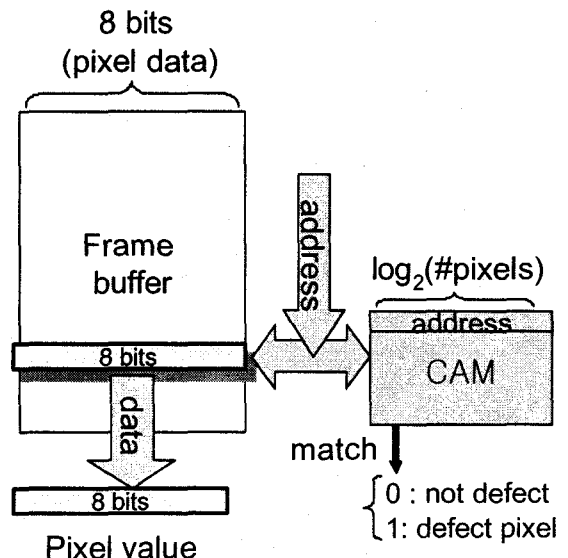


그림 2. DefectMap과 데이터 메모리의 상세도

그림 3은 메모리 오류 복원을 위한 기존 방식과 본 논문에서 제안된 새로운 방식을 메모리 면적 오버헤드 면에서 비교한 그래프이다. Defectmap의 프레임 버퍼에 대한 면적 오버헤드는 다음 식들에 의해서 계산된다.

$$OVR = PER \times (AddrBits / 8) \times (9 / 6)$$

$$AddrBits = \log_2(\# Pixels / Frame \times 1.5 \times \# Frame)$$

$$PER = 1 - (1 - BER)^{PixelBits}$$

위 식에서, (9/6)은 CAM 메모리와 SRAM 메모리의 셀 크기의 비율이고 (AddrBits/8)은 CAM과 SRAM의 워드 크기의 비율이다. PixelBits은 한 픽셀 당 비트 수이다. 여기에서는 PixelBits=8을 사용한다. PER은 화소 결함율이다.

그림 3을 보면, 주어진 메모리 비트 결함율(BER)에 따라서 양쪽 방식 모두 면적 오버헤드가 늘어나지만, 기존 방식이 훨씬 더 급격히 늘어난다. 제안된 새로운 방식은 0.1% 메모리 비트 결함율에서 기존 방식에 비해서 매우 적은 면적을 사용함을 알 수 있다.

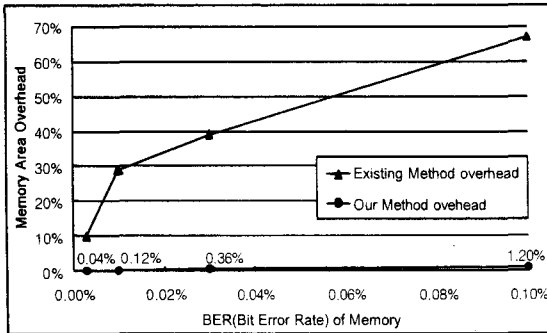


그림 3. 메모리의 오류를 정정 방식들의 면적 요구 비교

### 3. 복잡도가 낮은 영상 필터의 설계

본 제안된 방식은 면적 오버헤드가 적게 들기 때문에 기존 방법에서 처리가 어려운 비교적 높은 메모리 에러율 (BER=0.1%)에서도 사용이 가능하다. 단, 성능 및 전력 소모면에서 오버헤드가 적은 영상 필터가 필요하다. 이 특수한 영상 필터는 다음과 같은 조건을 만족해야 한다.

- (1) 일반 영상 필터의 경우에 비해서는 낮은 에러율에서 작동한다 (최대 0.1% 비트 결함율 또는 최대 0.8% 화소값 결함율).
- (2) 일반 영상 필터와는 달리 에러가 한군데 집중되지 않고 골고루 임의 분산되어 존재한다.
- (3) 메모리의 크기가 주로 큰 곳이 YUV 영상 정보 저장 소이므로 본 영상 필터는 YUV 값 영역에서 작동한다.
- (4) 본 영상 처리 알고리즘은 결함이 있는 메모리의 위치를 Defectmap을 통해서 정확히 알고 있다.
- (5) 이 영상 필터의 알고리즘은 복잡도가 현저히 낮아서 성능 및 전력 소모에 거의 영향을 주지 않아야 한다.

