



Digital Forensics

MO447 / MC919

Prof. Dr. Anderson Rocha

*Microsoft Research Faculty Fellow
Affiliate Member, Brazilian Academy of Sciences
Reasoning for Complex Data (Recod) Lab.*

anderson.rocha@ic.unicamp.br
<http://www.ic.unicamp.br/~rocha>

Reasoning for Complex Data (RECOD) Lab.
Institute of Computing,
University of Campinas (Unicamp)

Av. Albert Einstein, 1251 – Cidade Universitária
CEP 13083-970 • Campinas/SP – Brasil

View from the window at Le Gras – Joseph Niepce (GoogleGram)

Organization

Organization

- ▶ Recap.
- ▶ Image Authentication
- ▶ *Thumbnails*
- ▶ How to estimate a thumbnail?
- ▶ Experiments & Validation
- ▶ Final Remarks

Recap.!

Introduction

▶ What is Computational/Digital Forensics?

“ Collection of scientific techniques for the preservation, collection, validation, identification, analysis, interpretation, documentation, and presentation of digital evidence derived from digital sources for the purpose of facilitating or furthering the reconstruction of events, usually of a criminal nature. ”

Edward Delp – Purdue University

Introduction

- ▶ What is **Multimedia Forensics**
 - Subfield of Digital Forensics
 - Objectives
 - ▶ Source Attribution
 - ▶ Authenticity Verification
 - ▶ Reconstruction of Manipulation Events in Multimedia Objects

Image Authentication

Based on “*Digital Image Authentication from Thumbnails*” **Kee, E., Farid, H.** SPIE Symposium on Electronic Imaging (SEI), pp. 1–10, San Jose, USA, 2010



Image Authentication

- ▶ What is authentication?
- ▶ Can we do it via **Watermarks**?
- ▶ What are the implications (pros/cons)?

Image Authentication

- ▶ Digital images, videos and audio recordings are ever-present in our daily lives and are paramount source of evidence in many legal cases
- ▶ In these cases, **integrity** is crucial

Image Authentication

- ▶ Many approaches have been proposed in the literature for detecting manipulations as we will cover in this class
 - Copy/Move detection
 - Inconsistencies in acquisition, illumination, shadows, structure, compression etc.
- ▶ However, such methods focus on manipulation detection, source attribution or on separating CG images and photographs

Image Authentication

- ▶ In authentication, different from manipulation detection, we are interested in
 - Pointing out whether **an image is authentic or not**
 - In this case, the very presence of a modification be it as simple or small as possible, renders the image not authentic anymore

Image Authentication

- ▶ [Avcibas et al. 2006] have presented a technique based on ML and image descriptor fusion for determining authenticity
 - Image Quality Metrics (IQM)
 - Binary Similarity Measures (bit planes) (BSMs)
 - *Wavelet* high-order descriptors (HOWS)

Image Authentication

- ▶ [Kee & Farid 2010] have presented a technique based on the formation and storage of thumbnails in JPEG images

Image Authentication

- ▶ What is a thumbnail?
 - It is a pictorial representation of a higher resolution image
 - Typically 160×120 pixels
 - Stored along with the image

Image Authentication

- ▶ [Kee & Farid 2010] model the *thumbnail* creation process as a series of operations involving **filtering, contrast/brightness adjustments and compression**
- ▶ The authors present an approach for estimating the thumbnails generation process parameters automatically

Image Authentication

- ▶ What's the matter with all of this?
 - These parameters, **although not unique**, are different among camera models and image-editing software packages
 - In this way, we can use these parameters to detect inconsistencies in the image finding an image signature useful for use in **Image Authentication**

