

# 행/열 단위 증분 부호화를 이용한 JPEG 압축 기법 확장에 관한 연구

박대현<sup>o</sup>안영훈<sup>†</sup>신현준<sup>†</sup>위영철<sup>†</sup>큐램(주)<sup>o</sup>아주대학교<sup>†</sup>

daely@quram.com

{amouse,joony,ycwee}@ajou.ac.kr

## On Extending JPEG Compression Method Using Line-based Differential Coding

Daehyun Park<sup>o</sup>Younghoon Ahn<sup>†</sup>Hyun Joon Shin<sup>†</sup>Young Cheul Wee<sup>†</sup>Quram Co.<sup>o</sup>Ajou University<sup>†</sup>

### 요약

본 논문은 영상의 손실 압축에 널리 사용되는 JPEG 표준을 확장하여 압축률을 향상시키기 위한 방법을 제안한다. 연속적인 신호를 압축하는데 널리 사용되는 증분 압축 기법을 보다 효과적으로 손실 영상 압축 방법에 적용하기 위하여 행이나 열 단위로 예측(prediction)과 이산 코사인 변환을 수행하고 양자화, 엔트로피 부호화 등의 과정을 거쳐 영상을 압축하기 위한 방법을 제안한다. 특히 JPEG 표준과 행/열 단위 압축 방법의 장점을 동시에 갖게 하기 위해 JPEG 표준의 틀에서 선택적으로 블록에 행/열 단위 압축을 수행함으로써 압축률을 향상시킨다. 실험 결과 본 논문에서 제안하는 압축 방법은 사용자가 압축 시 지정한 화질이 높을 경우 JPEG보다 높은 압축률을 나타내는 것을 확인할 수 있다. 따라서 본 논문에서 제안하는 압축 방법은 간단하게 기존의 JPEG 압축 방식에 추가되어 고화질의 영상을 효율적으로 압축하는데 사용될 수 있다.

### Abstract

In this paper, we introduce a novel method to extend the JPEG standard, which is the most widely used for lossy image compression, in order to improve compression ratio. To employ two of the most successful methodologies for the data compression: differential coding and quantization simultaneously, we propose a line-based approach. For each line in a block, we apply one-dimensional discrete cosine transformation to the increments instead of the pixel values. Those values are quantized and entropy-coded similarly to the JPEG standard. To further increase compression ratio, the proposed method is plugged into the JPEG standard to form a new compression method, in which the proposed method are applied to only selected JPEG blocks. In our experiment, we found that the proposed method outperform the JPEG standard when the qualities of the coded images are set to be high. We believe the proposed method can be simply plugged into the standard to improve its compression ratio for higher quality images.

키워드: 영상 처리, 영상 압축

Keywords: Image Processing, Image Compression

## 1 서론

JPEG(Joint Picture Expert Group)에서 제안한 압축 기술은 현재 세계적으로 가장 널리 사용되고 있는 정지영상의 손실 압축 방법이라 할 수 있다 [1]. JPEG은 특히 사람의 눈에 잘 띄는 부분은 정밀하게 표현하고 눈에 잘 띄지 않는 부분의 정보를 효과적으로 제거함으로써 높은 압축률을 가질 때에도 비교적 높은 화질을 유지할 수 있는 손실 영상 압축 방법으로 알려져 있다[2, 3]. 또한 JPEG의 중요한 특징 중 하나는 빠른 부호화 및 복호화 시간을 들 수 있다. JPEG 표준에서 허용하고 있는 다양한 엔트로피 부호화 방법 중 고정 표에 의한 허프만 부호화(Huffman coding with fixed-table) 방식은 적절한 수준의 압축율을 유지하면서도 매우 빠르게 부호화와 복호화가 가능하기 때문에 널리 사용된다[4].