위의 조건들을 만족하는 가장 단순한 영상 필터는 결함이 있는 픽셀의 좌우 (또는 상하)의 값의 평균을 구하는 mean2 필터이다. 그림 4는 mean2 필터가 9가지 상황에서의 화소 값을 계산하는 방식을 보여준다. 그림 4에서 어두운 부분이 결함이 있는 화소의 위치이고, 흐린 빗금으로 표시된 사각형은 화소가 존재하지 않는 영상의 경계면 밖을 나타낸다. 각 경우에 두개의 화소값은 결함이 있는 화소의 값을 추정하기 위해서 사용되는 2개의 이웃 화소들을 나타낸다. 대부분의 경우는 좌우의 두 화소값을 사용하지만, 영상의 모서리나 양측면에서는 결함 화소의 상하에 있는 화소들의 값을 이용하기도 한다.

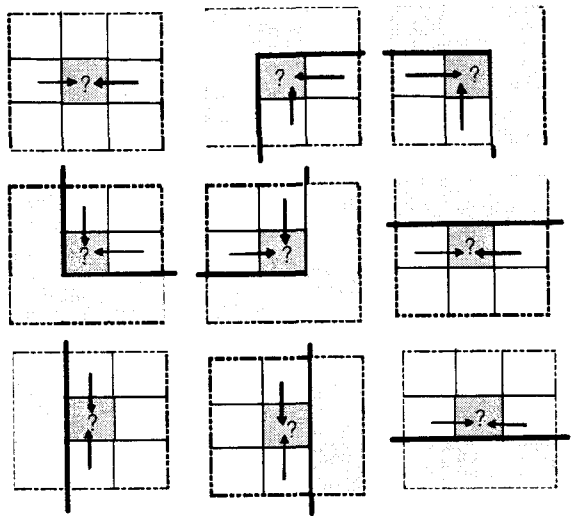


그림 4. mean2 필터의 여러 상황에서의 픽셀 값 계산법

다음은 mean2 알고리즘을 pseudo code로 서술한 것이다.

```

Function mean2 (pixel [x][y]) begin
  Y1 = y; Y2 = y; X1 = x-1; X2 = x+1;
  if(x == 0 and Y == 0) then
    Y1 = Y+1; X1 = x;
  else if (x == 0 and Y == MAXY) then
    Y1 = Y-1; X1 = x;
  else if (x == 0 and 0 < y < MAXY) then
    Y1 = Y-1; X1 = x; Y2 = Y+1; X2 = x;
  else if (x == MAXX and Y == 0) then
    Y2 = Y+1; X2 = x;
  else if (x == MAXX and Y == MAXY) then
    Y2 = Y-1; X2 = x;
  else if (x == MAXX and 0 < Y < MAXY) then
    Y1 = Y-1; X1 = x; Y2 = Y+1; X2 = x;
  end if
  return (pixel[X1][Y1]+pixel[X2][Y2])/2
End Mean2
    
```

4. 실험 결과 및 분석

본 논문에서는 0.1% 이하의 메모리 비트 결함을 상황에서 제안된 시스템의 효율을 분석하기 위해서 H.264 복호화 소프트웨어 사용하여 시뮬레이션을 수행하였다. 실험용 데이터는 QCIF 크기의 foreman 동영상이다. 이 영상은, QP=28, 참조 프레임의 수는 2개, 1 프레임은 매 60프레임마다 한번씩 나타나고, 인코딩된 프레임 타입의 순서는 IBPBP...IBPBP.. 형태로 압축되었다. 실험은 0.003%, 0.01%, 0.03%, 0.1%의 메모리 비트 결함에 대해서 실시되었다. 시뮬레이션은 다음 과정으로 진행되었다.

- (1) Defectmap을 주어진 오류율에 따라서 임의로 생성하거나 외부에서 읽어들인다.
- (2) H.264 복호화 과정에서 모든 화소의 프레임 버퍼 읽기 동작에 대해서 DefectMap으로 결함 화소인지 점검한다.
- (3) 결함 화소에 대해서는 정해진 결함값을 프레임 메모리에 삽입한다 (저장된 화소의 값을 변경함)
- (4) 결함 화소의 값을 영상 필터(mean2 필터)를 이용해서 추정 복원시켜서 사용한다 (버퍼에 저장하지는 않는다).