Thumbnails

Thumbnails

- ▶ Given an image $f(x, y)$ we can create its thumbnail in roughly six steps

1. Cropping
2. Pre-Filtering
3. Down-Sampling
4. Post-Filtering
5. Brightness/Contrast adjustments
6. JPEG compression

Thumbnails

- ▶ If the input image has a different aspect ratio, we perform cropping or zero-padding
- ▶ The amount of cropped or padded pixels to the left, right, above and below is given by four parameters
- ▶ c_l, c_r, c_t, c_b
- ▶ Value + (*padding*), Value – (*cropping*)

Thumbnails

- ▶ Denoting the cropped image as $\hat{f}(x, y)$, the next four steps consist of

$$t(x, y) = \alpha \left(D \{ \hat{f}(x, y) \star h_1(x, y) \} \star h_2(x, y) \right) + \beta$$

Diagram illustrating the thumbnail generation process:

- Thumbnail** (output) is derived from $t(x, y)$.
- Contrast** is represented by α .
- Downsampling** is represented by D .
- Cropped Image** is represented by $\hat{f}(x, y)$.
- Convolution** is represented by \star .
- Pre-Filtering** is represented by $h_1(x, y)$.
- Post-Filtering** is represented by $h_2(x, y)$.
- Brightness** is represented by β .

