

# Blind Detection of Image Rotation and Angle Estimation

Miroslav Goljan, Department of ECE, SUNY Binghamton, NY, USA, [mgoljan@binghamton.edu](mailto:mgoljan@binghamton.edu)

## Abstract

*The reliability of many digital forensic techniques can be negatively affected by geometrical transforms applied to the image under investigation because it breaks pixel-to-pixel synchronization needed for example for forensic methods that rely on sensor fingerprints. The geometrical transform typically needs to be detected and then its parameters estimated to allow subsequent successful and reliable forensic analysis. This paper focuses on blind detection of rotation and estimation of the angle by which the image was rotated. The proposed method utilizes the so-called Linear Pattern (LP) as a global template. In particular, no side information, such as watermark or the EXIF header, is required. The method is generally applicable whenever the image under investigation had a sufficiently strong LP before rotation. The performance of the method is assessed experimentally and by comparing to previous art. The main advantage of the proposed method is its accuracy for estimating small rotation angles (less than 3 degrees). It will also work after resizing.*

## Introduction

Digital cameras have proliferated into mobile phones and tablets in enormous numbers. These devices are capable of fast image processing. Easy to use software and apps allow users to further manipulate the taken photos, improve the layout by cropping, resize to smaller dimensions to save bandwidth when sharing them and even do much more. If the device does not have such software installed online tools are also available for free, such as at [fotor.com](http://fotor.com). One of typical mistakes a photographer makes and wants to correct for is not having a straight horizon. Rotating the image, often by a small angle, fixes the problem. Another scenario where rotation of images occurs is during electronic image and video stabilization.

Images that have undergone geometric transforms, in general, pose a challenge to camera identification based on photo-response non-uniformity (PRNU), the camera fingerprint. Matching these fingerprints is done by registration and correlation. Successful matching thus relies on pixel-to-pixel synchronization. In case of rotated images, a straightforward approach is the generalized likelihood ratio test that maximizing the correlation coefficient computed between the fingerprint estimated from the rotated image and the reference fingerprint while sampling a range of rotation angles [9]. This approach has several disadvantages. First, this time consuming process needs to be repeated for each matching test with the same image and different reference fingerprint. Second, with a denser sam-

pling of the rotation angle the detection threshold has to be correspondingly increased to keep the false alarm rate low [4]. With two or more geometric transforms applied to the same image, the registration problem with noisy fingerprints quickly becomes computationally intense. This is when methods capable of blind estimation of the image rotation angle can reduce the computational complexity tremendously. By “blind” we mean that the image before rotation is not accessible and the only data assumed to be available is the image pixels.

Previous methods estimate the rotation angle from interpolation artifacts detected in the Discrete Fourier Transform (DFT) domain. Earlier work of this kind by Wei *et al.* [11] analyzes the location of peaks in the spectrum of the image edge map and determines the angle via closed-form formulas. Vázquez-Padín *et al.* [10] applied cyclostationarity theory for estimating the scaling factor and the rotation angle. In 2014, Chen *et al.* built upon the work of Vázquez-Padín *et al.* and proposed a more sophisticated spectrum-based method (SBM) [1]. All these methods, however, work poorly for estimating small angles, including the case when the image is not rotated. A drawback of the SBM is its computational complexity as it requires to compute a pseudoinverse of an  $18 \times 18$  matrix as many times as there are pixels in a large square block cropped from the image.

In this work, we propose to utilize a different forensic entity called the linear pattern instead of interpolation artifacts. The LP was first recognized in the digital forensic community in [2] where it was identified as a nuisance undesirable signal (a part of Non-Unique Artifact or NUA) that had to be removed or suppressed to lower the false-alarms in camera identification using sensor fingerprints. The LP was shown to be a useful forensic entity by itself in [3] for identifying the camera model or brand. It was also used as a template for blind estimation of lens distortion correction [5]. In this paper, the LP is used to estimate the rotation angle of a rotated image.