Defectmap과 영상 필터를 이용해서 복원된 영상의 질을 객관적으로 측정하기 위해서 PSR 값을 사용한다. PSNR 값은 영상 처리 분야에서 가장 보편적으로 사용되는 객관적인 영상의 질의 척도이다. PSNR값은 다음 식에 따라서 H.264로 압축되기 전의 원본 영상에 대한 비교로 얻어진다.

$$MSE = \frac{\sum [f(i, j) - F(i, j)]^2}{N \times M}$$

$$PSNR = 20 \log_{10} \left( \frac{255}{\sqrt{MSE}} \right)$$

위 식에서 f(i,j)는 복원된 영상의 화소 값이고 F(i,j)는 비교 대상인 원본 영상의 화소 값이다. N과 M은 영상의 크기로서 영상의 가로 및 세로 화소수이다.

그림 5에서 그림7에서 Mean2 필터를 적용한 결과 영상의 PSNR 값을 필터를 적용하기 전의 PSNR과 비교하였다. 그림 5는 Lumma 성분인 Y에 대한 비교이고, 그림 6과 그림 7은 모두 Chroma 성분인 U와 V 성분에 대한 비교이다. 그림5에서 그림7의 그래프에서 알 수 있듯이 밝기 성분인 Y는 메모리 결함에 민감하지만 컬러 성분인 U와 V성분은 결함에 덜 예민한 것을 알 수 있다.

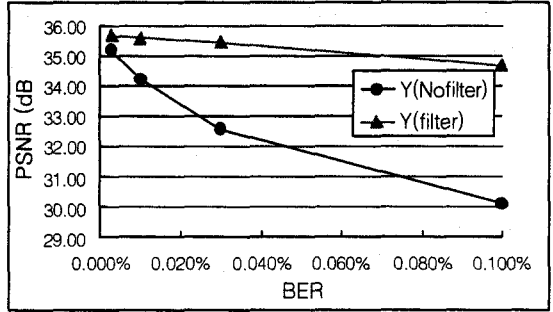


그림 5. 필터 적용에 의한 영상 질의 향상 (Y 성분)

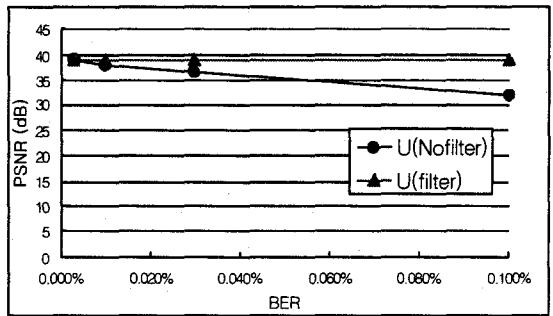


그림 6. 필터 적용에 의한 영상 질의 향상 (U 성분)

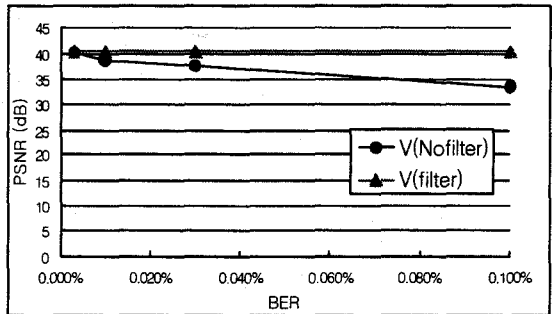


그림 7. 필터 적용에 의한 영상 질의 향상 (V 성분)

그림 8에서 그림10은 영상 필터 적용 전과 후의 주관적 화질의 차이를 보여준다. 시범 동영상의 전체 398개의 프레임 중 오류가 가장 심한 58번째 프레임 캡처 화면이다.

3장에서 제안된 매우 간단한 영상 필터를 적용한 결과는, BER=0.1%의 경우에도 매우 뛰어난 것을 알 수 있다. 비트 오류가 0.1% 개입된 후 필터에 의해서 복귀된 그림 11과 전혀 오류가 삽입되지 않은 영상인 그림9를 비교해 보면 거의 차이를 발견할 수 없다. 따라서, 본 논문에서 제안된 방식으로 메모리의 오류를 복귀한 결과의 영상은 주관적 객관적인 화질이 매우 우수함을 알 수 있다.



