

A JPEG CORNER ARTIFACT FROM DIRECTED ROUNDING OF DCT COEFFICIENTS

Shruti Agarwal and Hany Farid

Department of Computer Science, Dartmouth College, Hanover, NH 03755, USA
 {shruti.agarwal.gr, farid}@dartmouth.edu

ABSTRACT

JPEG compression introduces a number of well known artifacts including blocking and ringing. We describe a lesser known or understood artifact consisting of a slightly darker or lighter pixel in the corner of 8×8 pixel blocks. This artifact is introduced by the directed rounding of DCT coefficients. In particular, we show that DCT coefficients that are uniformly rounded down or up (but not to the nearest neighbor) give rise to this artifact. An analysis of thousands of different camera models reveals that this artifact is present in approximately 61% of cameras. We also propose a simple filtering technique for removing this artifact.

Index Terms— JPEG Compression, JPEG Artifact

1. INTRODUCTION

The JPEG image standard is the most popular lossy compression scheme [1]. Despite its relatively high compression rates, JPEG compression introduces perceptual artifacts [2,3]. Most notably, *blocking artifacts* manifest themselves with a regular grid structure on an 8×8 pixel lattice and *ringing artifacts* manifest themselves with spatial aliasing that are particularly salient at high frequency edges.

We describe a less visually salient compression artifact—which we term JPEG dimples—that manifests as a slightly darker or lighter pixel in the top-left corner of 8×8 pixel blocks, Fig. 1. Although this artifact has previously been noted [4, 5], its root cause has not previously been explained. We describe the nature of this artifact, its prevalence in commercial cameras, and a simple filtering technique for removing this artifact.

The primary source of compression and information loss in the JPEG standard results from quantization of the discrete cosine transformed (DCT) coefficients [1]. Here, we are interested in the rounding operator used to convert DCT coefficients from floating-point to integer values. Three common rounding operators are: round to nearest integer (round-nearest), round down to nearest integer (round-down), and

round up to nearest integer (round-up). Although each of these operators converts from floating-point to integer values, each yields slightly different values. The round-down operator displaces all of the original values in one direction — towards $-\infty$. In contrast, the round-up operator displaces all of the original values towards $+\infty$. And, the round-nearest operator does not consistently displace values in one direction or another. We will show that the directional rounding performed by the round-down and round-up operators—but not the round-nearest operator—yields a compression artifact.

To see the nature of this artifact, consider the following 1-D example. Let \vec{s} be the following 1-D signal:

$$\vec{s} = (1.2 \quad 2.8 \quad 7.1 \quad 3.3 \quad 6.7 \quad 8.9). \quad (1)$$

For simplicity, we will quantize this 1-D signal with $q = 1$. The quantized values, as computed with the round-nearest operator, $\text{round}(\vec{s}/q) = [\vec{s}/q]$, are:

$$\vec{s}_n = (1 \quad 3 \quad 7 \quad 3 \quad 7 \quad 9). \quad (2)$$

The quantized values, as computed with the round-down operator, $\lfloor \vec{s}/q \rfloor$, and round-up operator, $\lceil \vec{s}/q \rceil$, are:

$$\vec{s}_d = (1 \quad 2 \quad 7 \quad 3 \quad 6 \quad 8) \quad (3)$$

$$\vec{s}_u = (2 \quad 3 \quad 8 \quad 4 \quad 7 \quad 9). \quad (4)$$

In this toy example, the relationship between the three quantized signals and the original signal are:

$$\begin{aligned} \vec{s}_n &= \vec{s} + (-0.2 \quad 0.2 \quad -0.1 \quad -0.3 \quad 0.3 \quad 0.1) \\ \vec{s}_d &= \vec{s} + (-0.2 \quad -0.8 \quad -0.1 \quad -0.3 \quad -0.7 \quad -0.9) \\ \vec{s}_u &= \vec{s} + (0.8 \quad 0.2 \quad 0.9 \quad 0.7 \quad 0.3 \quad 0.1). \end{aligned}$$

Notice that the values in \vec{s}_n are intermittently larger or smaller than the original signal \vec{s} . On the other hand, the values in \vec{s}_d are consistently smaller than the original signal \vec{s} and the values in \vec{s}_u are consistently larger than the original signal. To a first approximation, therefore, we can express the relationship between the results of the round-down and round-up operators as follows:

$$\vec{s}_d \approx \vec{s} - \alpha_d \vec{1} \quad (5)$$

$$\vec{s}_u \approx \vec{s} + \alpha_u \vec{1}, \quad (6)$$

This research was developed with funding from the Defense Advanced Research Projects Agency (DARPA FA8750-16-C-0166). The views, opinions, and findings expressed are those of the authors and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