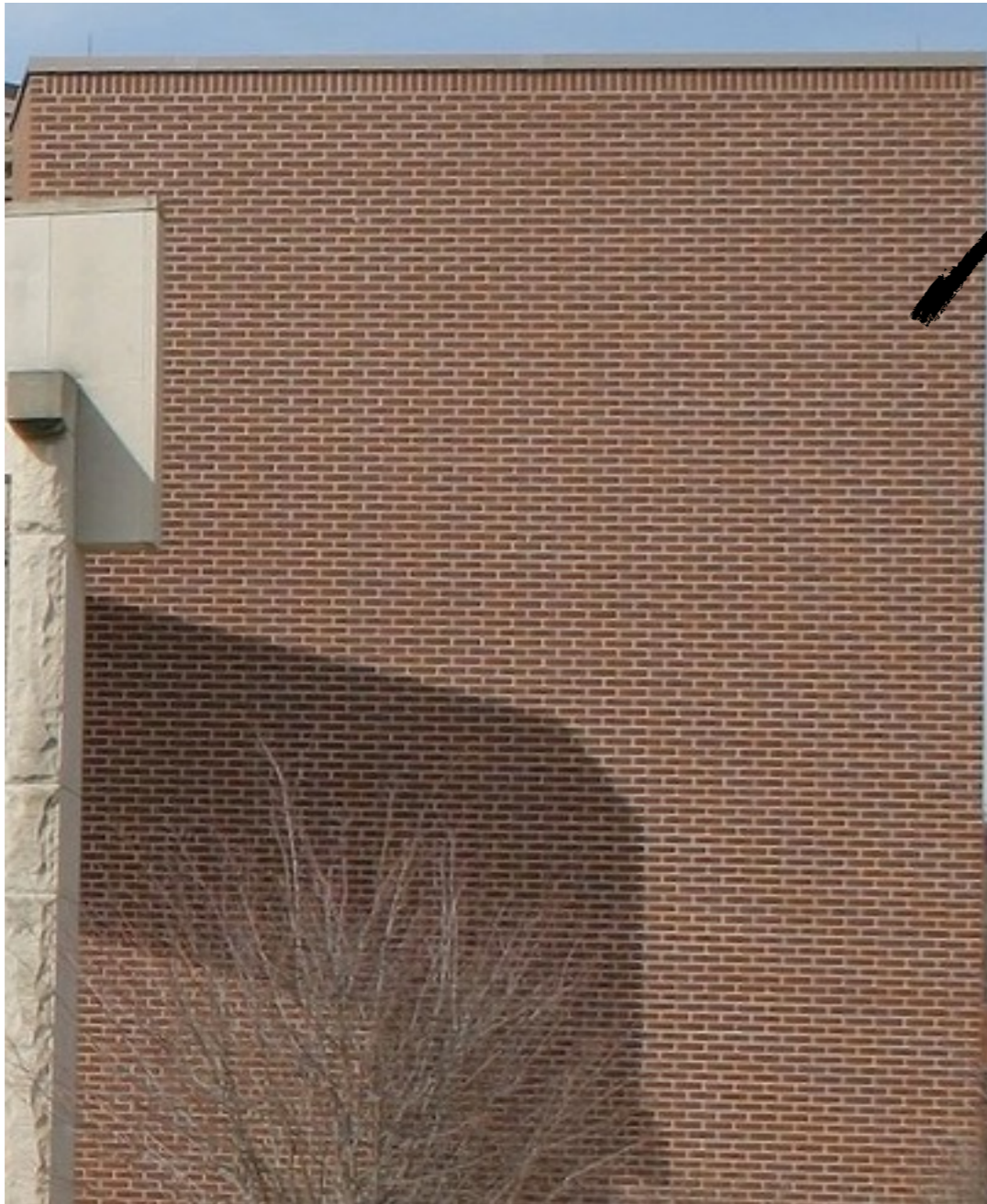
Thumbnails

▶ Normally,

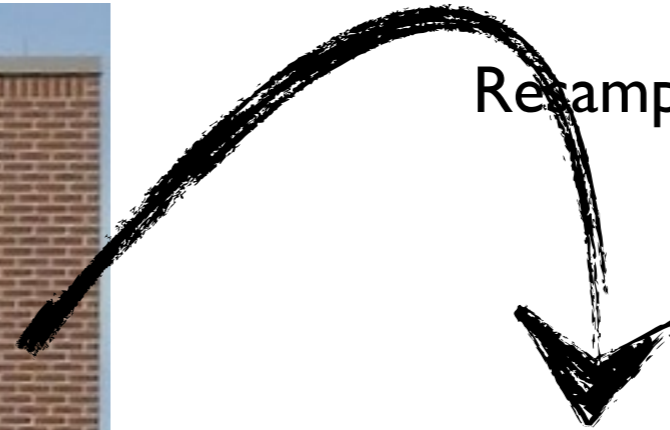
- **Pre-Filtering** is a low-pass filter to avoid *aliasing effects* before compression (e.g., distortions, Moiré patterns etc.)
- **Post-Filtering** is a sharpening filter

Thumbnails

© Wikimedia Commons



Resampling with *Moiré Artifacts*



© Wikimedia Commons

Thumbnails



Sharpening Filter

© A. Rocha

Thumbnails

- ▶ In the last step, we perform JPEG compression of the created *thumbnail*
- ▶ In practice, we do some assumptions to simplify the model

Thumbnails

► First, we assume that:

- A circular and symmetric Gaussian pre-filter
 $\exp(-(x^2 + y^2)/\sigma^2)$
- Pre-filter is unitary and sums 1
- The Post-Filter has a 3x3 kernel

Thumbnails

► We assume that:

- Post-Filter is also symmetric

$$h_2(x, y) = h_2(-x, y) \text{ e } h_2(x, y) = h_2(x, -y)$$

which results in the filter $(aba; bcb; aba)$

- Post-Filter is unitary with sum 1 subject to

$$c = 1 - (4a + 4b)$$

Thumbnails

- ▶ With these restrictions, a *thumbnail* is specified by
 - 11 processing parameters
 - 2 for the *thumbnail's size*
 - 4 for *cropping/padding*
 - 1 for *pre-filtering*
 - 2 for *post-filtering*
 - 2 for *brightness/contrast*

Thumbnails

- ▶ In addition, we use **128 parameters** for the JPEG compression
 - 64 (8x8) for luminance
 - 64 (8x8) for chrominance – the authors assume the two chrominance channels have the same JPEG quantization table

**Ok... but how to
calculate the
parameters**

Calculating the thumbnail

- ▶ The first step in the *thumbnail construction*, consists of generating a delimiting rectangle wrt. the high-res image.
- ▶ This is done using translations and scale operations (**anysotropically**) over the delimiting rectangle such that the sampled image is compatible with the *thumbnail* extracted from the high-res image

