# Detection and Reduction of JPEG Artifacts Caused by Large Dynamic Range

*Reiner Eschbach and James Bollman*
*Xerox Digital Imaging Technology Center*
*Webster, New York*

## Abstract

In order to efficiently transmit image data, compression is commonly employed. The standard compression method used for continuous tone images is the JPEG defined algorithm. Certain artifacts can arise from this lossy compression method. One of the artifacts is connected to Gibbs over- and under-shoots for signals with large dynamic range. These artifacts occur, even if the best quality is selected for the JPEG compression. This paper describes the artifact and gives an example of a noise metric that can be derived for this specific artifact. The noise metric is then used in a post-processing step to adaptively filter the output image to reduce the artifacts.

## Introduction

Digital images are commonly compressed for storage and/or transmission in order to reduce the amount of data needed to describe the image. The standard compression algorithm is the Joint Photographic Expert Group ( JPEG ) Standard.[1] The JPEG algorithm was originally designed for continuos tone images, or photographs. Due to its generally high quality output, more and more applications emerged that were based on JPEG, but where the input data would not normally be described as "continuous tone" or "photographic". A good example of this is the use of JPEG for mixed raster documents, i.e.: documents that contain multiple objects on a page, where some objects might be photographic, while other objects might be text, line art or computer graphic.

As a logical consequence, some of these applications uncover shortcomings of the compression algorithm in areas that are outside of its original design scope. One of the shortcomings is the edge ringing for large dynamic range inputs, such as scanned text or line art. This edge ringing is even present when the compression ratio is set to be approximately "1", or no compression.

This paper will describe a simple method to detect the likelihood of such edge ringing and it will give an estimate of the strength of the artifact, along with a simple method to compensate for it. The advantage of this method over previous methods is that it does not rely on an iterative and thus slow decompression. The disadvantage of the described method is that it offers less flexibility than the iterative approach.

## JPEG Edge Ringing

In the JPEG algorithm, the input image is divided into 8x8 non-overlapping blocks. Each block is transformed using the Discrete Cosine Transform (DCT). The DCT coefficients are subsequently quantized, using a quantization table specifying the integer denominator for each component. The quantized DCT coefficients are then re-ordered and lossily encoded by a Huffman-style coding. The DC terms are encoed using a predictive scheme.

In decompression, the process is applied in reverse order. A Huffman decoder is followed by a multiplier. The multiplier is, in general, the same as the denominator used in compression. Some exceptions are edge enhancement in facsimile transmission and the like. An inverse DCT is applied as the last step to recreate the 8x8 image block in an approximate fashion.

Two sources of noise can be found in this process. The first source is the approximative nature of the "integer division" and "integer multiplication" process, e.g.: $\lfloor 7 / 2 \rfloor = 3$, followed by $3\times 2 = 6$, resulting in an input output difference of "1". The other source is caused by the Gibbs phenomenon followed by hard-limiting the values to the original 8 bit input range.

The first of the two noise sources manifests itself as either blocking artifacts in slowly varying areas [ differences in DC and low frequency AC terms ] or as "mosquito noise" around edges [ differences in higher AC terms ]. The second artifact has the same appearance as the mosquito noise, but its existence is restrained to very high contrast edges, as they often appear in scanned text or line art.

The following numerical example shows an input signal of high dynamic range and the corresponding decompressed signal, where the bounding to the 8 bit dynamic range was not yet performed.

The input 8x8 block is:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 |
| 255 | 255 | 255 | 255 | 255 | 255 | 255 | 167 |
| 255 | 255 | 255 | 255 | 255 | 255 | 119 | 5 |
| 255 | 255 | 255 | 255 | 255 | 173 | 0 | 0 |
| 255 | 255 | 255 | 247 | 143 | 39 | 0 | 0 |
| 255 | 255 | 255 | 137 | 0 | 0 | 0 | 0 |
| 255 | 255 | 167 | 37 | 0 | 0 | 0 | 0 |
| 255 | 119 | 5 | 0 | 0 | 0 | 0 | 0 |

 

Resulting after decompression in

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 246 | 268 | 253 | 246 | 268 | 255 | 237 | 261 |
| 267 | 239 | 259 | 261 | 241 | 278 | 261 | 145 |
| 249 | 260 | 242 | 233 | 259 | 240 | 130 | 18 |
| 254 | 279 | 259 | 255 | 261 | 158 | 23 | -15 |
| 258 | 235 | 267 | 249 | 127 | 43 | 18 | -20 |
| 252 | 262 | 233 | 131 | 19 | -18 | -7 | -5 |
| 264 | 266 | 150 | 33 | 19 | 2 | -16 | 23 |
| 247 | 113 | 23 | 1 | -13 | -3 | 9 | -12 |

As can be seen, several of the values of the 8x8 example block are decompressed to values that lie outside the [0,255] interval. In standard decompression, all outside values are clipped to the interval, resulting in

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 246 | 255 | 253 | 246 | 255 | 255 | 237 | 255 |
| 255 | 239 | 255 | 255 | 241 | 255 | 255 | 145 |
| 249 | 255 | 242 | 233 | 255 | 240 | 130 | 18 |
| 254 | 255 | 255 | 255 | 255 | 158 | 23 | 0 |
| 255 | 235 | 255 | 249 | 127 | 43 | 18 | 0 |
| 252 | 255 | 233 | 131 | 19 | 0 | 0 | 0 |
| 255 | 255 | 150 | 33 | 19 | 2 | 0 | 23 |
| 247 | 113 | 23 | 1 | 0 | 0 | 9 | 0 |

The mosquito noise can be clearly seen in this example, with an amplitude reaching a digital count of "23".

