Digital Image Authentication from Thumbnails

Eric Kee and Hany Farid

Department of Computer Science, Dartmouth College, Hanover NH 03755, USA

ABSTRACT

We describe how to exploit the formation and storage of an embedded image thumbnail for image authentication. The creation of a thumbnail is modeled with a series of filtering operations, contrast adjustment, and compression. We automatically estimate these model parameters and show that these parameters differ significantly between camera manufacturers and photo-editing software. We also describe how this signature can be combined with encoding information from the underlying full resolution image to further refine the signature's distinctiveness.

Keywords: Digital Image Forensics, Digital Image Authentication

1. INTRODUCTION

Many organizations are struggling with the issue of photo tampering. For example, digital images, videos, and audio are now routinely introduced as evidence in civil, criminal, and national security cases. In such cases, the integrity of digital evidence is central. Digital forensic techniques have been developed to detect: region duplication[1, 2]; inconsistencies in camera response function[3]; inconsistencies in lighting[4]; and inconsistencies in sensor noise[5] – see[6] for a general survey. However, relatively benign modifications such as cropping or contrast adjustment either cannot be detected by these techniques, or render these techniques ineffective. In the legal setting we are sometimes interested in determining if a digital image has been altered in any way from the time of recording, including something as simple as cropping. Here we approach this problem of image authentication by exploiting a camera's customized image encoding format.

We describe how to exploit the formation and storage of an embedded image thumbnail for image authentication. Specifically, we model the creation of a thumbnail with a series of filtering operations, contrast adjustment, and compression. We describe a technique for automatically estimating these model parameters, and show that these parameters, although not unique, can differ significantly between camera manufacturers and photo-editing software, Figure 1. As such, these parameters can be used as a signature for image authentication. We also describe how this signature can be combined with encoding information from the underlying full resolution image[7] to further refine the signature's distinctiveness.

2. METHODS

Embedded in an image header is a thumbnail version of the full resolution digital image. We describe a model for the creation of a thumbnail, and then describe how to estimate these model parameters.

2.1. Thumbnail Model

A thumbnail is typically on the order of 160×120 pixels in resolution. Given an image $f(x, y)$, the thumbnail is created by a series of six steps: crop, pre-filter, down-sample, post-filter, contrast and brightness adjustment, and JPEG compression. If the original resolution image is of a different aspect ratio than the thumbnail, then the image needs to either be padded or cropped accordingly. The amount of padding/cropping is specified by four parameters c_l, c_r, c_t, c_b, where c_l and c_r correspond to the padding/cropping on the left and right, and c_t and c_b on the top and bottom, Figure 2. A positive value corresponds to a padding (with a pixel value of 0), and a negative value corresponds to a cropping. Denoting the cropped image as $\hat{f}(x, y)$, the next four processing steps are specified as follows:

$$t(x, y) \quad = \quad \alpha \Big(D\{\hat{f}(x, y) \star h_1(x, y)\} \star h_2(x, y) \Big) + \beta, \tag{1}$$

Contact: kee@cs.dartmouth.edu and farid@cs.dartmouth.edu

Figure 1. Shown from left to right are an original thumbnail generated by a Nikon D200, a thumbnail of the same image generated by Adobe Photoshop CS3, and the difference between these thumbnails. The Nikon thumbnail is 160×120 pixels in size, while the Photoshop thumbnail is 160×107. In addition these JPEG encoded thumbnails employ different cropping boundaries and quantization tables leading to further differences as can be seen in the right-most panel.

where $t(x, y)$ is the thumbnail, $h_1(\cdot)$ is the pre-filter, $D\{\cdot\}$ is the down-sampling operator, $h_2(\cdot)$ is the post-filter, $\star$ is the convolution operator, and α and β are the multiplicative contrast and additive brightness adjustment terms, respectively. The pre-filter is typically a low-pass filter applied to avoid spatial aliasing prior to down-sampling, and the optional post-filter is typically a sharpening filter. In the final step, the thumbnail is JPEG compressed with a specified quantization table. In order to simplify this model, a series of assumptions are made:

- the pre-filter is assumed to be a circularly symmetric Gaussian, $\exp(-(x^2 + y^2)/\sigma^2)$, with width σ
- the pre-filter is unit sum
- the post-filter is 3×3 pixels in size
- the post-filter is symmetric ($h_2(x, y) = h_2(-x, y)$ and $h_2(x, y) = h_2(x, -y)$), yielding a filter of the form $(a \;\; b \;\; a \; ; \; b \;\; c \;\; b \; ; \; a \;\; b \;\; a)$
- the post-filter is unit-sum, constraining $c = 1 - (4a + 4b)$.

With these constraints, the full model for creating a thumbnail is specified by 11 processing parameters: 2 for the size of the thumbnail, 4 for the cropping/padding, 1 for the pre-filter, 2 for the post-filter, 2 for the contrast and brightness. In addition, there are 128 compression parameters: the JPEG quantization table is specified by two 8×8 tables corresponding to the quantization for the luminance and chrominance channels.* This yields a total of 139 model parameters. In the following section we describe how to estimate these model parameters.