Calculating the thumbnail

- ▶ For calculating the cropping parameters, first we extract the image's stored thumbnail. Let's call it t .
- ▶ Then we create a delimiting rectangle with the same aspect ratio as t .
- ▶ We scale such rectangle to comprise the whole image in its largest dimension

Calculating the thumbnail

- ▶ The high-res image is then cropped according to this rectangle, *padded* and *downsampled* to create the initial *thumbnail*

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

- ▶ The sampling ration (D) is given by: $\max\left(\frac{N_x}{n_x}, \frac{N_y}{n_y}\right)$
- ▶ N and n refer to image dimensions and thumbnail dimensions, respectively

Calculating the thumbnail

- ▶ The initial *thumbnail* undergoes anisotropic scaling and translation to *match* the real *thumbnail* extracted from the high-res image

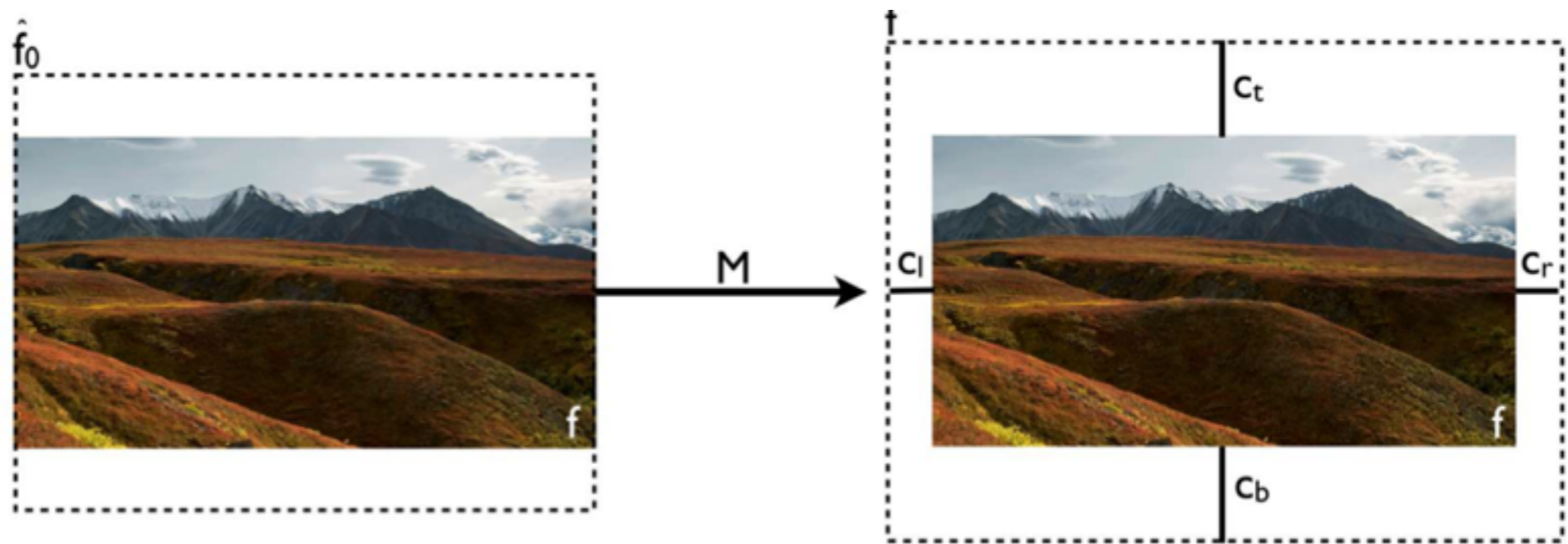
$$t(x, y) = \hat{t}_0(s_x x + \Delta_x, s_y y + \Delta_y)$$

