# Image Quality Estimation for JPEG-Compressed Images without the Original Image

Atanas Petkov and Scott Cottier

**Abstract—Methods for estimating JPEG-compressed image quality without the original image are investigated. Estimation of peak signal-to-noise (PSNR) is calculated by utilizing the Laplacian nature of the image transform coefficients along with the quantization step size used to generate the image. Estimation of the Laplacian lambda (λ) values are shown as well as improvements to the PSNR estimation that are achieved when compensating for large quantization step sizes. PSNR estimation using the ITU-T J.240 standard is also implemented.**

## 1. INTRODUCTION

The use of digital images has become the dominant method for capturing and sharing visual data. Along with the use of digital images, the amount and usage of image editing software has become prevalent. The knowledge and ability to detect the quality of an image is a vital for understanding the manipulations that have been performed on the original image. Forgery, region modification and medical image quality are all examples that call for a measure of image quality.

This paper investigates the quality of images using peak signal-to-noise (PSNR) estimation when the source or master image is not available. PSNR estimation without the original image excludes the use of the mean squared error as the error metric. Instead, the mean error must be estimated through a secondary approach. The discrete cosine transform (DCT) of natural images produces AC coefficient values with Laplacian probability distribution functions (PDFs). This Laplacian nature is exploited in the calculation of the PSNR. Also, image quality is estimated by applying the ITU-T J.240 standard. Section 2 describes the methods for estimating the lambda values for the AC coefficient Laplacian distributions. The calculation of PSNR from the estimate lambda values and test results for PSNR estimation are shown in section 3. Section 4 explores different methods of estimating the JPEG quantization matrix if it is not available, which is often the case. Section 5 briefly describes the J.240 video coding standard and how it can be applied for PSNR estimation of JPEG images. Section 6 summarizes the results and concludes this study.

## 2. LAMBDA ESTIMATION

The AC coefficients can be modeled with a Laplacian PDF. The estimation of lambda (λ)

$$f_X(x) = \frac{\lambda}{2} e^{-\lambda|x|} \quad (1)$$

for the distribution is shown in [1] for the maximum-likelihood (ML) method with the result shown in (2). The DCT coefficient values for each frequency are analyzed to generate an 8x8 matrix of lambda values

$$\lambda_{ML} = -\frac{2}{\Delta} \log \frac{-N_0\Delta + \sqrt{N_0^2\Delta^2 - 4(N\Delta + 2S_1)(N_1\Delta - 2S_1)}}{2N\Delta + 4S_1} \quad (2)$$

where $\Delta$ is the quantization step size, N is the number of coefficient values, $N_0$ is the number of zero-valued coefficients, $N_1$ is the number of non-zero valued coefficients and $S_1$ is the sum of the absolute values for the coefficients. The weakness of this approximate, also shown in [1], is that as $\Delta$ increases $\lambda_{ML}$ approaches infinity. The high frequency coefficients in JPEG compressed images have a much larger quantization step size that the low frequency coefficients. Therefore the lambda values for the high frequency coefficients will not be estimated with high quality. Practical limits for the maximum lambda value have been implemented for the cases where the lambda value becomes large.

Lambda compensation is performed by scaling the lambda values. The lambda values are split into six groups of similar frequencies and all values within a single group get the same scale factor applied. Fig. 1 shows the arrangement of the six groups overlaid on the standard JPEG quantization matrix. The scale values are determined by minimizing the PSNR error between the true PSNR for a set of control images and the corresponding JPEG compressed images and the estimated PSNR using the technique described in section 3. The lambda scale values used for this work are shown in Fig. 2.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 16 | 11 | 10 | 16 | 24 | 40 | 51 | 61 |
| 12 | 12 | 14 | 19 | 26 | 58 | 60 | 55 |
| 14 | 13 | 16 | 24 | 40 | 57 | 69 | 56 |
| 14 | 17 | 22 | 29 | 51 | 87 | 80 | 62 |
| 18 | 22 | 37 | 56 | 68 | 109 | 103 | 77 |
| 24 | 35 | 55 | 64 | 81 | 104 | 113 | 92 |
| 49 | 64 | 78 | 87 | 103 | 121 | 120 | 101 |
| 72 | 92 | 95 | 98 | 112 | 100 | 103 | 99 |