그림 8. BER=0.0%의 영상 (복호화 결과)



그림 9. BER=0.1%의 복호화된 이미지 (필터 적용전)



그림 10. BER=0.1% 복호화된 영상 (필터 적용 후)

이제, 질문은 과연 이런 결과를 얻기 위해서 어떤 비용이 얼마나 필요한가이다. 면적 비용은 각 방식에서 요구하는 메모리 면적 오버헤드로 비교하였다. 그림 11은 그림 3을 로그 스케일로 다시 그린 것이다. 그림 11에서 막대 그래프는 두 가지 방식의 면적 오버헤드를 나타내고 있고, 점선은 기존 방식과 새로운 방식의 오버헤드 비율을 나타내

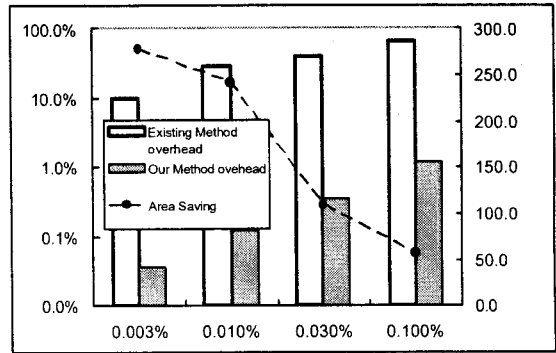


그림 11. 기존 방식과 새로운 방식의 면적 오버헤드의 비교

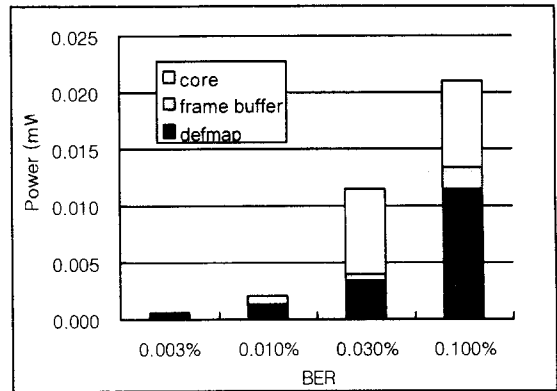


그림 12. 제안된 방식 (영상 필터와 Defectmap)에 의해서 소비되는 전력 소모량의 구성비

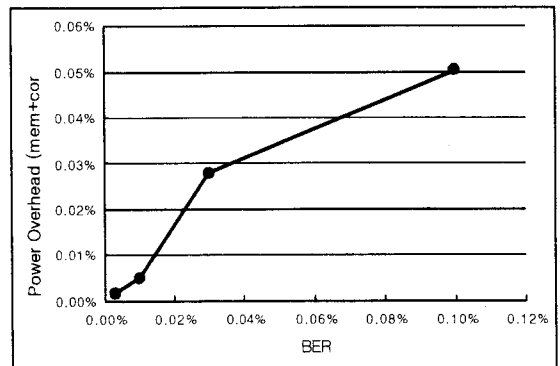


그림 13. 새로운 메모리 오류 복원 방법에 의해서 요구되는 전력소모의 전체 시스템의 소비 전력에 대한 오버헤드  
고 있다. 새로운 방식은 기존 방식에 비해서 불과 1/56배 ~ 1/270 배의 매우 적은 메모리 면적이 소요된다.

새로운 방식의 복잡도의 분석은 전력 소비량으로 비교하였다. 본 논문에서 제안된 영상 필터를 90 nm 기술 공정에서 하드웨어로 구현했을 경우의 전력 소비량을 분석한다. 그림 12는 새로운 방식에 의해서 요구되는 전력량의 구성 비율을 각 메모리 비트 결함율에 따라 보여준다. 대개의 경우, 메모리 (Defectmap과 프레임버퍼 참조)에 의한 전력 소모가 코어 로직 회로에 의한 것보다 조금 많은 것을 알 수 있다. 새로운 방식에 의한 메모리 결함 복원에 필요한 전력 소모 오버헤드를 계산하기 위해서 기존 시스템 소비 전력으로 [6]의 H.264 칩의 전력 소비량(QVGA, 15 fps에서 63mW)를 참고하였다. 오버헤드 계산시에는 [6]의 H264 복호화기의 전력 소비량을 영상 크기가 QCIF이고 30 fps인 상황에 맞도록 선형적으로 스케일링하여 비교하였다. 그림 13에서는 본 논문에서 제안된 메모리 오류 복원 방식에 의해서 추가로 요구되는 전력 소비 오버헤드를 [6]을 기준으로 보여주고 있다. 그림 13의 분석에 따르면, 가장 많은 전력이 요구되는 BER=0.1%의 경우이라도 전체 전력 소모의 0.05% 만이 추가로 요구된다. 따라서, 제안된 방식에 의한 전력 오버헤드는 거의 무시할 만한 수준임을 알 수 있다.