본 논문에서는 JPEG의 특징을 가급적 유지하면서 압축률을 증가시키는 방법을 제안한다. 일반적으로 영상과 같이 공간

적인 상관 관계(spatial coherency)를 가지는 매체(media)나 동영상, 음향과 같이 시간적 상관 관계(temporal coherency)를 가지는 매체를 부호화 하는 과정에는 증분 부호화(differential coding) 방식이 널리 사용된다[5]. 증분 부호화는 공간적, 혹은 시간적으로 가까이 있는 정보가 유사한 특징을 나타낸다는 점에 착안하여, 실제 신호를 저장하는 대신 인접한 정보를 이용하여 저장할 값을 예측(prediction)하고 실제 값과 차이만을 저장하는 방식이다. 예측이 정확하면 정확할수록 저장해야 하는 값의 절대값이 작아지고 전체적으로는 작은 값들의 빈도가 높아진다. 따라서 특정한 값들의 빈도가 높은 경우 압축률이 향상되는 엔트로피 부호화의 특성상 압축률이 높아지게 된다.

JPEG 방법에 직접적으로 증분 부호화를 적용하는 데는 한계가 있다. 손실 압축의 경우 예측을 위해 주변 값을 참조할 때 참조하는 값은 복호화 하는 과정에서도 복원할 수 있는 값이어야 한다. 따라서 손실 압축의 경우에는 손실 과정을 포함하여 부호화와 복호화가 완료된 부분의 값을 참조하여 예측을 수행하여야 한다. JPEG의 경우에는 영상을 8x8 픽셀 단위의



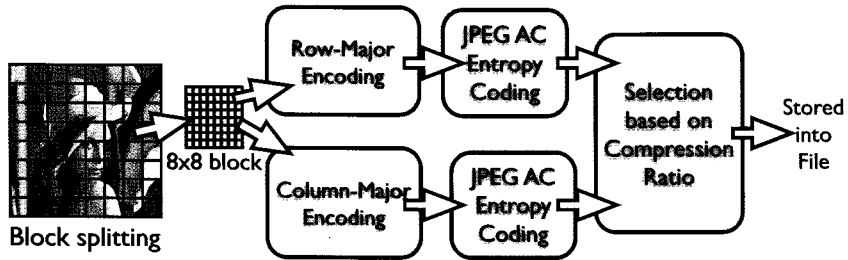


그림 2: 행/열 기반 증분 부호화의 전체적인 흐름

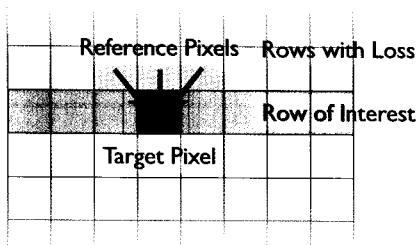


그림 3: 픽셀의 예측 참조 점

행 단위 혹은 열 단위로 부호화 된다. 블록을 행 단위로 압축하는 것이 유리한지 열 단위로 압축하는 것이 유리한지는 각각의 블록 내의 영상의 특징에 따른 예측의 정확도와 사용자가 지정한 압축률 등에 따라 결정되며 실제 압축을 수행하기 전에는 정확하게 판단하기 어렵다. 따라서 본 논문에서 제안하는 방법은 한 블록에 대하여 열 단위 압축과 행 단위 압축을 각각 적용하여 압축률을 측정하고 압축률이 높은 쪽을 선택함으로써 압축률을 향상시킨다. 이후 행/열 단위 압축 기법은 서술의 편의를 위해 행 단위 압축을 기준으로 설명한다.

특정 블록의 특정 행에 대한 행 단위 부호화는 예측, 1차원 이산 코사인 변환, 양자화의 과정으로 이루어진다. 먼저 예측 과정은 행을 구성하는 8개의 픽셀에 대하여 각각 예측을 수행한다. 전술한 바와 같이 예측을 위해 참조할 수 있는 픽셀은 이미 부호화되어 있는 픽셀로 제한되기 때문에 행 단위 압축에서는 미리 부호화되어 있는 위쪽 행들과 왼쪽 블록만을 참조할 수 있다. 따라서 본 논문에서는 그림 3에서와 같이 해당 픽셀의 위쪽 픽셀, 왼쪽 위 픽셀, 오른쪽 위 픽셀의 평균 값을 사용하여 예측을 수행한다. 이와 같은 방법은 비교적 간단하게 예측을 수행할 수 있을 뿐만 아니라 각각의 픽셀에 존재할 수 있는 오차를 분산할 수 있는 특징을 가진다. 행에 포함되는 8개의 픽셀의 예측 값이 주어지면 실제 픽셀 값과 예측 값 사이의 차이를 계산한다.