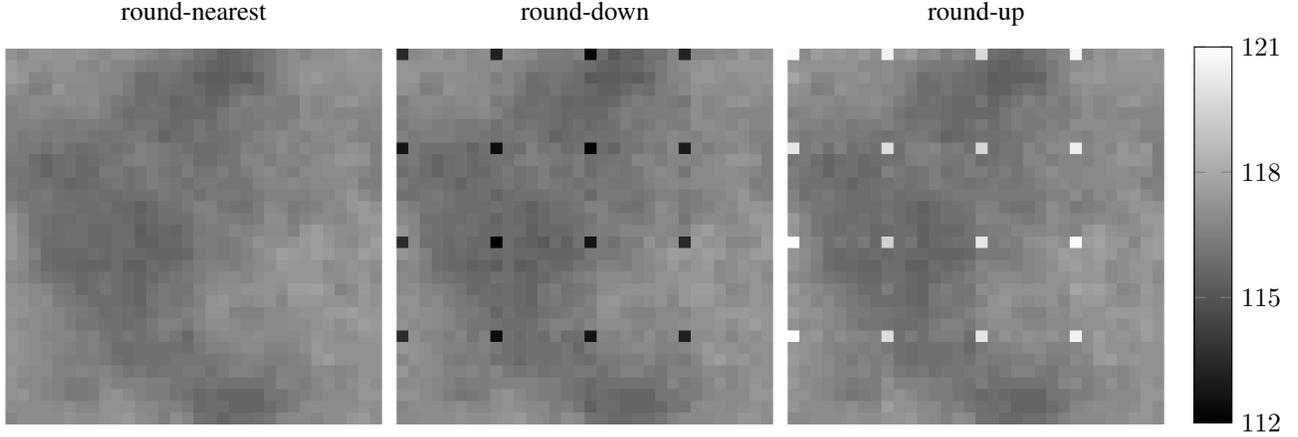


Fig. 1. Each panel shows a 32×32 intensity block computed by averaging all non-overlapping blocks from a fractal image. From left to right the image is JPEG compressed using the round-nearest, round-down, or round-up operator. The periodic JPEG dimples—a single dark or bright pixel in the upper left corner of each 8×8 pixel block—are introduced by the directed rounding operators but not by the round-nearest operator.

where $\vec{1} = (1 \ \dots \ 1)$ is a constant signal, α_d is the mean of $\vec{s}_d - \vec{s}$, and α_u is the mean of $\vec{s}_u - \vec{s}$.

Since this quantization is performed in the frequency domain, let's now consider the result of converting back into the spatial domain:

$$D^{-1}(\vec{s}_d) = D^{-1}(\vec{s} - \alpha_d \vec{1}), \quad (7)$$

where $D(\cdot)$ is the forward and $D^{-1}(\cdot)$ is the inverse DCT operator. Because of the linearity of the DCT, the right-hand side of this equation can be expressed as:

$$\begin{aligned} D^{-1}(\vec{s}_d) &= D^{-1}(\vec{s}) - \alpha_d D^{-1}(\vec{1}) \\ &= D^{-1}(\vec{s}) - \alpha_d \vec{\delta}, \end{aligned} \quad (8)$$

where the inverse DCT of a constant signal, $\vec{1}$, is an impulse $\vec{\delta}$.¹ The round-up operator yields a similar result except that impulse is now additive:

$$D^{-1}(\vec{s}_u) = D^{-1}(\vec{s}) + \alpha_u \vec{\delta}. \quad (9)$$

Due to the subtraction or addition of an impulse, the left-most value in $D^{-1}(\vec{s}_d)$ and $D^{-1}(\vec{s}_u)$ will be slightly smaller or larger than $D^{-1}(\vec{s}_n)$. In the 2-D case, this process is repeated for every 8×8 pixel block leading to a periodic artifact in which the top-left corner of each block is consistently dark (round-down) or light (round-up). We informally refer to this artifact as JPEG dimples.

Shown in Fig. 1 are three 32×32 intensity blocks computed by averaging all non-overlapping intensity blocks of a

synthetic image². From left to right, the image is compressed using a custom JPEG encoder with either the round-nearest, round-down, or round-up operator. The dimples, as predicted, are clearly visible in each 8×8 block and are darker for the round-down operator and brighter for the round-up operator, but are not introduced by the round-nearest operator. Although the JPEG dimples are clearly visible in the average intensity block, the artifact is not as salient in the absence of this averaging.