Scaling Translation
↑ ↑

- ▶ For *matching*, we use **image registration techniques**

Calculating the thumbnail

- ▶ After registering, we find the scaling and translation parameters to be applied to the image in high-res



© Kee & Farid

Calculating the thumbnail

- ▶ After estimating the cropping and scaling parameters, we need to estimate the pre and post filter parameters along with the brightness and contrast parameters

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

- ▶ This function is defined in terms of the filters and not of the brightness and contrast parameters

Calculating the thumbnail

- ▶ Therefore, given a pair of filters, we estimate the brightness and contrast upon minimizing the following function

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

where

$$\hat{t}_h(x, y) = D\{\hat{f}(x, y) \star h_1(x, y)\} \star h_2(x, y)$$

Calculating the thumbnail

- ▶ For minimizing E_1 , we use **brute force in the parameter space (grid-search)**
- ▶ In each step of E_1 , **we minimize E_2 analytically**

Calculating the thumbnail

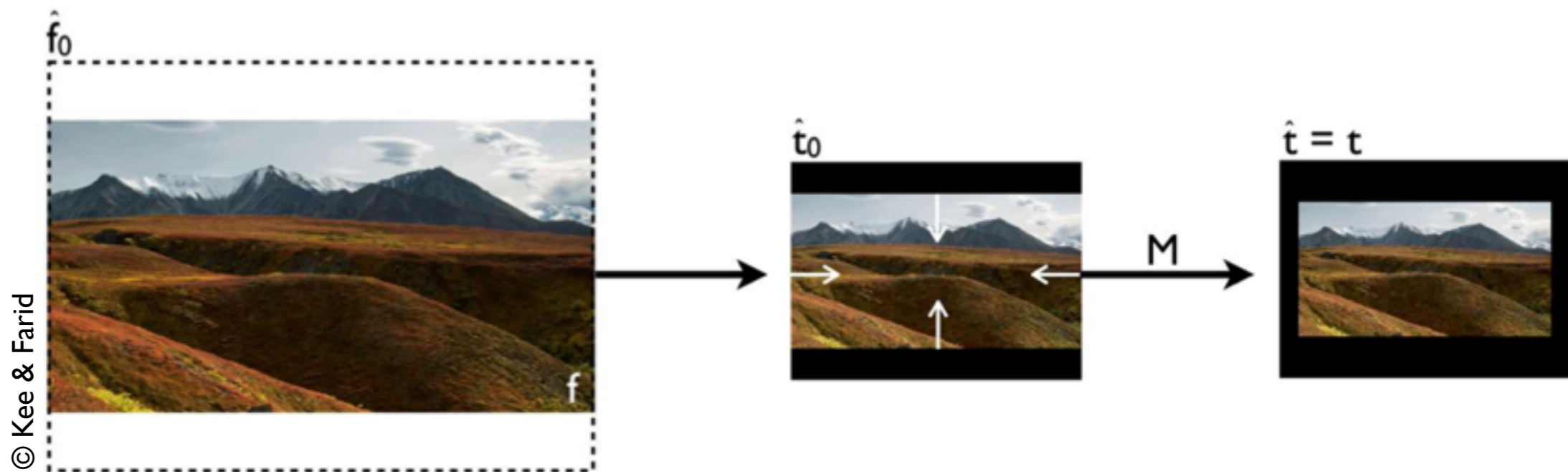
- ▶ Finally, the dimensions of the “real” *thumbnail* and the 128 compression parameters are extracted from the JPEG container
- ▶ Given that the original (stored) *thumbnail t* was compressed with such parameters, we need also to compress our estimations

Calculating the thumbnail

► Specifically,

- $\hat{t}_0(\cdot)$ is compressed before estimating the scale and translation parameters
- $\hat{t}_h(\cdot)$ is compressed before estimating the brightness and contrast parameters