각 행에 대하여 시각적으로 중요한 성분과 비교적 덜 중요한 성분을 구분하기 위해 예측 값과 실제 픽셀 값과 차이를 나타내는 벡터에 1차원 이산 코사인 변환을 적용한다. 1차원 이산

코사인 변환은 다음과 같이 수식화 된다.

$$X_k = \sum_{n=0}^7 x_n \cos \left[ \frac{\pi}{N} \left( n + \frac{1}{2} \right) k \right] \quad k = 0, \dots, 7. \quad (1)$$

여기서  $x_n$ 는 행에서  $n$ 번째 차이 값이고  $X_k$ 는  $k$ 번째 계수이다.  $X_k$ 는 순서대로 가장 낮은 주파수 대역부터 높은 주파수 순으로 코사인 함수의 계수를 반환한다. 이때 JPEG에서와 같이 낮은 주파수에 해당되는 계수는 시각적으로 중요한 성분을 나타내고 높은 주파수에 해당하는 계수는 상대적으로 덜 중요한 성분을 나타낸다. 복호화 과정에서 사용되는 역변환은 다음과 같이 나타낸다.

$$x_k = \frac{1}{2} X_0 + \sum_{n=1}^7 X_n \cos \left[ \frac{\pi}{N} \left( k + \frac{1}{2} \right) n \right] \quad k = 0, \dots, 7. \quad (2)$$

각각의 주파수 대역별 계수는 중요도에 따라서 양자화 된다. JPEG에서와 같이 저주파 성분은 정밀하게 양자화 되고, 고주파 성분은 넓은 범위로 양자화 하여 품질 대비 압축률을 증가시킨다. 실제 양자화에 사용되는 값은 JPEG에서 사용하는 값과 동일하게 결정될 수 있다. 즉, JPEG에서와 같이 표준 양자화 표를 이용하는 방법과 RD-OPT 등의 최적 양자화 값을 계산하는 방법을 모두 사용할 수 있다. 또한 양자화 정도를 조절하여 압축률과 최종 영상의 품질을 조절할 수 있다. 본 논문에서는 JPEG에서 사용하는 표준 양자화 표의 첫 행을 사용하고, 압축률을 조절하기 위해서 양자화 표 전체를 스케일링하는 방법을 사용하였다.

### 3.2 블록 엔트로피 부호화

양자화가 완료된 정보는 엔트로피 부호화 과정을 거쳐 최종적으로 부호화 된다. JPEG에서 사용하는 엔트로피 부호화가 높은 효율을 보이는 대표적인 이유로 높은 주파수에 해당하는 계수들이 심하게 양자화 되어있고 이들이 지그재그 스캔에 의해 블록의 후반부에 집중되기 때문에 런 길이가 부호화를 이용하여 후반부의 연속적으로 나타나는 0 값들을 효과적으로 압축할 수 있다는 점을 들 수 있다. 본 논문에서도 JPEG의 이와 같은 장점을 유지하도록 하기 위해 8개의 열을 블록 단위로 모아 최대한 많은 수의 0이 연속적으로 나타내게 하고 런 길이 부호화를 적용한다.

행 단위로 이산 코사인 변환을 거친 블록은 왼쪽에서 오른쪽 방향으로 점차 높은 주파수를 나타내며 일반적으로 점차 넓은 범위로 양자화 된다. 따라서 최대한 연속되는 0을 확보하기

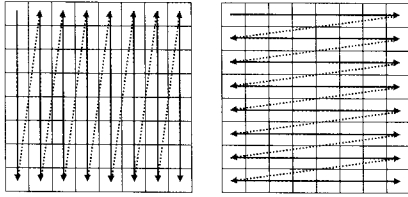


그림 4: 행(왼쪽)/열(오른쪽) 기반 블록의 계수 정렬 순서

위해 왼쪽 열 부터 스캔하여 점차 오른쪽으로 진행되는 스캔 방식을 사용한다. 행단위로 압축된 블록과 열 단위로 압축된 각각의 블록의 스캔 순서는 그림 4에 나타나 있다. 이와 같은 스캔 순서로 양자화된 값을 정렬하면 큰 값들은 블록의 앞 부분에 작은 값과 0들은 블록 뒷부분에 모이게 할 수 있다.