2. PREVALENCE

In this section we explore the prevalence of JPEG dimples in a wide range of commercial cameras. The presence or absence of dimples is determined by using a simple template-based approach. To begin, a 3-channel RGB image is partitioned into non-overlapping blocks of size $N \times N$ pixels (where N is a multiple of 8). A single average intensity block is computed by averaging all blocks across all three channels. This averaging makes the measurement of dimples more reliable by reducing the regularity of the underlying image content. A template of size $N \times N$, is then constructed in which the entire image is black (pixel value 0) except for a single unit impulse (pixel value 1) in the top left corner of every 8×8 pixel block. This template models the expected pattern of the JPEG dimples. The correlation between the template and the averaged block is computed using the peak to correlation energy (PCE) [5]. The absolute PCE value indicates the strength of dimples, with a larger value corresponding to a more prominent artifact.

¹Depending on the type of DCT transform (I, II, III, or IV) and the length of the signal, the impulse may contain some spatial ringing – we assume a DCT-I. The location of this impulse in the spatial domain is dictated by the phase of the constant signal in the frequency domain. In our case, this phase is zero and so the impulse is positioned at the left-most sample.

²A fractal image is generated in the frequency domain with a $1/\omega$ power spectrum and random phase

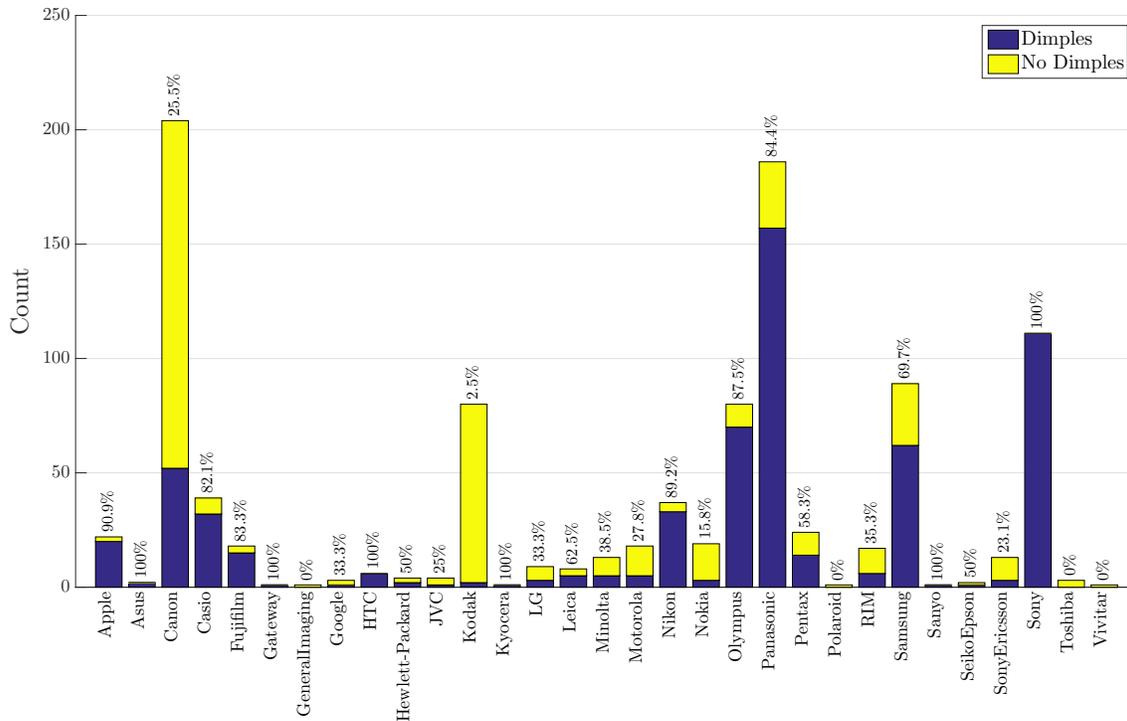


Fig. 2. The prevalence of JPEG dimples per camera manufacturer. Each bar corresponds to the total number of models per camera manufacturer. The portion of each bar shaded blue/yellow corresponds to those models with/without dimples. The numeric value above each bar corresponds to the percentage of models with dimples.