**Figure 1: Groupings for the lambda compensation overlaid on the standard JPEG quantization matrix**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0.2 | 0.2 | 2.0 | 2.0 | 2.0 | 2.0 | 5.0 | 5.0 |
| 0.2 | 2.0 | 2.0 | 2.0 | 2.0 | 5.0 | 5.0 | 5.0 |
| 2.0 | 2.0 | 2.0 | 2.0 | 5.0 | 5.0 | 5.0 | 5.0 |
| 2.0 | 2.0 | 2.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 |
| 2.0 | 2.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 |
| 2.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 |
| 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 |
| 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 |

**Figure 2: Scale values for lambda compensation**

The test results show that a hybrid approach between the maximum-likelihood method and the compensated method gives the best results for PSNR accuracy. The ML method is a better predictor for JPEG images compressed with lower quality settings while the compensation method give better results for images compressed with higher quality settings.

## 3. PSNR ESTIMATION

PSNR is a metric that compares a set of reference values to a set of estimated values, usually incorporating the mean squared error as the error function. For this work, the reference image is not known, therefore the PSNR cannot be calculated directly. The error function incorporating the estimated lambda values is described in [1] and shown here.

$$\mathrm{PSNR}[dB] = 10 \cdot \log_{10} \frac{255^2}{\frac{1}{M}\sum_{k=1}^{M} \varepsilon_k^2} \qquad (3)$$

M is the number for AC coefficient frequencies and $\varepsilon_k^2$ is the error function shown in (4). $X_k$ is the quantization level, $P(X_k)$ is the probability of the occurrence of the quantization level $X_k$, and $f_X(x)$ is the Laplacian PDF from (1).

$$\varepsilon_k^2 = \frac{1}{P(X_k)} \int_{X_k - \frac{\Delta}{2}}^{X_k + \frac{\Delta}{2}} f_x(x)(X_k - x)^2\, dx \qquad (4)$$

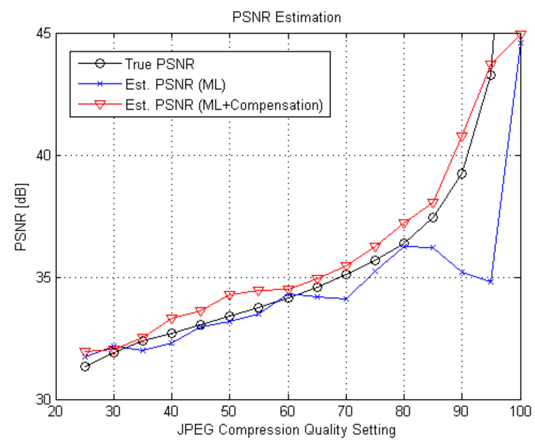


**Figure 3: PSNR estimation using ML and compensation methods**

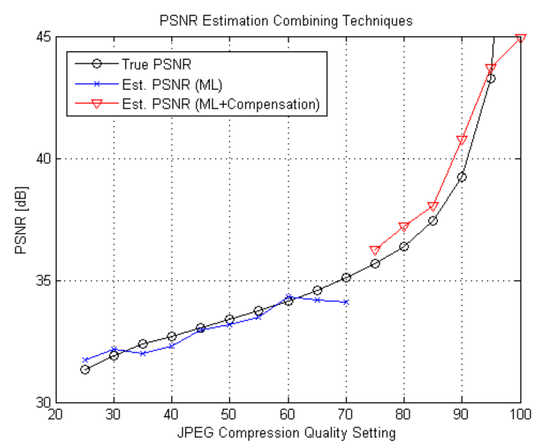


**Figure 4: Hybrid PSNR estimation combining the ML and compensation methods**

Fig. 3 shows the PSNR estimations for a test image with the ML and compensation methods. Fig. 4 shows the hybrid approach of taking the ML estimates for the lower JPEG quality settings and the compensation estimates for the high qualities. The JPEG quality setting is calculated from the raw image data through the techniques discussed in section 4.

## 4. *Q* MATRIX ESTIMATION

In previous sections, in order to estimate the $\lambda$ parameter and consequently – the PSNR, it was assumed that the quantization matrix for the JPEG transform coefficients was known. Often, however, this is not the case. Extracting the quantization matrix from the compressed image can be considered the first step in the quality evaluation routine.