이와 같이 정렬된 값들은 JPEG 블록에서 나타난 값과 유사한 특징을 가진다. 따라서 본 논문에서는 행 단위로 혹은 열 단위로 압축된 블록에 JPEG에서 사용하는 엔트로피 부호화 과정을 동일하게 적용하여 압축한다. 이때 허프만 부호화에 사용되는 허프만 표는 JPEG과 동일하게 영상에 적합한 최적화 표를 생성하고 사용하는 방법과 JPEG 표준 허프만 부호화 표를 그대로 사용하는 방법을 사용할 수 있다. 단 직류 성분으로 불리는 첫번째 계수를 다른 계수들과 다른 방식으로 부호화 하는 JPEG과는 달리 블록내의 모든 계수를 교류 값으로 하여 부호화 한다. 이는 중분 부호화의 경우에는 각 행의 첫번째 계수 값 역시 중분에 의해 나타나는 값으로 대역폭이 교류 성분과 유사하기 때문이다. 본 논문의 실험에서는 블록 내의 64개 계수에 대하여 JPEG 표준 교류 허프만 부호화 표를 사용하여 압축을 수행한 결과가 화질이나 압축률의 손실이 거의 없이 가장 빠른 성능을 보임을 확인하였다.

#### 4 JPEG과 행/열 단위 중분 압축의 혼합

행/열 단위 중분 압축 방법은 정확한 예측이 수행될 경우 화질 대비 높은 압축률을 나타낸다. 하지만 블록 내의 영상의 특성 상 정확한 예측 값의 획득이 어려운 경우에는 높은 압축률을 기대하기 어렵다. 이와 같은 문제점은 사용자가 압축률을 향상시키기 위해 품질을 낮게 지정한 경우에도 나타난다. 양자화 과정에서 손실이 커지면 참조 값이 심하게 변화하기 때문에 정확한 예측을 수행하기 어렵기 때문이다. 따라서 사용자가 높은 품질을 지정한 경우에는 행/열 단위 중분 압축 방법이 JPEG에 비해 높은 압축률을 보이지만 화질의 손상을 감수하는 경우에는 JPEG이 본 논문에서 제안하는 방법보다 우세한 압축률을 보이는 경우가 나타난다.

이와 같은 제한점을 해결하기 위해 본 논문에서는 각 블록에 대하여 JPEG 압축 방법과 행/열 단위 중분 압축 중 보다 효과적인 방법을 선택하여 파일에 저장하는 방법을 제안한다. JPEG과 행/열 단위 중분 압축을 혼합하여 사용하는 방법의 전체적인 흐름은 그림 5에 나타나 있다. 혼합 방법에서는 압축 방식을 결정하기 위해 먼저 각 블록을 JPEG 방식과 중분 압축 방식을 각각 사용하여 부호화, 복호화하고 블록에 대한 압축률과 품질을 평가한다. 중분 압축 방법의 품질과 압축률이 동시에 우월할 경우에만 중분 압축을 적용하여 압축률과 품질을 동시에 향상시킨다. 여기서 품질은 원본 영상과 복호화 영상을

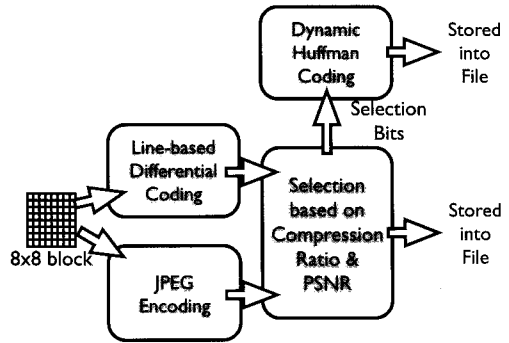


그림 5: 혼합 압축 방법의 흐름

비교하고 PSNR(Peak Signal to Noise Ratio)를 측정하여 PSNR이 높은 경우 화질이 좋은 것으로 판단한다.