Calculating the thumbnail



Experiments & Validation

Experiments & Validation

- ▶ 1,514 non modified Flickr images
- ▶ **Non-modified** means
 - Classified as original by Flickr
 - Creation/modification dates are consistent
 - Metadata are not modified/removed
 - Metadata “*software*” is empty

Experiments & Validation

- ▶ 1.514 images span **142 câmeras** (brand and models)
- ▶ Cameras of the same model and maker, sometimes, vary the thumbnail size as well as *their* quantization tables (e.g., different *firmware*)

Experiments & Validation

- ▶ These variations affect the parameter calculations
- ▶ In this way, the 1,514 images are grouped into 245 camera classes (called equivalence class henceforward)
 - Same maker/model
 - Same features for the quantization tables and thumbnail sizes

Experiments & Validation

- ▶ First, we calculate the parameters for a representative in each class
- ▶ Pre- and Post- filters as well as brightness and contrast are calculated by brute force
- ▶ **Pre-filter**: $[0.005, 1]$, 20 equally spaced steps
- ▶ **Post-filter**: $[-0,5, 0,5]$, 11 equally space steps
- ▶ Brightness and contrast are analytically calculated

Experiments & Validation

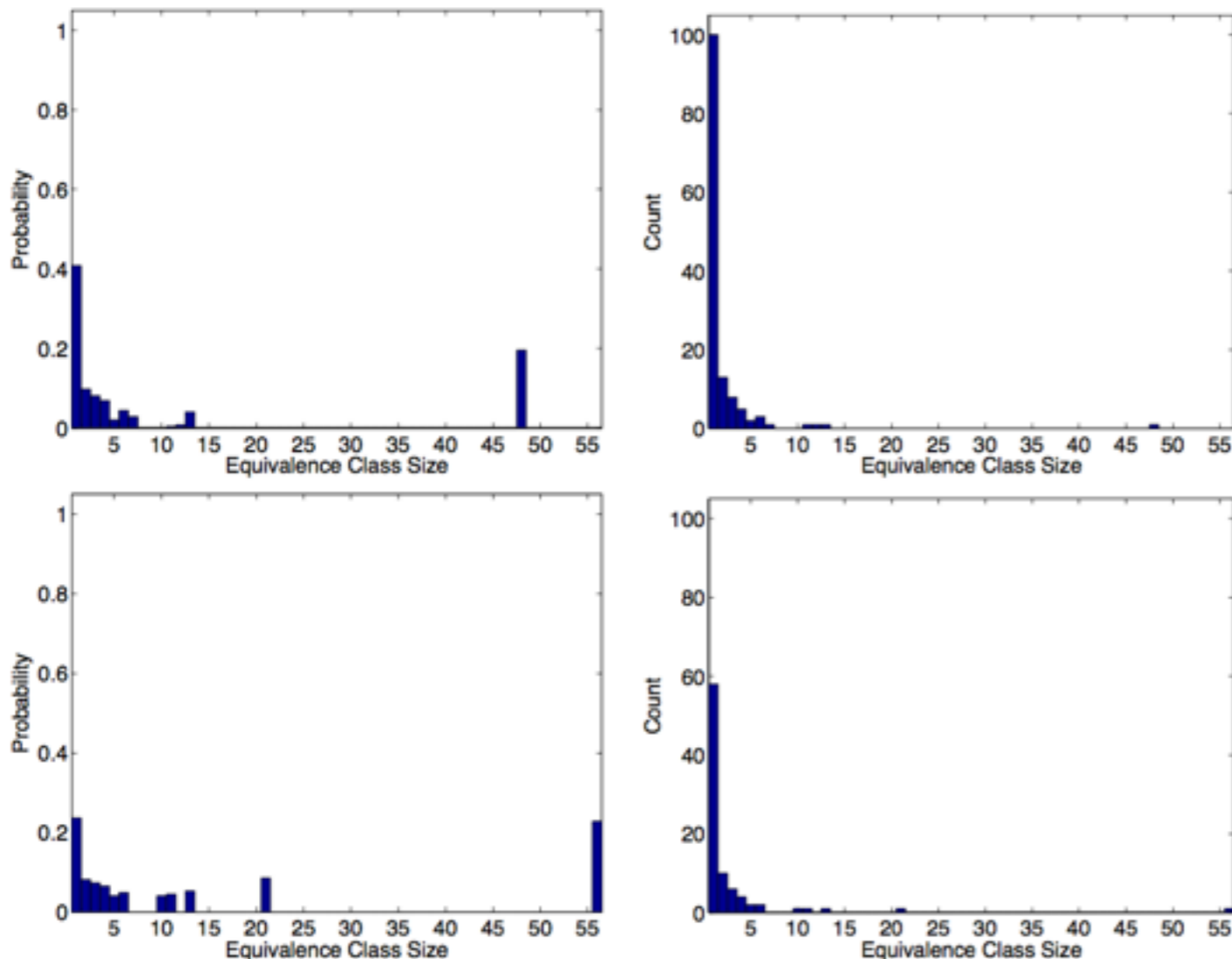
- ▶ Thereafter, it is evaluated the consistency of these parameters for identifying/differentiating images of a given class of among classes of equivalence
- ▶ The *thumbnail parameters* for a given camera class are given by the average of the cropping limits, pre- and post-filtering values as well as the average of the brightness/contrast of all images in a given class