2.2. Thumbnail Estimation

In the first step of thumbnail construction, a rectangular cropping boundary relative to the full resolution image is specified. This cropping boundary is determined by anisotropically scaling and translating a bounding box of the same size as the full resolution image such that the cropped and downsampled image matches the extracted thumbnail, Figure 2. The cropping parameters are estimated by first specifying an initial boundary that has the same aspect ratio as the thumbnail, is scaled to encompass the maximum image dimension, and is translated such that the image is centered within the boundary, Figure 2. The full resolution image $f(x, y)$ is then cropped according to this boundary, padded with zeros (where necessary), and downsampled to yield an initial thumbnail:

$$\hat{t}_0(x, y) \quad = \quad D\{\hat{f}_0(x, y)\}, \tag{2}$$

where $\hat{f}_0(x, y)$ is the initial cropped image, and the downsampling rate is $\max(N_x/n_x, N_y/n_y)$, where (N_x, N_y) and (n_x, n_y) are the image and thumbnail dimensions, respectively. This initial thumbnail is then anisotropically scaled and translated to match the extracted thumbnail $t(x, y)$:

$$t(x, y) \quad = \quad \hat{t}_0(s_x x + \Delta_x, s_y y + \Delta_y), \tag{3}$$

*We assume that the two chrominance channels employ the same quantization table.