이때 각 방법의 양자화 정도를 적절히 결정하여야 PSNR 대비 최적의 압축률을 얻을 수 있다. 본 논문에서는 JPEG을 기준으로 사용자가 지정한 바에 따라 JPEG 양자화 스케일 값을 결정하고 행/열 기반 압축에서는 이와 유사한 PSNR 값을 가질 것으로 예상되는 스케일 값을 찾아 양자화 정도를 결정한다. 행/열 기반 압축에 사용되는 양자화 스케일 예상 값은 다양한 그라데이션과 평면, 예지, 세부적인 형태를 고루 가지고 있는 것으로 알려진 Lenna 영상(그림 6의 왼쪽 위)을 사용하여 예측한다. 사진에 Lenna 영상에 대하여 다양한 양자화 스케일 값을 이용하여 행/열 기반 압축을 수행하고, 양자화 스케일 값 대비 PSNR 값을 미리 저장한다. 부호화 시에는 JPEG의 PSNR이 결정되면 이와 대응되는 양자화 스케일 값을 미리 저장된 값을 기준으로 결정하고 행/열 기반 압축에 적용한다. 이와 같은 방법으로 일반적인 경향을 가지는 영상에 대하여 JPEG을 사용한 블록과 행/열 단위 압축을 사용한 블록의 품질을 고르게 유지하면서 압축률을 향상시킬 수 있다. 물론 Lenna 영상의 특징을 따르지 않는 영상의 경우에는 이와 같은 방법으로 결정된 스케일을 사용할 경우 최적의 압축률을 얻지 못할 수 있다. 즉, 행/열 단위 압축이 유리한 경우에도 JPEG을 선택하는 경우가 생길 수 있다. 하지만 이러한 경우에도 JPEG 방법만을 사용한 경우에 비해 높은 압축률을 보임을 실험적으로 확인하였다.

복호화 과정에서 각 블록이 JPEG 방법을 이용하여 압축된 블록인지 중분 압축 방법을 사용한 블록인지에 따라 적절한 복호화 방법을 사용하여야 한다. 따라서 부호화 과정에서는 각각의 블록이 어떤 방식으로 압축되었는지를 나타내는 블록당 1비트의 정보를 추가로 저장하여야 한다. 블록 당 1비트의 정보량은 JPEG과 같은 손실 압축의 경우에는 무시하지 못하는 추가 부담(overhead) 될 수 있다. 따라서 본 논문에서는 그림 5에서와 같이 각 블록의 압축 방식을 나타내는 비트를 모아 동적 허프만 압축 방식(zip)을 이용하여 압축하여 저장함으로써 전체적인 압축률을 향상시킨다.

#### 5 실험 결과

본 논문에서 제안하는 행/열 기반의 압축 기법의 성능을 실험하기 위한 실험 방법은 다음과 같다. 먼저 실험에 사용한 영상



그림 6: 실험 영상: Lenna(왼쪽 위), Baboon(오른쪽 위), Airplane(왼쪽 아래), Peppers(오른쪽 아래)



그림 8: 압축 결과: PSNR 37.5일때 JPEG(왼쪽 위), 제안한 방법(오른쪽 위), JPEG의 오차 영상(왼쪽 아래), 제안한 방법의 오차 영상(오른쪽 아래). 각각의 파일 크기는 24951 byte, 24560 byte이다.

은 그림 6에 나타나 있는 4개의 영상이다. 각각은 512×512의 크기를 가지며 밝기 채널(luminance channel)만을 가지는 영상을 사용하였다. 실험 영상들은 복잡도나 영상의 지역적, 전체적 특징이 다양하게 나타나 압축률 평가에 적절하다고 할 수 있다. 양자화에는 JPEG의 경우 표준 JPEG 양자화 표를 스케일하여 사용하였고, 행/열 기반 압축에서는 표준 JPEG 양자화 표의 첫 행을 스케일하여 사용하였다. 스케일에 따라 화질과 압축률이 달라지기 때문에 스케일 값을 0.05 단위로 변화시키며 각 방법의 압축률과 PSNR을 계산하는 방법으로 화질 대비 압축률을 비교하였다.