The paper is organized as follows. The next section introduces notation and contains a formal definition of the linear pattern. The following section explains advantages of the LP for image blind rotation estimation and outlines the proposed method, which is then described in detail in the following section. The experimental section presents tests of the proposed LP-based method (LPM) on 48 uncompressed images, comparing the results with SBM. The tests continue with JPEG compressed rotated images and downsized rotated images. The paper is concluded in the final section.

## Preliminaries

Matrices will be denoted with bold capital letters, vectors with lower-case bold letters, and scalars in lower-case italic font. An 8-bit grayscale  $m \times n$  image ( $m$  rows,  $n$  columns) will be represented with a two-dimensional array  $\mathbf{I} \in \{0, \dots, 255\}^{mn}$  with elements  $I_{ij}$ ,  $1 \leq i \leq m$ ,  $1 \leq j \leq n$ . Its noise residual is defined as  $\mathbf{W}(\mathbf{I}) = \mathbf{I} - F(\mathbf{I})$ , where  $F$  is a denoising filter. Then,  $\mathbf{W}$  is normalized to zero mean and scaled to unit sample variance with samples  $w_{ij}$ ,  $1 \leq i \leq m, 1 \leq j \leq n$ .

Linear Pattern derived from  $\mathbf{W}$  is matrix  $\mathbf{L}$  of size  $m \times n$ , such that the following two conditions are satisfied:

1. The average of each column and row of  $\mathbf{Z} = \mathbf{W} - \mathbf{L}$  is zero.
2.  $\mathbf{L} = \mathbf{L}(\mathbf{W}) = (l_{ij})$  is fully determined by its row and column averages, i. e.,  $l_{ij} = r_i + c_j$  for all  $i = 1, \dots, m$ ,  $j = 1, \dots, n$ .

The first condition implies that row  $\mathbf{r} = (r_1, r_2, \dots, r_m)$  and column  $\mathbf{c} = (c_1, c_2, \dots, c_n)$  averages of  $\mathbf{L}$  are the same as of  $\mathbf{W}$ ,

$$\begin{aligned} r_i &= \frac{1}{n} \sum_{j=1}^n l_{ij} = \frac{1}{n} \sum_{j=1}^n w_{ij}, \\ c_j &= \frac{1}{m} \sum_{i=1}^m l_{ij} = \frac{1}{m} \sum_{i=1}^m w_{ij}. \end{aligned} \quad (1)$$

The definition of  $\mathbf{L}$  is self-consistent because  $\sum_{i,j=1}^{m,n} w_{ij} = 0$  and a the  $j$ th column average of  $\mathbf{L}$  is

$$\begin{aligned} \frac{1}{m} \sum_{i=1}^m l_{ij} &= \frac{1}{m} \sum_{i=1}^m (r_i + c_j) = \frac{1}{m} \sum_{i=1}^m r_i + \frac{1}{m} (mc_j) \\ &= \frac{1}{m} \sum_{i=1}^m \frac{1}{n} \sum_{j=1}^n w_{ij} + c_j = c_j, \end{aligned} \quad (2)$$

By the same token, the average of the  $i$ th row is equal to  $r_i$ .

The pair of vectors  $[\mathbf{r}, \mathbf{c}]$  is a one-dimensional representation of the LP. We define the energy of the LP as the pair  $[\|\mathbf{r}\|^2, \|\mathbf{c}\|^2]$ , where  $\|\cdot\|$  stands for the Euclidean norm. We also define the *normalized LP energy*  $e(\mathbf{L})$ ,

$$e(\mathbf{L}) = [\mathbf{e}_r, \mathbf{e}_c] = [n/m \|\mathbf{r}\|^2, m/n \|\mathbf{c}\|^2]. \quad (3)$$