Finding the quantization matrix from the quantized coefficients can be difficult when dealing with a random quantization matrix. Fortunately, in 90% of the cases JPEG files are quantized with a standardized quantization matrix [2] or some scalar multiple of it. The unknown quantization matrix $Q$ can then be represented as some scalar multiple of the JPEG standard matrix $W$ as shown in (5).

$$Q = sW \tag{5}$$

The value of the scalar $s$ determines the quality of the JPEG compression. JPEG quality ranges from 0 to 100, where a quality of 100 means no compression while a quality of 0 means maximum compression. The convention is that a quality of 50 represents a value of $s = 1$, meaning the standard JPEG matrix was used. A quality of 75 corresponds to $s = 0.5$ and a quality of 25 to $s = 2$. For the purpose of this project, only quality values between 15 and 90 are considered (corresponding to a 30 to 42 dB range).

The purpose of this exercise then becomes to estimate the $s$ value in order to find the $Q$ matrix and consequently – the PSNR. A scheme to estimate the quantization parameters has been developed by Turaga et. al. which applies to frames used in MPEG-2 video compression [3]. The scheme relies on estimating the differences between dominant representative levels (DRLs) in quantization bins corresponding to different frequency coefficients. The method suggested to extract those differences is by “choosing the lag that maximizes the autocorrelation function of the histogram of the corresponding AC coefficient” [3]. For this project a similar method will be used to estimate $s$.

Once a JPEG image (x’) is adjusted with a level offset (by subtracting the mid-level – 128 for 8-bit images) and transformed block-wise with the DCT, the resulting coefficients (X’) are all quantized versions of the original image’s DCT coefficients (X). Furthermore, all of them are multiples of the scaling factor – $s$, since every value in the quantization matrix $W$ was multiplied by $s$. To make things easier by working with smaller numbers, each 8x8 block of the quantized transform coefficients (X’) is divided by the standard JPEG matrix $W$. The resulting coefficients (Y’) are smaller multiples of $s$. If a histogram of those coefficients (Y’) is plotted, the value of $s$ can usually be inferred from the histogram visually by looking at the distances between each histogram peak (Fig.5).

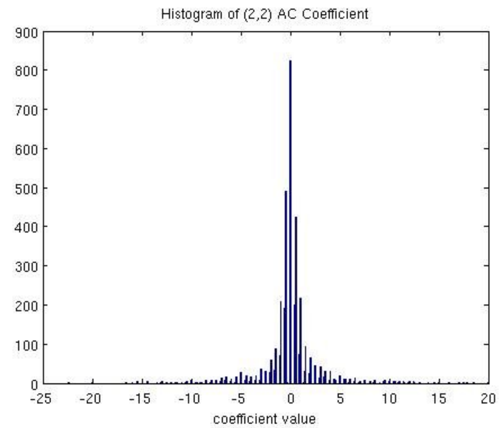


**Figure 5: Histogram of the (2, 2) AC coefficient. The distance between the peaks corresponds to the value of the scaling factor *s*.**

However, the project calls for a more robust and automated approach, which is where the auto-correlation comes into play. The highest peak of the histogram’s auto-correlation function corresponds to a displacement of zero. However, the displacement corresponding to the peak closest to the peak at zero is the value of $s$ that is sought (Fig. 6).

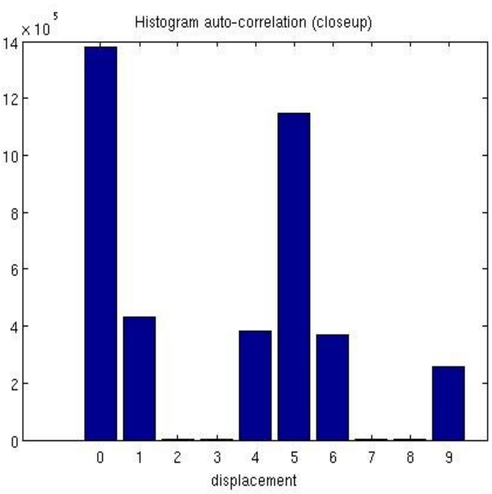


**Figure 6: Close-up of the auto-correlation function of the histogram of the (2, 2) AC coefficient. The value of *s* corresponds to the location of the non-zero peak closest to zero. The horizontal resolution is 0.1 per unit.**