먼저 그림 7은 본 논문에서 제안하는 행/열 기반 압축 방법을 사용하여 압축한 경우 화질 대비 압축률을 JPEG 방식과 비교한 결과이다. 표준 영상인 Lenna의 경우에는 고화질에서는 JPEG보다는 높은 압축률을 가지고 PSNR이 37.5 근처에서 같은 압축률을 보이는 것으로 나타났다. 이는 압축률이 낮은 경우 화질이 좋아지고, 그 결과 예측에 사용되는 값이 정확해 지기 때문에 증분 부호화가 보다 효과적이 되기 때문이다. Baboon의 경우에는 높은 주파수 대역을 많이 포함하고 있는 영상으로 JPEG에서는 압축률 대비 고화질을 보장할 수 없는 반면, 증분 압축에서는 적은 단위에 대한 이산 코사인 변환을 수행하기 때문에 화질을 유지하는데 유리한 것을 확인할 수 있다. Airplane 영상의 경우에는 단순한 면들이 비교적 많이 나타나기 때문에 PSNR 기준으로 34.5정도에서도 증분 부호화 기법이 JPEG보다 우수한 압축률을 나타냈다. Peppers 영상은 밝기가 부드럽게 변화하는 면이 많아 JPEG과 제안하는 방법이 공히 높은 성능을 보였으며 높은 화질에서는 역시 제안하는 방법이 보다 높은 압축률을 보임을 알 수 있다.

그림 8은 JPEG 압축 방식과 본 논문에서 제안하는 압축 방식의 화질을 보여준다. 그림의 위쪽 줄에 나타난 영상은 좌, 우 각각 JPEG과 본 논문에서 제안한 방법으로 Lenna 영상을 부호화하고 복호화한 결과를 보여준다. 각 영상은 공정한 비교를 위

하여 양자화 표의 스케일을 같은 수준의 PSNR(37.5)을 갖도록 조절하여 얻은 영상이다. 결과에서는 전체적인 영상의 품질이 같은 PSNR에서 유사한 특징을 가지는 것을 볼 수 있다. 따라서 4절에서 언급한 혼합 방법에서도 각 블럭이 가지는 오차의 양상이 유사하기 때문에 블럭간의 특징 차이는 거의 나타나지 않는다.

그림 9는 JPEG 압축 방식과 본 논문에서 제안하는 JPEG블럭과 행/열 단위 부호화 블럭을 혼합한 방식의 압축률을 비교한 그래프이다. 각각의 그래프에서 본 논문에서 제안하는 증분 부호화를 이용한 확장된 JPEG 압축 방식이 같은 PSNR에서 높은 압축률을 가지는 것을 알 수 있다.

## 6 결론

본 논문에서는 증분 압축 기법을 손실 영상 압축에 적용하여 JPEG 영상 압축 방식을 개선하기 위한 방법을 제안하였다. 증분 압축에서 필수적인 참조 예측을 정확하게 수행하여 압축률을 향상시키기 위해 행이나 열 단위로 손실 압축을 수행하고 그 결과를 다음 행이나 열에 사용하는 방법을 제안하였다. 특히 JPEG이 가지는 장점을 유지하기 위해 증분 압축 기법을 JPEG과 혼합할 수 있도록 하여 선택적으로 JPEG과 제안하는 압축방식을 선택하여 사용함으로써 압축률을 향상시키기 위한 방법을 제안하였다.

결과적으로 본 논문에서 제안하는 방법은 전반적인 손실 패턴이나 잡음은 JPEG과 유사한 형태를 보여 두 방법을 혼용할 때 시각적 품질의 손실이 JPEG과 유사한 것으로 나타났다. 압축률의 경우에는 화질이 높은 경우에는 정확한 예측이 가능하여