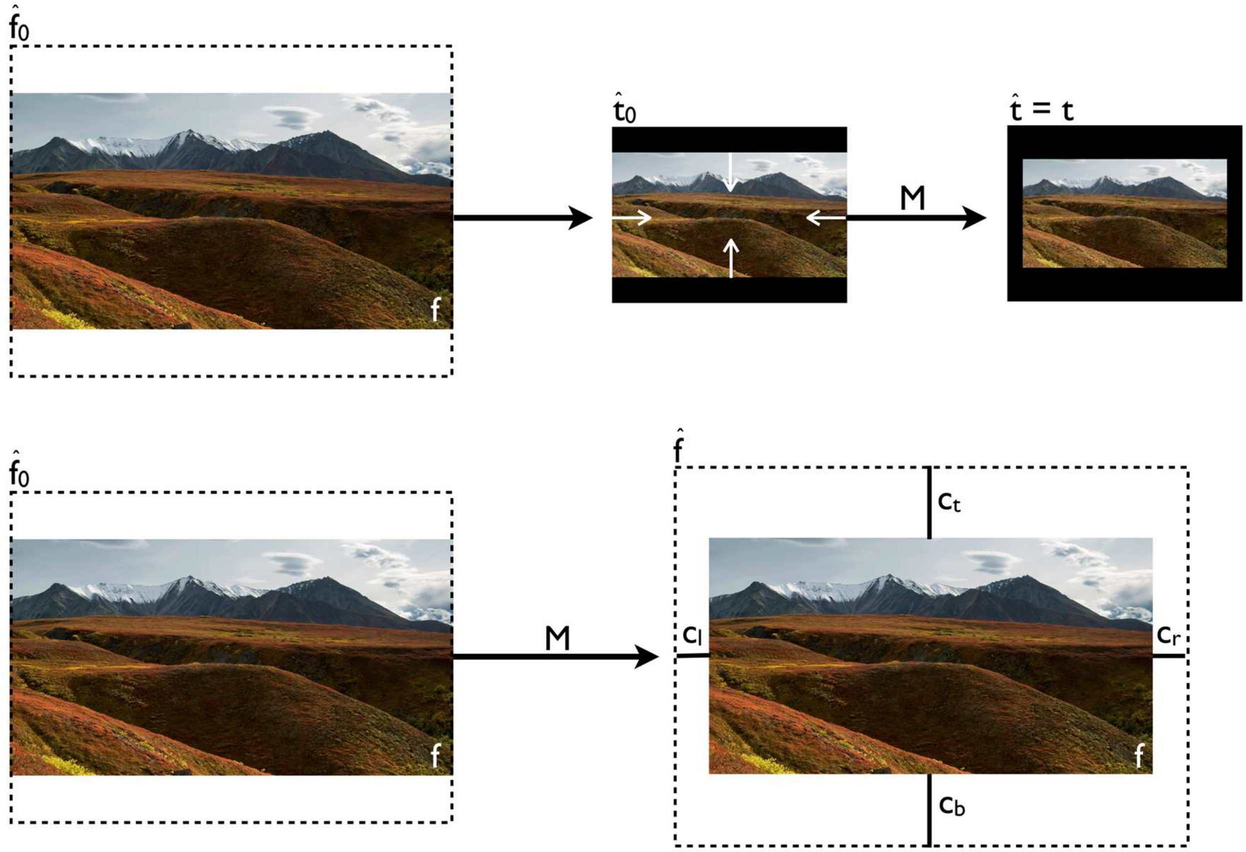

Figure 2. Creating the crop boundary for the full resolution image $f(x, y)$. Top row: create an initial crop boundary $\hat{f}_0(x, y)$ so that its aspect ratio matches the thumbnail $t(x, y)$; down-sample to create $\hat{t}_0(x, y)$; align with the actual thumbnail $t(x, y)$ by scaling and translating by $M = (s_x, s_y, \Delta_x, \Delta_y)$ to yield $\hat{t}(x, y)$. Bottom row: adjust the initial crop boundary $\hat{f}_0(x, y)$ by M to yield the desired crop boundary $\hat{f}(x, y)$. The crop boundary is specified by the margins c_l, c_r, c_t, c_b.

where (s_x, s_y) and (Δ_x, Δ_y) are the scaling and translation parameters. These parameters are estimated using a coarse-to-fine differential registration technique.[8,9] This registration is performed on a grayscale version of the color thumbnail. The scaling and translation parameters are then used to specify the cropping boundary for the full resolution image. Specifically, the coordinates of the four corners of the initial cropped image $\hat{f}_0(x, y)$ are scaled and translated[†] by (s_x, s_y) and (Δ_x, Δ_y), yielding the desired rectangular cropping boundary $\hat{f}(x, y)$, Figure 2. This boundary is parametrized by the horizontal and vertical margins c_l, c_r, c_t, and c_b, specified as a percentage of the image dimensions, Figure 2. Note that when down-sampling $\hat{f}(x, y)$, the sampling rate in the horizontal and vertical directions must be independently adjusted so that the final dimensions match that of the thumbnail $t(x, y)$.

Next, the pre-filter $h_1(\cdot)$, post-filter $h_2(\cdot)$, and contrast and brightness terms α and β are estimated by minimizing the following error function:

$$E_1(h_1, h_2) = \sum_{x,y} \left[t(x, y) - \alpha \left(D\{\hat{f}(x, y) \star h_1(x, y)\} \star h_2(x, y) \right) - \beta \right]^2 . \tag{4}$$

[†]Because the translation parameters (Δ_x, Δ_y) are specified in thumbnail coordinates, they must be scaled by the downsampling rate between $\hat{f}_0(x, y)$ and $\hat{t}_0(x, y)$.

Note that this error function is only specified in terms of the pre- and post-filter, and not the contrast and brightness. Given a pair of filters, the contrast and brightness are estimated by minimizing the following error function:

$$E_2(\alpha, \beta) \quad = \quad \sum_{x,y} \left[t(x,y) - (\alpha \hat{t}_h(x,y) + \beta) \right]^2, \tag{5}$$

where $\hat{t}_h(x,y) = D\{\hat{f}(x,y) \star h_1(x,y)\} \star h_2(x,y)$. This error function is minimized using standard least-squares estimation. The summation in $E_1(\cdot)$ and $E_2(\cdot)$ are performed over all three color channels of the thumbnail. The error function $E_1(\cdot)$ is minimized using a brute-force search over the filter parameters, where on each iteration of this search, the error function $E_2(\cdot)$ is minimized analytically to yield the best contrast and brightness terms for the specified filters. These parameters are refined using an iterative Nelder-Mead minimization, which is bootstrapped with the results of the brute-force minimization.

In practice, we have found that the minimization of $E_1(\cdot)$ is slightly more effective when performed in the Fourier domain. Specifically, we minimize:

$$E_1(h_1, h_2) \quad = \quad \sum_{\omega_x, \omega_y} W(\omega_x, \omega_y) \left(|\mathcal{T}(\omega_x, \omega_y)| - |\hat{\mathcal{T}}(\omega_x, \omega_y)| \right)^2, \tag{6}$$

where $\mathcal{T}(\cdot)$ is the Fourier transform of the actual thumbnail $t(\cdot)$, $\hat{\mathcal{T}}(\cdot)$ is the Fourier transform of our constructed thumbnail $\alpha(D\{\hat{f} \star h_1\} \star h_2) + \beta$, and $|\cdot|$ denotes the magnitude of the Fourier transform. This error function is frequency weighted with a highpass filter, $W(\cdot)$, because after the initial alignment of Equation (3) the low-frequencies are already well aligned.

Lastly, the thumbnail dimensions and 128 thumbnail compression parameters are extracted directly from the JPEG header. Note that since the original thumbnail $t(\cdot)$ was compressed with these parameters, we must also compress our estimates of the thumbnail during the parameter estimation stages described above. Specifically, $\hat{t}_0(\cdot)$ is compressed prior to estimating the scaling and translation parameters, and $\hat{t}_h(\cdot)$ is compressed prior to estimating the contrast and brightness terms.