Experiments & Validation

- ▶ The *thumbnail size* and the quantization table are the same for all of the images in a given class by construction
- ▶ Two *thumbnail models* are considered equivalent if their parameters are similar up to an accepted level of difference (threshold)

Experiments & Validation

© Kee & Farid



Left: Probability of a camera be in a given class of equivalence of a determined size

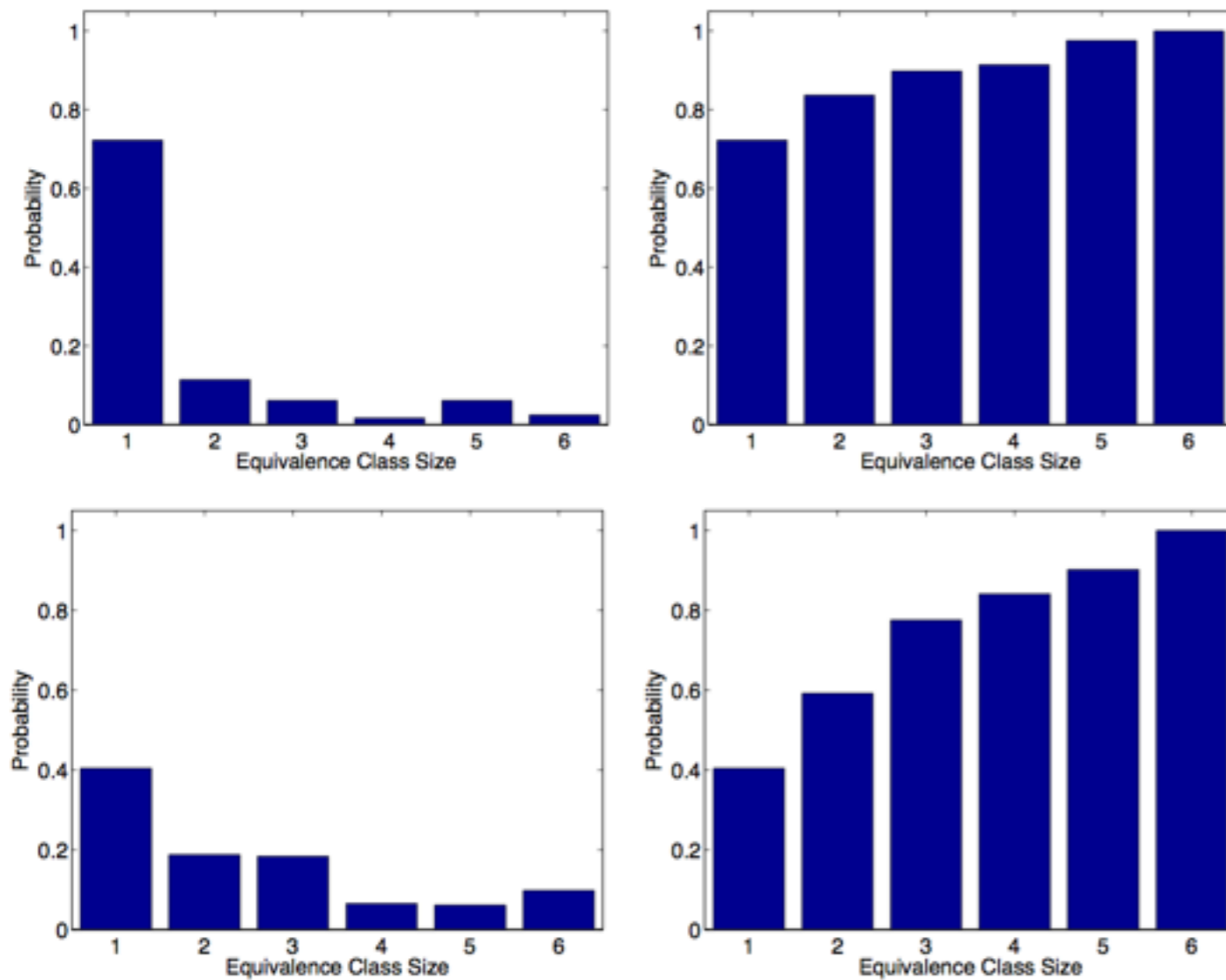
Right: number of classes of equivalence for a given thumbnail size