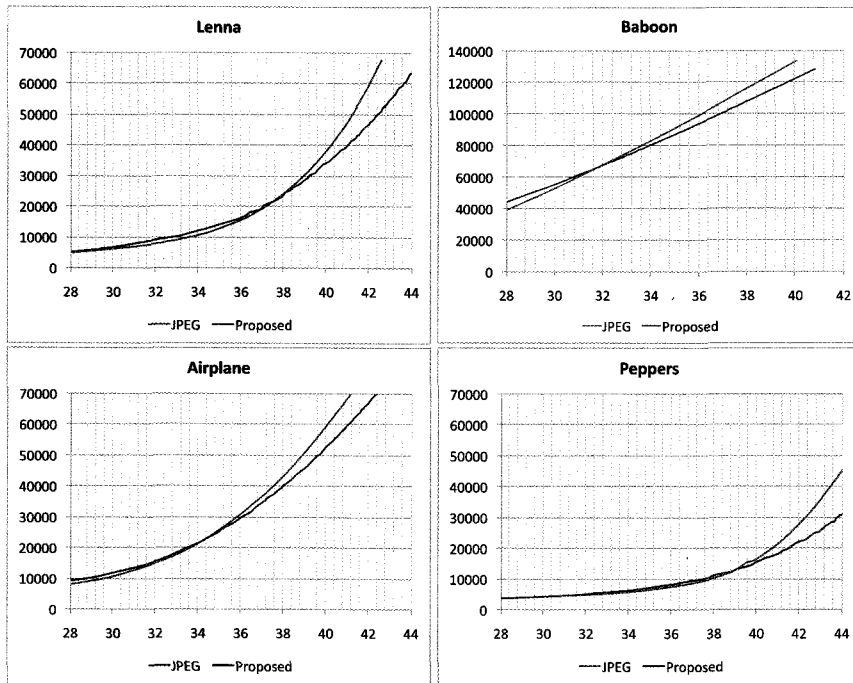


그림 7: JPEG과 행/열 단위 부호화 방식의 PSNR 대비 압축 파일 크기 비교 (x축은 PSNR, y축은 압축 파일의 크기, Baboon의 경우 축의 크기가 다른 그래프와 다름)

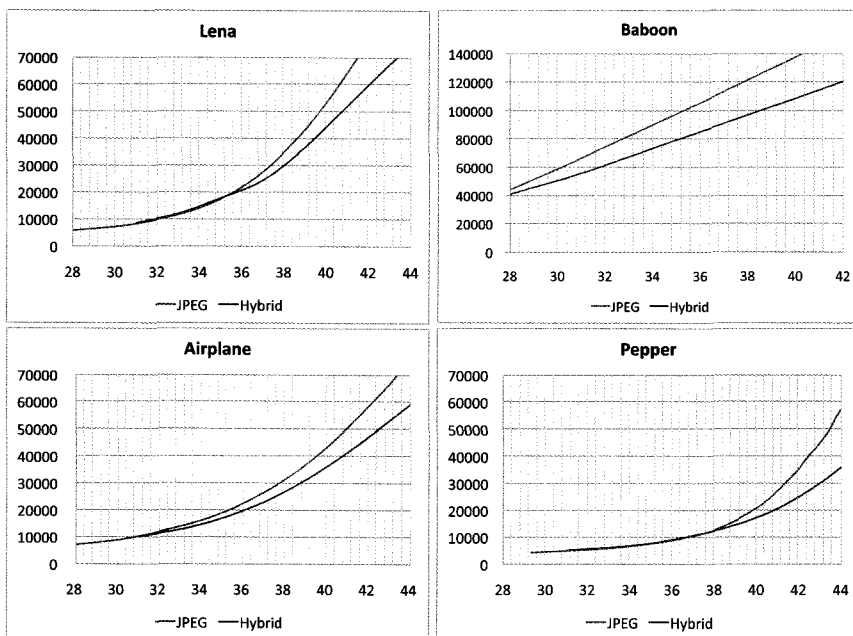


그림 9: JPEG과 혼용 방식의 PSNR 대비 압축 파일 크기 비교 (x축은 PSNR, y축은 압축 파일의 크기, Baboon의 경우 축의 크기가 다른 그래프와 다름)

중분 압축 기법의 장점을 최대한으로 살릴 수 있고 따라서 높은 압축률을 나타낼 수 있었다. 사용자가 낮은 화질을 선택한 경우 예측이 부정확해 지기 때문에 JPEG 압축 방식을 그대로 사용하게 되는 블록의 개수가 많아지고, 압축 방법을 저장하기 위한 약간의 추가 부담(overhead)가 발생한다.

본 논문에서 제안하는 방법의 단점으로 압축 속도를 들 수 있다. 본 논문에서는 각각의 블록을 행단위, 열단위, JPEG 방식의 세 가지 방법을 각각 적용하고 이들 중 가장 압축률과 PSNR이 좋은 방법을 선택하기 때문에 실험적으로 JPEG에 비하여 2.8배 정도의 시간을 요구하는 것으로 나타났다. 하지만 압축 해제 시간은 PSNR과 영상의 특성에 따라서 0.91배의 속도를 보이는 것을 확인하였다.