To deal with some bogus peaks that might be closer to zero than the desired one, an extra condition is enforced, which is that the peak has to be higher than the mean value of the autocorrelation function. This is a reasonable condition to impose, since the transform coefficients are all multiples of *s* and a displacement corresponding to *s* units will result in a high value of the autocorrelation function. One remaining issue is which frequency components to use when creating the histogram of the transform coefficients. We could use all coefficients or coefficients corresponding to just a single frequency band or of several frequencies. Following a trial-and-error approach, it was discovered that the higher frequencies are often all quantized to zero and do not contribute to the quantizer step size estimation. On the other hand, the DC coefficient is not a very good choice either, because the DC transform coefficients do not follow a Laplacian distribution and sometimes throw the measurement of *s* off. At the end, the coefficients corresponding to the (2, 2) AC frequency band were chosen for the histogram needed to estimate the value of *s*.

The method described above was tested for several different JPEG images with qualities ranging from 15 to 90 (PSNR of 30 to 40) and the scaling factor of the quantization matrix was estimated.

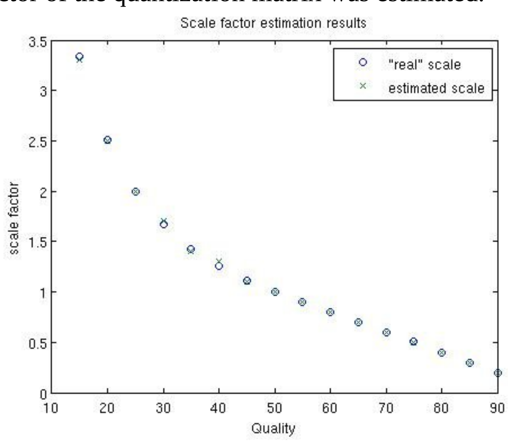


**Figure 7: Scalar multiple estimation results.**

The results displayed in Fig. 7 showed that *s* could be estimated very accurately for JPEG qualities in increments of 5 units.

Despite the successful measurement of the scaling factor, there are two problems with this approach. The quantization step sizes in the matrix Q are all integers. The values of Q obtained through this approach from equation (5) might be fractional due to a fractional value of *s*. These fractional values would be rounded off before proceeding to encode a raw image file. Thus, the effective value of *s* would not be the same for all frequency bands. Furthermore, due to round-off errors in the process of DCT computation, the transform coefficients extracted from the JPEG file will not be exact multiples of the values in *W*. As a result, the method described in this section to estimate *s* might not be accurate enough for the purpose of this project.

In order to account for these two problems, a slightly different method to estimate Q was implemented. First, it was assumed that nothing was known about the quantization matrix. Therefore, the standard JPEG matrix was not available. Two key modifications needed to be made to the procedure used to estimate *s*. Once the transform coefficients were obtained, they were not divided by the standard JPEG matrix each. Also, a separate histogram was created for each of the 64 frequency coefficients to estimate the respective step sizes. Therefore, instead of calculating a single *s* factor, the step size for each frequency was estimated separately.

For the low frequency coefficients this approach worked really well and the quantization step sizes were estimated exactly (with integer precision). However, there was a problem with the high frequency coefficients. Sometimes all the coefficients for a single frequency band would be quantized to zero. In those cases it is impossible to determine the quantization step size.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 16 | 11 | 10 | 16 | 24 | 40 | 51 | 61 |
| 12 | 12 | 14 | 19 | 26 | 58 | 60 | 0 |
| 14 | 13 | 16 | 24 | 40 | 57 | 0 | 0 |
| 14 | 17 | 22 | 29 | 0 | 0 | 0 | 0 |
| 18 | 22 | 0 | 0 | 0 | 0 | 0 | 0 |
| 24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Figure 8: Estimation of Q without the JPEG standard matrix. The zero values correspond to frequencies where all the coefficients were quantized to zero. Thus, their step sizes are unknown.**

The only way to obtain the high-frequency step sizes is to once again use the standard JPEG matrix. The value of *s* can be estimated from the known step size coefficients. The unknown ones can be filled with their corresponding values in the JPEG standard matrix multiplied by the estimated factor of *s*.

In the end, a hybrid approach was used to determine the quantization matrix *Q*. The low-frequency coefficients were determined directly, while the higher-frequency ones were estimated from the standard JPEG matrix scaled by a factor of "s", which was determined from the known coefficients.