The normalization coefficients  $n/m$  and  $m/n$  are justified in the following way. If  $\mathbf{W}$  was random i.i.d. noise then the expected normalized energy of the LP becomes equal to 1. For the rows part, the expected normalized energy becomes

$$\begin{aligned} E\left\{\frac{n}{m} \|\mathbf{r}\|^2\right\} &= E\left\{\frac{n}{m} \sum_{i=1}^m \left(\frac{1}{n} \sum_{j=1}^n w_{ij}\right)^2\right\} \\ &= E\left\{\frac{n}{m} \sum_{i=1}^m \frac{1}{n^2} \left(\sum_{j=1}^n w_{ij}^2 + 2 \sum_{i \neq k} w_{ij} w_{ik}\right)\right\} \\ &= \frac{1}{mn} E\left\{\sum_{i=1}^m \left(\sum_{j=1}^n w_{ij}^2\right)\right\} \\ &\quad + \frac{2}{mn} E\left\{\sum_{i=1}^m \left(\sum_{i \neq k} w_{ij} w_{ik}\right)\right\} \\ &= 1 + 0. \end{aligned} \quad (4)$$

The 1 is due to the zero mean and unit variance property and the zero due to the independence between  $w_{ij}$  and  $w_{ik}$  for  $j \neq k$ . The same can be verified for the columns part of the LP. For most images from digital cameras and further processed images, at least one of the normalized energies of the LP is larger than 1,  $\mathbf{e}_r > 1$  or  $\mathbf{e}_c > 1$  or both. The reason for it is that the neighboring elements in the noise residual  $\mathbf{W}$  are not independent, for example due to color filter array interpolation.

In this work, the LP from color images is obtained from  $\mathbf{W}$  computed by converting the three residuals  $\mathbf{W}_R, \mathbf{W}_G, \mathbf{W}_B$ , from the red, green, and blue color channels, to one "grayscale"  $m \times n$  matrix  $\mathbf{W}$  using standard  $\text{rgb} \rightarrow \text{gray}$  linear transform with coefficients  $[0.2125, 0.7174, 0.0721]$  for red, green, and blue color channels. In this paper, we use the popular denoising filter described in [7]. The noise parameter  $\sigma$  was optimized in a small test and rounded to  $\sigma = 0.5$ , which is much smaller than what is typically used for estimating camera sensor fingerprint [6]. A small value of  $\sigma$  works better for JPEG compressed images.

## Linear pattern as a reference template

The linear pattern is introduced into the image during its on-board camera signal and image processing. Filler *et al.* [3] showed that features extracted from the LP can help identify the camera model of the camera that took the image. This observation suggests that the LP as defined in Eq. 1 is an estimate of a LP that is introduced into every image (of the same dimensions) the camera takes. Unlike the PRNU noise, the LP is not unique for each camera.

An important property of the LP is its directionality, at least within a 90 degree range with the possibility to be expanded to a 180 degree range depending on whether the horizontal direction can be distinguished from the vertical direction. In this paper, we exploit this property for rotation estimation. Another useful property is its robustness

to scaling. This opens up a possibility for fast camera identification from rotated, scaled, and cropped images using the PRNU and the LP. Instead of doing a multidimensional search for all parameters of the geometric transform at once, it will be possible to first recover the rotation using the proposed LP-based method (LPM) and then find the scaling factor that maximizes the cross-correlation [4]. Such experiments are left for future work.

Every image processing has some effect on the LP. What is important for our blind estimation of rotation angle is the observation that after image rotation by a small angle, the normalized energy of the LP approaches 1 or a smaller value (see Figure 1). This is the result of random-like values in the noise residual  $\mathbf{W}$  in other than vertical and horizontal directions. The original linear pattern was rotated along with the image. The LP energy computed from an image that was already rotated tends to be lower than before the rotation. We can see from Figure 1 that rotation quickly eliminates most of the LP energy. After rotating it back to the original position, the LP energy increases. Once we make the same plot computed from an image that was already rotated by angle  $\alpha$  (and cropped to remove padding), the peak energy clearly marks the angle of rotation with the opposite sign (see Figure 2 and Figure 3 in case of JPEG compressed rotated image). This proves that the LP can survive two rotations (forth and back) of the image, both with interpolation and rounding pixels to 8-bit values. The above two cases shown in the figures represent a) an uncompressed digital image with a very strong LP, and b) a heavily compressed image with much weaker LP.