3. RESULTS

To verify that thumbnail parameters can be reliably used for image authentication, a database of 1514 unmodified images were downloaded from `Flickr`. An image was considered to be unmodified if the image was tagged by `Flickr` as "original", no metadata fields had been removed, the metadata "modification" and "original" dates were consistent, and the metadata "software" field was empty. These 1514 images spanned 142 cameras of different make and model, Appendix A. Cameras of the same make and model sometimes vary their image and thumbnail size and quantization table[‡]. Because these variations affect the overall thumbnail parametrization, we partitioned the 1514 images into 245 "camera classes" containing images from the same make and model and image/thumbnail size and quantization table.

For computational efficiency the thumbnail parameters were first estimated for a representative image from each of the 245 camera classes. The crop boundary was estimated as described in the previous section. The pre-filter, post-filter, and contrast and brightness were estimated using a brute-force search, Equation (4). The parameter space for the pre-filter was partitioned into the range $[0.005, 1]$ in 20 equal steps.[§] The parameter space for the two components of the post-filter were partitioned into the range $[-0.5, 0.5]$ in 11 equal steps. Recall that the contrast and brightness terms were estimated analytically within each iteration of the brute force search, Equation (5). These brute force solutions were then used to bootstrap the Nelder-Mead minimization for the remaining images within each camera class.

[‡]Variations in image/thumbnail size and quantization table can, for example, be due to differences in firmware.

[§]In order to appropriately compare the width σ of the pre-filter $h_1(\cdot)$ across images of varying size, the size of the pre-filter $h_1(\cdot)$ is normalized relative to the image dimensions. Specifically, the pre-filter is sampled over the interval $[-1, 1]$ at a rate 1/150 of the maximum dimension of the original image.