# 5. USING J.240 VIDEO COMPRESSION STANDARD

This section is dedicated to a different approach of estimating PSNR based on the video coding standard J.240 [4]. J.240 is a standard for remote monitoring of video quality. It involves extracting a set of feature pixels from each video frame and sending it separately from the video stream. Those features extracted from the source video and received video respectively are then compared to estimate video quality.

Even though J.240 is used mainly for video encoding, it could also be used for image quality estimation. For instance, if the J.240 feature bitstream of the original image is available, it can be used to estimate the PSNR. The specifics of the J.240 feature extracting algorithm are described in [4]. It involves randomizing the original image block-wise and performing the Walsh-Hadamard transform. Then for each transformed 8x8 block, one feature pixel is selected. The MSE is then computed between the original set of feature pixels and the ones extracted from the JPEG image.

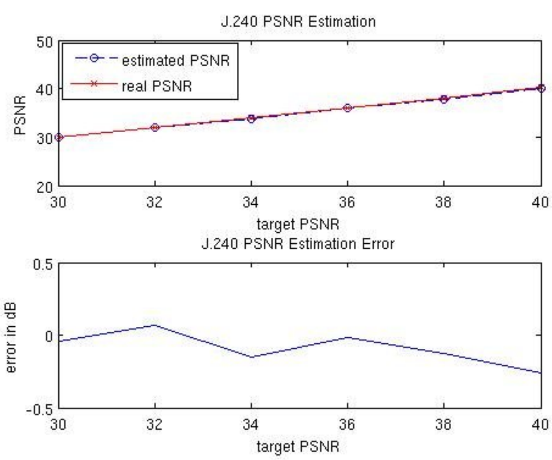


**Figure 9: Quality estimation using J.240 bitstream. The PSNR is estimated to within 0.5 dB as shown.**

Using an algorithm developed by Chono [5], the J.240 standard was applied on a set of JPEG images and the PSNR was calculated. It can be seen from the results in Fig. 9 that the J.240 algorithm provides a very accurate PSNR estimation to within 0.3 dB. The only drawback to it is that a J.240 feature bitstream of the original image needs to be available.

# 6. CONCLUSIONS

Methods for the estimation of JPEG image quality through the PSNR metric have been presented for cases where the original image data is not available. The AC DCT coefficients of natural images are approximated by Laplacian distributions. This Laplacian nature is exploited to enable the calculation of PSNR through the determination of the quantization step sizes used to compress the image. Techniques for determining the quantization step sizes have been shown to produce very accurate results. The lambda parameters for the Laplacian distributions were then calculated through two processes: the maximum-likelihood method and a compensation technique for improved accuracy for high quality images. PSNR results using a hybrid approach incorporating both the maximum-likelihood and compensation techniques was achieved to an accuracy of 1-2dB. In addition, the J.240 standard was used to calculate PSNR when the access to the feature bitstream is available. PSNR results using J.240 were within 0.3dB to the true PSNR.

# 7. REFERENCES

[1] T. Brandao and M. P. Qualuz, "Blind PSNR estimation of video sequences using quantized DCT coefficient data", Proc *Picture Coding Symp.* , Lisbon, Portugal, Vol. CDROM, pp. -- ---, November, 2007.

[2] ITU-T Rec. T.81, "Information Technology – Digital compression and coding of continuous-tone still images – Requirements and guidelines", Annex K, 1992.

[3] D.S. Turaga, C. Yingwei and J. Caviedes, "No reference PSNR estimation for compressed pictures," Proc. IEEE International Conference on Image Processing, vol.3, pp. 61-64, June 2002.

[4] ITU-T Rec J.240, "Framework for remote monitoring of transmitted picture signal-to-noise ratio using spread-spectrum and orthogonal transform", 2004.

[5] K. Chono, J.240 code provider, personal conversation, Stanford University, 2008.

## 8. DIVISION OF LABOR:

Scott Cottier:

- Lambda parameter estimation for Laplacian distribution of AC DCT coefficients
- PSNR estimation using quantization matrix and lambda values found before

Atanas Petkov:

- Quantization matrix estimation techniques
- J.240 investigation and application to JPEG images

Both:

- Paper and power point presentation preparation

## 9. ACKNOWLEDGEMENTS

We would like to thank David Varodayan and Keiichi Chono for their help and guidance for this project.