## Reduction of JPEG Ringing

In this paper we propose to use the JPEG data before dynamic range clipping as a detector for the likelihood of edge ringing. In this case, we assume that at every place, where the dynamic range exceeds the 8 bit interval some ringing noise is visible in close spatial proximity. The simplest form of proximity which will also be used in this paper is a 3x3 window.

Additionally, we will assume that the amplitude of the dynamic range violation is a good indicator for the local mosquito noise and that the overall image grayscale should be preserved.

The method can best be explained by using a simple example. Assume, for simplicity, an area of 4x4 pixels with the following values:

| | | | |
|---|---|---|---|
| 10 | 12 | 7 | 255 |
| 3 | 2 | -4 | 253 |
| 9 | -9 | -6 | 255 |
| 4 | -12 | 4 | 254 |

In this case, the first out-of-range number that is encountered (in standard left-right, top-bottom processing) is the value "-4" . In a simple implementation, the neighborhood is examined, starting with the pixel having the value "12". The two pixels can be brought into dynamic range without any changes in overall brightness by reducing the value of the in-range pixel by "4", giving a new array having the values:

| | | | |
|---|---|---|---|
| 10 | 8 | 7 | 255 |
| 3 | 2 | 0 | 253 |
| 9 | -9 | -6 | 255 |
| 4 | -12 | 4 | 254 |

The next out-of-range pixel has the value "-9" which is corrected using the pixels valued "3" (above left), "2" (above) and "9" (right) in the processing order to give:

| | | | |
|---|---|---|---|
| 10 | 8 | 7 | 255 |
| 0 | 0 | 0 | 253 |
| 5 | 0 | -6 | 255 |
| 4 | -12 | 4 | 254 |

The next out-of-range pixel has the value "-6" which is partly corrected using the pixel valued "4", giving:

| | | | |
|---|---|---|---|
| 10 | 8 | 7 | 255 |
| 0 | 0 | 0 | 253 |
| 5 | 0 | -2 | 255 |
| 4 | -12 | 0 | 254 |

It should be noted that the pixels valued at "255" and "253" etc. are not used in the compensation. This is motivated by the fact that out-of-range values will occur at object edges and should not be corrected by softening the edge. For out-of-range values that are smaller than "0", only pixels are considered whose values are positive and smaller than a preset value $\Delta$, with $\Delta$ being somewhere between 20 and 50. For out-of-range values that are larger than "255", only pixels are considered whose values are inside the dynamic range and larger than a preset value $\Delta^*$, with $\Delta^*$ being somewhere between 205 and 235 ( i.e. $\Delta^* = 255 - \Delta$).

For the current pixel having a value of "-2", this means that no compensating pixel location can be found in the 3x3 neighborhood. The pixel is then clipped to the interval boundary.

| | | | |
|---|---|---|---|
| 10 | 8 | 7 | 255 |
| 0 | 0 | 0 | 253 |
| 5 | 0 | 0 | 255 |
| 4 | -12 | 0 | 254 |

The compensation of the last out-of-range pixel leads to a final output block

| | | | | |
|---|---|---|---|---|
| 10 | 8 | 7 | 255 | |
| 0 | 0 | 0 | 253 | (a) |
| 0 | 0 | 0 | 255 | |
| 0 | 0 | 0 | 254 | |

Comparing the resulting output block shown in (a) to the output block as it would be created using the standard decompression shown in (b)

| | | | | |
|---|---|---|---|---|
| 10 | 12 | 7 | 255 | |
| 3 | 2 | 0 | 253 | (b) |
| 9 | 0 | 0 | 255 | |
| 4 | 0 | 4 | 254 | |

on sees that the noise in the block is reduced.

## Conclusion

A simple filtering method can be used to reduce edge ringing in JPEG. This method requires that the data can be accessed before the dynamic range clipping is performed, since the out-of-range values are used to detect and reduce the ringing.

More complicated filtering methods can be envisioned, e.g.: spreading the out-of-range contribution to a group of pixels instead of the first pixel in scanline order. Additionally, the DC preservation can be softened to allow a stronger noise filtering, e.g.: one might want to compensate for up to N times the local out-of-range amplitude.

## References

1. ITU-T Rec. T.81, ISO 10918

## Biographies

Reiner Eschbach received his M.S. and Ph.D. in physics from the University of Essen in 1983 and 1986 respectively. He joined Xerox in 1988 where he became a Pricipal Scientist at the Xerox Digital Imaging Technology Center in 1994. His research interests include color image processing, digital halftoning and compression. He is IS&T Publication VP and the Editor of the Recent Progress Series of IS&T.

Jim Bollman received his BSEE from Valparaiso Technical Institute and has worked in imaging at Xerox since 1969. He started working in Digital Imaging in the early 70s and is currently a Project Leader developing image processing toolkits for engineering and research use.