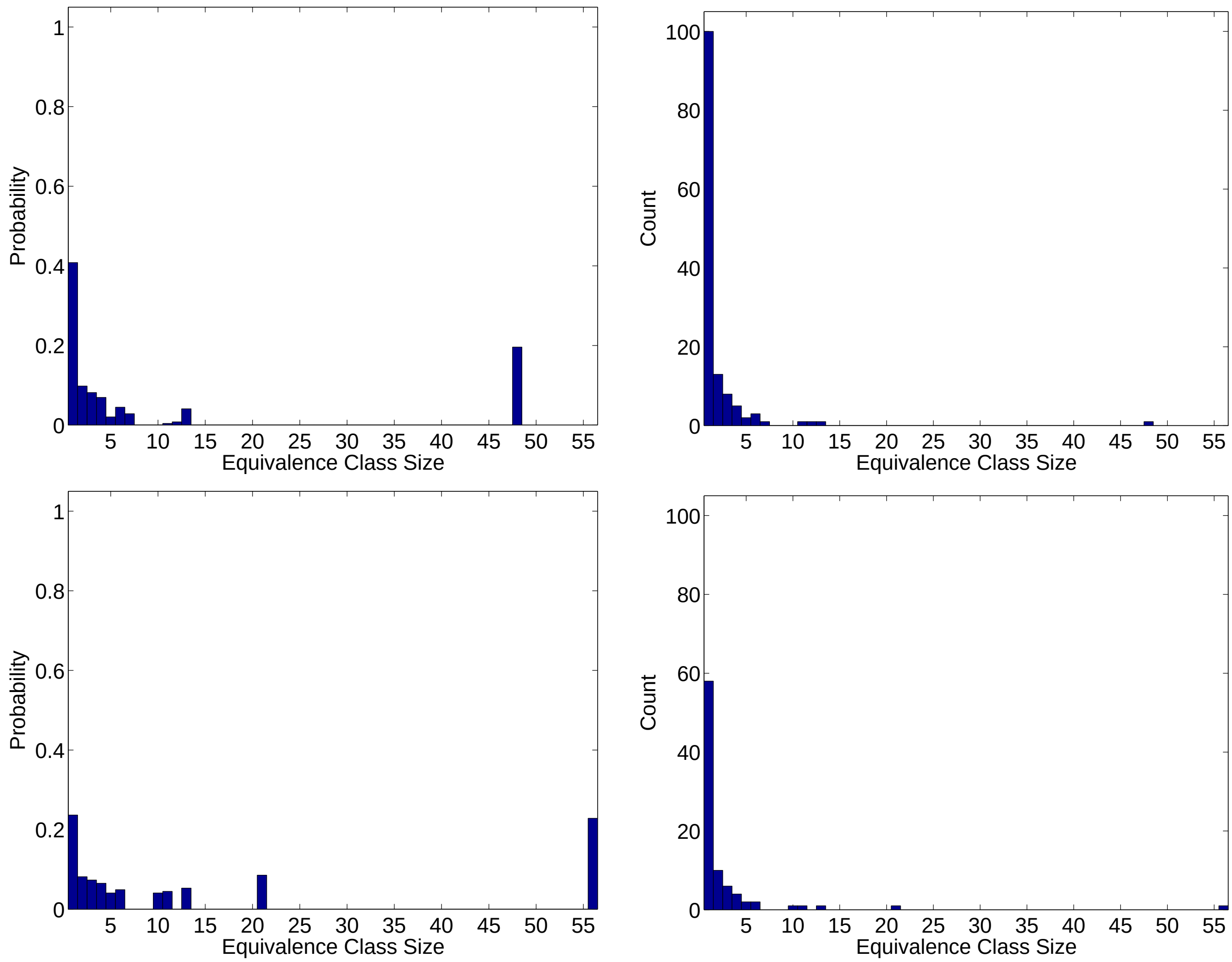

Figure 3. Shown in the left column is the probability that a camera is contained in an equivalence class of a specified size. Shown in the right column is the number of equivalence classes of a specified size. Shown along the top row are the distributions based on all of the thumbnail parameters (40.8% of the cameras have unique parameters, i.e., an equivalence class size of one). Shown in the bottom row are the distributions for only the thumbnail size and quantization table (23.7% of the cameras have unique parameters).

Because the absolute value of the contrast and brightness terms may depend on the underlying image content, these parameters are combined into a single binary-valued parameter corresponding to the presence or absence of contrast/brightness adjustment. Specifically, a camera is said to apply a contrast/brightness adjustment if the camera's contrast deviates by more than 0.075 from unity, or if the brightness deviates by more than 0.05 from zero (assuming a luminance scale of $[0, 1]$). These thresholds were determined by first computing the difference between each image's thumbnail parameter and the mean parameter of the corresponding camera class. The threshold was then taken to be the width of the 99% confidence interval of the distribution of these differences.

We next evaluate the consistency and distinctness of the thumbnail parameters within and across camera classes. The thumbnail parameters for a given camera class are determined by computing the mean of the crop boundary, the mean of the pre- and post-filters, and the mode of the contrast/brightness, across all images in a camera class. The thumbnail size and quantization table are, by definition, the same for all images in a camera class. In order to reasonably equate real-valued model parameters (crop boundary, pre-filter, post-filter), these parameters were considered as equivalent if their absolute difference was below a specified threshold. The

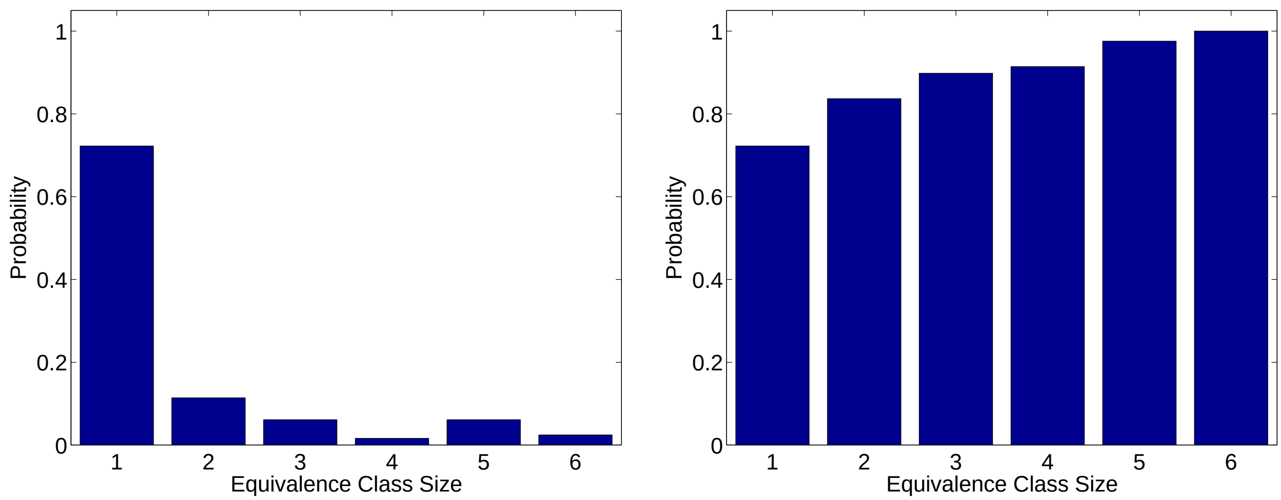

Figure 4. Shown on the left is the probability that a camera is contained in an equivalence class of a specified size based on all of the image and thumbnail parameters (72.2% of the cameras have unique parameters, i.e., an equivalence class size of one). Shown on the right is the cumulative probability distribution.

threshold for the crop boundary was 0.01, the pre-filter 0.2, and the post-filter 0.075. These thresholds were determined in the same way as for the contrast and brightness thresholds described above.

Two thumbnail models were considered to be equivalent if their integer-valued parameters (contrast/brightness, size, quantization table) were the same and if their real-valued parameters (crop boundary, pre-filter, post-filter) were within the specified thresholds described above. Cameras were then grouped into equivalence classes of identical models.