We performed two analyses to determine the prevalence of JPEG dimples in commercial cameras. For both analyses, approximately 40,000 unmodified images collected from Flickr were analyzed [6]. These images were acquired from 4,039 different camera configurations—defined as unique camera manufacturer, model, and capture resolution. The size N of the average block was fixed at 32×32 . A camera configuration with an absolute PCE greater than an empirically determined value of 15 is said to contain JPEG dimples.

For the first of the two analyses, we selected images from 1,017 of 4,039 camera configurations by considering configurations with maximum capture resolution afforded by a camera manufacturer and model (as determined by dpreview.com). Shown in Fig. 2 is the prevalence of dimples for each of 31 different camera manufacturers. For each camera manufacturer, we report the total number of camera models with (blue) and without (yellow) JPEG dimples. The length of each bar indicates the total number of models analyzed for that manufacturer.

Overall, 61% of camera models analyzed contain the JPEG dimple artifact. Images from Asus, HTC and Sony consistently contain dimples regardless of the camera model. Most models from a few other manufacturers (e.g., Apple, Fujifilm, Nikon, Olympus, and Panasonic) consistently introduce dimples. On the other hand, images from Kodak

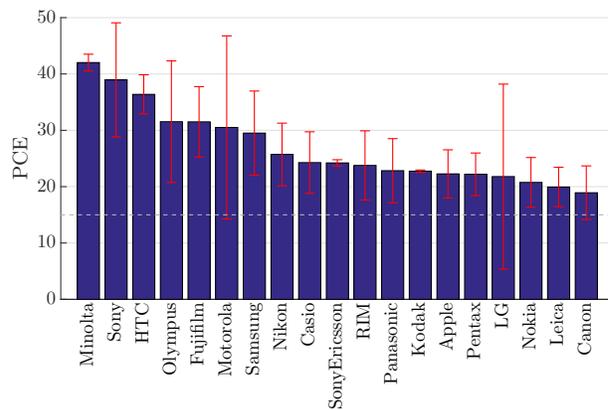


Fig. 3. The average strength of JPEG dimples per camera manufacturer. Each bar corresponds to the average PCE value for all available models per manufacturer and the error bars correspond to plus/minus one standard deviation.

cameras almost never contain dimples, except for two camera models. In between these extremes are, for example, Canon and Samsung in which the presence of dimples depends on the specific camera model.

In our second analysis, we observe that the strength—and presumably, therefore, the visual saliency—of the dim-

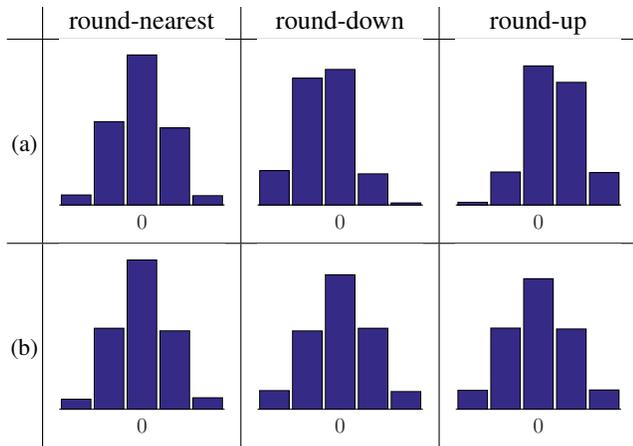


Fig. 4. The distributions along the first row correspond to a single AC frequency quantized with non-directional or directional rounding. The shift leftward and rightward introduced by the round-down and round-up operators lead to the JPEG dimple artifact. Shown in the second row are the distributions of this same AC frequency after removing the JPEG dimple artifact in which each distribution is now symmetric.

ple artifact varies by more than a factor of two across camera manufacturers. Shown in Fig. 3 is the average PCE observed for 19 camera manufacturers that have images from at least five different models that contain dimples. The average PCE ranges from a maximum of 42 (Minolta) to a minimum of 18 (Canon). We hypothesize that these variations are due to different optimized rounding implementations, but further ongoing work is required to fully confirm this hypothesis.

3. REMOVAL

We next describe a simple filtering technique for removing JPEG dimple artifacts. Shown in Fig. 4(a) is a representative distribution of a single AC frequency quantized with non-directional (nearest) and directional (down or up) rounding. As expected, the round-nearest distribution is zero-mean and symmetric about the origin [7, 8] while the round-down and round-up distributions are skewed with a negative and positive mean caused by the directional nature of the rounding. These skewed distributions give rise to the JPEG dimple artifact. We seek, therefore, to eliminate this skew in each AC frequency.