Row #1: distributions considering all parameters of the *thumbnail* (40.8% of the cameras have unique parameters – they have class of equivalence #1)

Row #2: distributions considering only the thumbnail size and the quantization table (23.7% of the cameras have unique parameters).

Experiments & Validation

© Kee & Farid



Left: Probability of a camera be in a class of equivalence of a determined size.

Right: CDF

Row #1: distributions considering all estimated parameters for the *thumbnail plus* the image resolution (**72.2% of the cameras have unique parameters** – tem have class of equivalence #1)

Row #2: distributions considering only the image size and the quantization table (40.4% of the cameras have unique parameters).

Experiments & Validation

- ▶ For pushing the technique even more, the authors have analyzed also the thumbnails normally generated by software such as **Adobe Photoshop (e.g., CS3)**
- ▶ Some images are saved with 13 different compression parameters
- ▶ The parameters are estimated and compared to the 245 classes of equivalence

Experiments & Validation

- ▶ **None of the thumbnail parameters** generated in Photoshop are shared by any of the 245 classes of equivalence
- ▶ The same happens for the image quantization tables and the thumbnail quantization tables
- ▶ **This is a strong discriminative signature**

Final Remarks

Considerações Finais

- ▶ Parameters relative to the *thumbnails generation process* may be effectively used for **Image Authentication**
- ▶ Better results are obtained when combining compression and input image's dimensions

Considerações Finais

- ▶ The approach's effectiveness relies upon
 - a good thumbnail database and their parameters (classes of equivalence)
 - a good image collection
- ▶ Authors are now reporting results with about 1.2M original images

References

References

1. [Avcibas et al. 2006] Avcibas, I., Bayram, Sankur, B., S., Memon, N., (2006). *Image manipulation detection*. In Journal of Electronic Imaging (JEI), 15(4):1–17.
2. [Kee & Farid 2010] Kee, E., Farid, H., (2010). *Digital Image Authentication from Thumbnails*. In SPIE Symposium on Electronic Imaging (SEI), pp. 1–10, San Jose, USA.

View from the window at Le Gras – Joseph Niepce (GoogleGram)

Obrigado!