Shown in the top left panel of Figure 3 is the probability that a camera is contained in an equivalence class of a specified size based on using all of the thumbnail parameters: 40.8% of the cameras are in an equivalence class of size one (i.e., are unique), 9.8% are in an equivalence class of size two, 8.2% are in an equivalence class of size three, and the largest equivalence class is of size 48, with 19.6% of the cameras. There is only one class of size 48 and it contains 42 Canon Powershots and 6 Canon Digital Ixus of varying models. Shown in the top right panel of Figure 3 is the distribution of equivalence class sizes (i.e., the number of equivalence classes of each size). For comparison, shown in the bottom row of Figure 3 are the distributions for only the thumbnail size and quantization table: 23.7% of the cameras are in an equivalence class of size one, 8.25% are in an equivalence class of size two, 7.4% are in an equivalence class of size three, and the largest equivalence class is of size 56, with 22.9% of the cameras. Note that the addition of the thumbnail processing parameters improves the distinctiveness of the signature.

Shown in Figure 4 is the probability that a camera is contained in an equivalence class of a specified size based on using all of the thumbnail parameters and the full resolution image size and quantization table. In this case, 72.2% of the cameras are in an equivalence class of size one (i.e., are unique), 11.4% are in an equivalence class of size two, 6.1% are in an equivalence class of size three, and the largest equivalence class is of size 6, with 2.5% of the cameras. The addition of the image parameters significantly improves the distinctiveness of the signature. A closer look at these equivalence classes reveals some interesting trends. There is one equivalence class of size 6 and it contains six 8-megapixel Canon Powershot cameras of varying models. There are three equivalence classes of size 5: one contains 8-megapixel Canon Powershot and Canon Ixus cameras and the remaining classes each contain a mixture of 7-megapixel Canon Powershot and Canon Ixus cameras of varying models. With only two exceptions, the remaining equivalence classes of size greater than 1 are populated with Canon cameras which seem to be particularly consistent in their thumbnail and image parameters. See Appendix A for a complete break-down of these distributions.

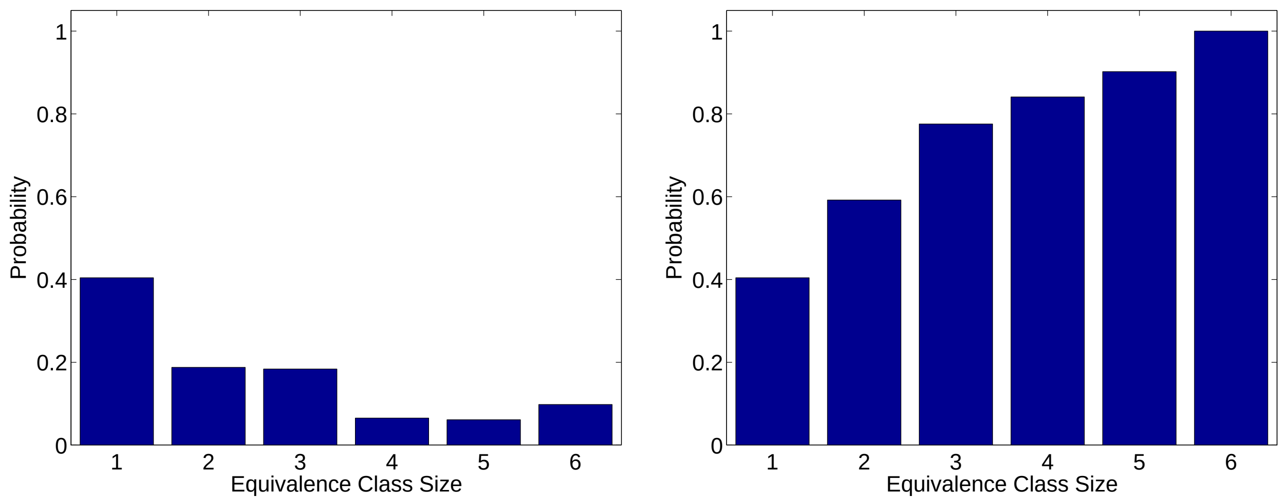

Figure 5. Shown on the left is the probability that a camera is contained in an equivalence class of a specified size based on only the image size and quantization table (40.4% of the cameras have unique parameters, i.e., an equivalence class size of one). Shown on the right is the cumulative probability distribution.

Lastly, shown in Figure 5 is the probability distributions based on using only the full resolution image size and quantization table. In this case, 40.4% of the cameras are in an equivalence class of size one, 18.8% are in an equivalence class of size two, 18.4% are in an equivalence class of size three, and the largest equivalence class is of size 6, with 9.8% of the cameras. Note that the addition of the thumbnail parameters, Figure 4, significantly improves the distinctiveness of the signature.