Denote μ as the mean of n AC coefficients at a single frequency quantized by an integer value q . We randomly choose $(\mu n)/q$ coefficients and shift them by an amount sq where s is $-\text{sign}(\mu)$. Note that with this strategy, coefficients are shifted by integer values so that the adjusted coefficients remain integers. Shown in Fig. 4 are the distributions before and after applying this adjustment. In each case, the adjusted distributions are zero-mean and symmetric.

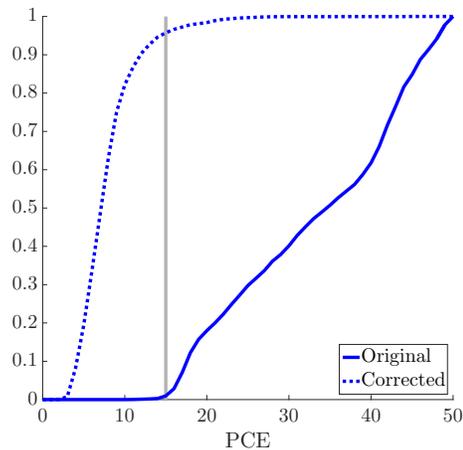


Fig. 5. Cumulative distribution of PCE values from 4,000 JPEG images before (solid blue) and after (dashed blue) dimple removal. The vertical line corresponds to our PCE threshold of 15. After removal 95.6% of images do not contain dimples ($\text{PCE} < 15$) as compared to 0.9% before removal.

As we will show next, this adjustment, when applied to all AC coefficients, results in removal of JPEG dimples from the image. We tested our removal technique on 4,000 JPEG images randomly selected from camera manufacturers that were found to have dimples. Shown in Fig. 5 is cumulative distribution of PCE values for these images before and after the dimple removal. After removal, the strength of the dimples in 95% of the images was reduced below the PCE detection threshold. At the same time, the average PSNR between the adjusted and original image is 52.1 db with a standard deviation of 1.8 db.

4. DISCUSSION

We have described a lesser known or understood JPEG artifact that results from the choice of mathematical operator used to convert DCT coefficients from floating-point to integer values. We argue that the presence of directed rounding during JPEG compression is the cause of this artifact, and have provided a theoretical and experimental validation to support this claim. The majority of commercial cameras that we analyzed introduce this artifact. Although not as perceptually salient as the better-known, and more visually salient, JPEG blocking and ringing artifacts, the JPEG dimple artifact described here can be avoided by simply using the round-nearest operator. We have also proposed a mechanism for the removal of dimples in JPEG images that are compressed using directed rounding. On the other hand, this artifact, as with other JPEG artifacts, can be exploited to authenticate digital images [9, 10].

5. REFERENCES

- [1] G. K. Wallace, “The JPEG still picture compression standard,” *Communications of the ACM*, vol. 34, no. 4, pp. 30–44, 1991.
- [2] M. Yuen and H. R. Wu, “A survey of hybrid MC/DPCM/DCT video coding distortions,” *Signal Processing*, vol. 70, no. 3, pp. 247–278, 1998.
- [3] M. A. Robertson and R. L. Stevenson, “DCT quantization noise in compressed images,” *IEEE Transactions Circuits Systems for Video Technology*, vol. 15, no. 1, pp. 27–38, 2005.
- [4] Y. L. Lee, H. C. Kim, and H. W. Park, “Blocking effect reduction of JPEG images by signal adaptive filtering,” *IEEE Transactions on Image Processing*, vol. 7, no. 2, pp. 229–234, 1998.
- [5] M. Goljan, J. Fridrich, and T. Filler, “Large scale test of sensor fingerprint camera identification,” in *Proceedings of SPIE, Electronic Imaging, Media Forensics and Security XI*, 2009, vol. 7254, pp. 72540I–72540I–12.
- [6] E. Kee, M. K. Johnson, and H. Farid, “Digital image authentication from JPEG headers,” *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 1066–1075, 2011.
- [7] M. C. Stamm, S. K. Tjoa, W. S. Lin, and K. J. R. Liu, “Anti-forensics of JPEG compression,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2010, pp. 1694–1697.
- [8] E. Y. Lam and J. W. Goodman, “A mathematical analysis of the DCT coefficient distributions for images,” *IEEE Transactions on Image Processing*, vol. 9, no. 10, pp. 1661–1666, 2000.
- [9] H. Farid, *Photo Forensics*, MIT Press, 2016.
- [10] S. Agarwal and H. Farid, “Photo forensics from JPEG dimples,” in *IEEE Workshop on Information Forensics and Security*, 2017.