본 논문에서 제안하는 방법에서 개선해야 할 가장 중요한 문제점으로 JPEG과 행/열 단위 압축 방법을 혼용할 경우 각 블록에 대하여 사용할 방법을 결정하는 과정을 들 수 있다. 블록에 따라서 JPEG과 행/열 단위 압축 방법이 선택적으로 사용되기 때문에 행/열 단위 압축 방법이 선택된 블록은 영상의 전체적인 특징과는 다른 특성을 갖게 되고, PSNR과 압축률 간의 관계가 Lenna 영상의 전체적인 양상과는 다를 수 있다. 따라서 최적의 결과를 얻기 위해서는 보다 정교하게 블록 별 압축 방법을 결정하는 방법에 관한 향후 연구가 필요하다.

## 감사의 글

본 연구는 위 연구는 2007학년도 아주대학교 일반연구비 지원에 의하여 연구되었으며 방위사업청과 국방과학연구소(UD060048AD)의 지원으로 수행되었습니다.

## 참고문헌

- [1] J. Miano, *Compressed Image File Formats*. Addison-Wesley, 2000.
- [2] A. B. Watson, "Dct quantization matrices visually optimize for individual images," in *Proceedings of the SPIE*, 1993.
- [3] A. B. Watson, "Image compression using discrete cosine transform," *Mathematica Journal*, vol. 4, no. 1, pp. 810–88, 1994.
- [4] L. Pitali, *Variable Length Coding*. XAPP621, 2005.
- [5] I. E. G. Richardson, *H.264 and MPEG-4*. Wiley, 2004.
- [6] G. K. Wallace, "The jpeg still picture compression standard," *Communications of the ACM*, vol. 34, no. 4, pp. 30–44, 1991.
- [7] J. L. Mitchell and W. B. Pennebaker, *JPEG Still Image Data Compression Standard*. Kluwer Academic Publishers, 1992.
- [8] F. L. V. Nes and M. A. Bouman, "Spatial modulation transfer in the human eye," *Journal of the Optical Society of America*, vol. 57, no. 3, p. 401, 1967.
- [9] V. Ratnakr and M. Livny, "Rd-opt: An efficient algorithm for optimizing dct quantization tables," in *Proceedings of Data Compression Conference*, 1995, pp. 332–341.

[10] Independent jpeg group. [Online]. Available: <http://www.ijg.org>

## 〈저자소개〉



### 박대현

- 2007년 2월 아주대학교 정보및컴퓨터공학부 학사
- 2009년 2월 아주대학교 정보통신공학과 석사
- 2009년 3월 ~ 큐램 연구소 연구원
- <관심분야> 영상압축



### 안영훈

- 2006년 2월 아주대학교 컴퓨터공학부 (학사)
- 2008년 2월 아주대학교 정보통신전문대학원 (석사)
- 2008년 3월 ~ 아주대학교 컴퓨터공학과 (박사과정)
- <관심분야> 이미지 손실/무손실 압축, 이미지 영상처리, 동영상 화질 개선



### 신현준

- 1995년 2월 한국과학기술원 전산학과 (학사)
- 1997년 2월 한국과학기술원 전산학과 (석사)
- 2002년 2월 한국과학기술원 전산학과 (박사)
- 2002년 3월 ~ 2002년 8월 한국과학기술원 박사후연구원
- 2002년 9월 ~ 2003년 12월 University of Wisconsin Visiting Scholar
- 2004년 1월 ~ 2004년 2월 한국과학기술원 박사후연구원
- 2004년 3월 ~ 2005년 2월 서울대학교 박사후연구원
- 2005년 2월 ~ 현재 아주대학교 미디어 학부 교수
- <관심분야> 컴퓨터그래픽스, 영상처리, 그래픽시스템 등



### 위영철

- 1982년 12월 State Univ. of NY at Albany 전산학과 (학사)
- 1984년 12월 State Univ. of NY at Albany 전산학과 (석사)
- 1989년 12월 State Univ. of NY at Albany 전산학과 (박사)
- 1990년 3월 ~ 1995년 4월 삼성종합기술원 수석연구원
- 1995년 5월 ~ 1998년 2월 현대전자 기획부장
- 1998년 3월 ~ 현재 아주대학교 정보및컴퓨터공학부 교수
- <관심분야> 영상처리, 정보 압축, 컴퓨터 그래픽스