### 5. 결론

본 논문에서는 멀티미디어 영상 응용 데이터 자체의 특성을 이용하여 칩 제조과정에서 발생한 데이터 메모리 셀의 결함을 회로 단계가 아닌 응용 단계에서 복원시키는 새로운 방식을 제시하였다. 제시된 방법에 의해서 복원된 영상의 질을 정량적인 방식과 주관적인 방식으로 모두 검사한 결과 0.1% 메모리 오류에서 거의 완벽한 화질을 보여주었다. 0.1% 메모리 비트 에러율은 현재 기술 수준보다 훨씬 앞선 공정의 메모리 결함율이므로 현재는 물론 가까운 미래에도 이 방식은 실용적으로 사용하기에 아무런 문제가 없을 것이다. 본 방식은 이러한 우수한 오류 복원 능력을 보이면서 기존의 메모리 결함 복원 방식에 비해서 칩의 면적 오버헤드는 1/56 이하로 요구되고, 전력 소모는 논문으로 발표된 H.264 칩의 소비 전력을 기준으로 0.05% 이하의 오버헤드가 요구된다.

본 연구 결과를 이용하면 무시할 만한 전력 오버헤드와 기존 방식보다 훨씬 더 적은 메모리 오버헤드로 오류가 있는 멀티미디어 데이터 메모리 블록을 재사용할 수 있게 된다. 따라서, 본 연구 결과를 상용 H.264 복호화기 등의 멀티미디어 제품의 영상 데이터 저장 메모리 설계에 응용할 경우 제품의 생산 단가를 크게 낮추는데 기여할 것으로 기대된다.

한편, 본 연구는 SRAM 메모리와 QCIF이미지에 대해서만 실험이 이루어졌으나 DRAM 등의 다른 형태의 메모리와 HDTV급의 대형 영상의 응용에도 확장 적용이 가능한 원리이다. 현재는 45 nm 이하의 초미세 나노 공정시의 높은 메모리 결함율에서의 메모리 데이터 복원방식에 대한 연구가 진행 중이다.

### 참고문헌

- [1] Shoukourian, S.; Vardanian, V.; Zorian, Y.: SoC yield optimization via an embedded-memory test and repair infrastructure. Design & Test of Computers, IEEE Volume 21, Issue 3, May-June 2004. pp. 200 ? 207
- [2] S. Prihar et. Al. " A High Density 0.10um CMOS Technology Using Low-L Dielectric and Copper Interconnect. " Proc. 2002 IEDM.
- [3] <http://public.itrs.net>
- [4] L. Anghel, N. Achouri, M. Nicolaidis. "Evaluation of Memory Built-in Self Repair Techniques for High Defect Density Technologies," Proc. of 10th IEEE Pacific Rim International Symposium on Dependable Computing (PRDC'04), 2004, pp. 315-320
- [5] F. J. Kurdahi, A. M. Eltawil, Y.-H. Park, R. N. Kanj, S. R. Nassif, "System-Level SRAM Yield Enhancement " , Proceedings of the 7th International Symposium on Quality Electronic Design, (ISQED 2006), 2006. pp. 179 ? 184.
- [6] T. Fujiyoshi et al. " A 63-mW H.264/MPEG-4 Audio/Visual Codec LSI With Module-Wise Dynamic Voltage/Frequency Scaling " . IEEE Journal. Of Solid State Circuits, vol. 41, No 1, Jan, 2006, pp. 54-62.
- [7] T. Gupta, A.H. Jayatissa, "Recent advances in nanotechnology: key issues & potential problem areas," Proceedings of IEEE Conference on Nanotechnology, Vol. 2, , 2003, pp. 469 ? 472.