To further test the utility of thumbnail parameters for authentication, we analyzed the thumbnail parameters used by Photoshop (CS3). An image was saved with Photoshop at each of 13 possible JPEG quality settings. The thumbnail and image parameters were estimated for each image and compared to the remaining 245 camera classes. None of the Photoshop thumbnail parameters were shared by any of the camera classes. We also individually compared the Photoshop parameters to the camera classes: None shared the same image quantization table, thumbnail quantization table, or pre/post filter; eleven of the camera classes used the same crop parameters: nikon d300 and coolpix l4; fujifilm finepix s5700/s700, s100fs, and z20fd; canon powershot sd400 and s2 is; casio ex-z77, ex-z1050, ex-s880, and ex-z60; the olympus e-510 and the sony dsc-w120.

4. DISCUSSION

An image thumbnail is created by a series of six steps: crop, pre-filter, down-sample, post-filter, contrast and brightness adjustment, and JPEG compression. We have described how these parameters can be estimated, and have shown that they vary significantly between camera manufacturers and photo-editing software. The distinctiveness of these parameters becomes even more pronounced when they are combined with the full resolution image size and compression parameters. As such, these thumbnail and image parameters can be used for image authentication (determining if an image was altered in any way from the time of its recording).

The effectiveness of this approach depends on building a library of thumbnail and image parameters extracted from a large collection of images. We have collected 1.2 million "original" images from which we plan to build this library. From our initial tests, we have found that 83% of the images have a thumbnail. For the images without thumbnails, the full resolution image parameters can still be used for authentication.

ACKNOWLEDGMENTS

This work was supported by a gift from Adobe Systems, Inc., a gift from Microsoft, Inc., and a grant from the National Science Foundation (CNS-0708209).

REFERENCES

1. J. Fridrich, D. Soukal, and J. Lukáš, "Detection of copy move forgery in digital images," in *Proceedings of Digital Forensic Research Workshop*, August 2003.
2. A. Popescu and H. Farid, "Exposing digital forgeries by detecting duplicated image regions," Tech. Rep. TR2004-515, Department of Computer Science, Dartmouth College, 2004.
3. Z. Lin, R. Wang, X. Tang, and H.-V. Shum, "Detecting doctored images using camera response normality and consistency," in *Computer Vision and Pattern Recognition*, (San Diego, CA), 2005.
4. M. Johnson and H. Farid, "Exposing digital forgeries in complex lighting environments," *IEEE Transactions on Information Forensics and Security* **3**(2), pp. 450–461, 2007.
5. J. Lukáš, J. Fridrich, and M. Goljan, "Digital camera identification from sensor noise," *IEEE Transactions on Information Security and Forensics* **1**(2), pp. 205–214, 2006.
6. H. Farid, "A survey of image forgery detection," *IEEE Signal Processing Magazine* **2**(26), pp. 16–25, 2009.
7. H. Farid, "Digital image ballistics from JPEG quantization: A followup study," Tech. Rep. TR2008-638, Department of Computer Science, Dartmouth College, 2008.
8. E. Simoncelli, *Handbook of Computer Vision and Applications*, ch. Bayesian Multi-scale Differential Optical Flow, pp. 397–420. Academic Press, 1999.
9. H. Farid and J. Woodward, "Video stabilization and enhancement," Tech. Rep. TR2007-605, Department of Computer Science, Dartmouth College, 2007.

Appendix A

Each entry in this table corresponds to an equivalence class of identical thumbnail and image parameters, Figure 4. The size of the equivalence class is in the first column and the make, model, and resolution are in the second column. For compactness, all cameras in an equivalence class of size one are listed in a single entry – for these cameras, the parenthetical following the make and model corresponds to the number of instances of the same make and model with different thumbnail and image parameters.

6	canon powershot sd870 is (8mp) – canon powershot sd850 is (8mp) – canon powershot s80 (8mp) – canon powershot a590 is (8mp) – canon powershot sx100 is (8mp) – canon powershot a720 is (8mp)
5	canon powershot sd1100 is (8mp) – canon powershot s5 is (8mp) – canon powershot a630 (8mp) – canon powershot a580 (8mp) – canon digital ixus 80 is (8mp)
5	canon powershot sd750 (7mp) – canon powershot sd800 is (7mp) – canon powershot a570 is (7mp) – canon powershot sd550 (7mp) – canon digital ixus 850 is (7mp)
5	canon powershot a710 is (7mp) – canon digital ixus 70 (7mp) – canon powershot a620 (7mp) – canon powershot tx1 (7mp) – canon powershot a550 (7mp)
4	canon powershot a530 (5mp) – canon powershot a610 (5mp) – canon powershot sd400 (5mp) – canon powershot a460 (5mp)
3	canon powershot sd600 (6mp) – canon powershot a540 (6mp) – canon powershot sd630 (6mp)
3	canon eos digital rebel xti (10.1mp) – canon eos 400d digital (10.1mp) – canon eos kiss digital x (10.1mp)
3	canon eos digital rebel(6.3mp) – canon eos 300d digital (6.3mp) – canon eos 10d (6.3mp)
3	sony dsc-w55 (7.2mp) – sony dsc-h5 (7.2mp) – sony dsc-w35 (7.2mp)
3	canon powershot s3 is (6mp) – canon digital ixus 800 is (6mp) – canon digital ixus 60 (6mp)
2	canon powershot sd600 (6mp) – canon powershot s3 is (6mp)
2	canon eos digital rebel xt (8mp) – canon eos 350d digital (8mp)
2	canon eos 30d (8.2mp) – canon eos 20d (8.2mp)
2	canon eos digital rebel xsi (12.2mp) – canon eos 450d (12.2mp)
2	canon eos digital rebel xsi (12.2mp) – canon eos 450d (12.2mp)
2	canon powershot a510 (3.2mp) – canon powershot a95 (5mp)
2	canon powershot sd450 (5mp) – canon powershot s2 is (5mp)
2	canon powershot g9 (12.1mp) – canon powershot sd950 (12.1mp)
2	canon powershot a520 (4mp) – canon powershot a85 (4mp)
2	canon powershot a75 (3.2mp) – canon powershot sd200 (3.2mp)
2	canon powershot sd1000 (7.1mp) – canon powershot s230 (3.1mp)
2	canon powershot a640 (8mp) – canon powershot sd790 is (10.1mp)
2	kodak easyshare m1033 (10mp) – kodak easyshare z1012 (10mp)
2	canon powershot g7 (10mp) – canon powershot sd900 (10mp)

1	apple iphone (2x) – camera 8mp-9hx (1x) – canon digital ixus 30 (1x) – canon digital ixus 50 (1x) – canon digital ixus 700 (1x) – canon digital ixus 75 (1x) – canon eos 20d (1x) – canon eos 30d (1x) – canon eos 400d digital (5x) – canon eos 40d (5x) – canon eos 5d (1x) – canon eos digital rebel (1x) – canon eos digital rebel xsi (1x) – canon eos digital rebel xt (5x) – canon eos digital rebel xti (3x) – canon eos kiss x2 (1x) – canon powershot a400 (1x) – canon powershot a560 (2x) – canon powershot a650 is (1x) – canon powershot a70 (1x) – canon powershot a80 (1x) – canon powershot a95 (1x) – canon powershot g9 (1x) – canon powershot s2 is (1x) – canon powershot s30 (1x) – canon powershot sd1000 (3x) – canon powershot sd1100 is (1x) – canon powershot sd300 (1x) – canon powershot sd400 (1x) – canon powershot sd600 canon powershot sd700 is (1x) – canon powershot sd750 (1x) – casio ex-s880 (1x) – casio ex-z1050 (2x) – casio ex-z60 (1x) – casio ex-z77 (1x) – casio ex-z80 (1x) – casio lex-z850 (1x) – digital camera 6mp-9fw (1x) – fujifilm finepix s100fs (1x) – fujifilm finepix s2000hd (1x) – fujifilm finepix s5700 s700 (7x) – fujifilm finepix z20fd (1x) – hp photosmart m537 (1x) – hp photosmart r717 (1x) – hp photosmart r740 (1x) – konica minolta dimage z3 (1x) – nikon coolpix l4 (1x) – nikon d100 (1x) – nikon d200 (4x) – nikon d300 (5x) – nikon d3 (2x) – nikon d40 (14x) – nikon d40x (2x) – nikon d50 (12x) – nikon d60 (4x) – nikon d70 (4x) – nikon d70s (4x) – nikon d80 (15x) – nikon d90 (1x) – olympus e-500 (2x) – olympus e-510 (2x) – olympus u1030sw,s1030sw (1x) – olympus u850sw,s850sw (1x) – panasonic dmc-fx9 (1x) – panasonic dmc-fz18 (1x) – panasonic dmc-fz20 (1x) – panasonic dmc-fz5 (1x) – panasonic dmc-fz7 (1x) – panasonic dmc-fz8 (1x) – panasonic dmc-tz1 (1x) – panasonic dmc-tz3 (1x) – panasonic dmc-tz4 (1x) – panasonic dmc-tz5 (1x) – pentax k100d (1x) – pentax k100d super (1x) – pentax k10d (3x) – pentax optio s5z (1x) – samsung techwin <kenox s630/samsung s630> (1x) – sanyo vpc-s600 (1x) – sanyo vpc-t700 (1x) – sony dsc-p200 (1x) – sony dsc-p92 (1x) – sony dsc-t300 (1x) – sony dsc-t70 (1x) – sony dsc-t9 (1x) – sony dsc-w120 (1x) – sony dsc-w50 (1x) – sony dsc-w55 (1x) – sony dsc-w70 (1x) – sony dsc-w80 (1x) – sony dslr-a100 (1x) – sony dslr-a200 (2x) – sony dslr-a350 (1x)