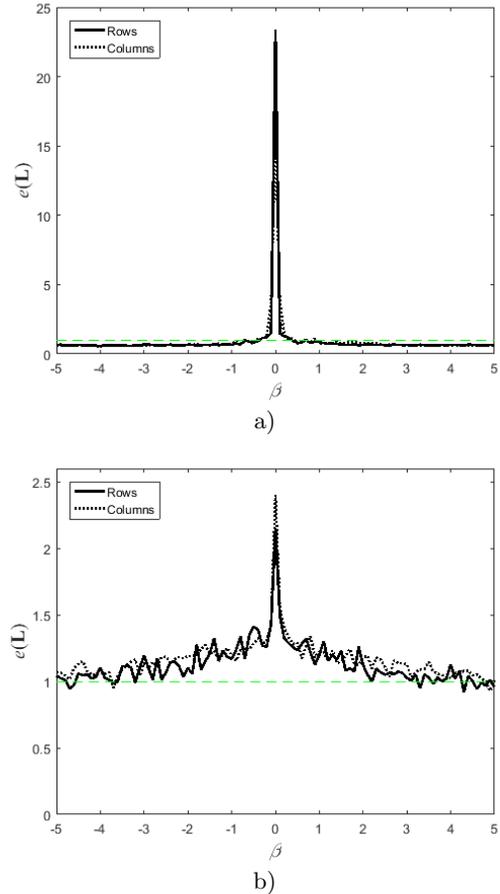
The initial to-be-detected rotation in our example was carried out with bicubic interpolation. Figure 4 shows the same experiment as in Figure 2 (b), using three different interpolations for rotations by angle  $\beta \in [-5, 5]$ . Bilinear and bicubic methods decrease the LP energy, which creates a false peak in the energy plot at  $\beta = 0$ . The same effect occurs no matter what the initial rotation method was (not shown here). For this reason and for its computational efficiency, it is important to implement the proposed search method using the nearest neighbor interpolation.

Depending on the image origin, its LP may be strong, i. e., having energy larger than 1, in either the row or the column direction or in both. Among many options how to merge these two parts of the LP when searching for the rotation angle, we choose the simple addition that defines the objective function  $G(\beta) = \mathbf{e}_r(\beta) + \mathbf{e}_c(\beta)$ , computed from the image after rotating it by  $\beta$  degrees. We next describe the step-by-step algorithm for blind rotation angle estimation.

## Description of the method

The method for estimation of the rotation angle from a (possibly) rotated image consists of six steps:

**Step 1:** Compute the noise residual  $\mathbf{W}$  from the inspected image  $\mathbf{I}$ ,  $\mathbf{W} = \mathbf{I} - F(\mathbf{I})$ , where  $F$  is the denoising filter [7]. Normalize  $\mathbf{W}$  to zero mean and unit sample variance.



**Figure 1.** Example of a linear pattern energy computed for a range of rotation angles for a) uncompressed image of size  $1704 \times 2272$  from Canon\_S40 b)  $1080 \times 1920$  HD video frame recorded by Kindle Fire tablet.

**Step 2:** For the search interval  $b = [\beta_1, \beta_2]$  of rotation angles, and for a chosen step  $s$ , take samples  $\alpha_i$ ,  $i = 1, \dots, n$  separated by  $s$  from interval  $b$ .

**Step 3:** Rotate  $\mathbf{W}$  by angle  $\alpha_i$  using the nearest neighbor interpolation to obtain  $\mathbf{W}_i$ . Crop to remove padding.

**Step 4:** Compute  $\mathbf{L}(\mathbf{W}_i)$ , and  $e(\mathbf{L}(\mathbf{W}_i)) = [\mathbf{e}_r(\alpha_i), \mathbf{e}_c(\alpha_i)]$  using Eq. 1 and 3.

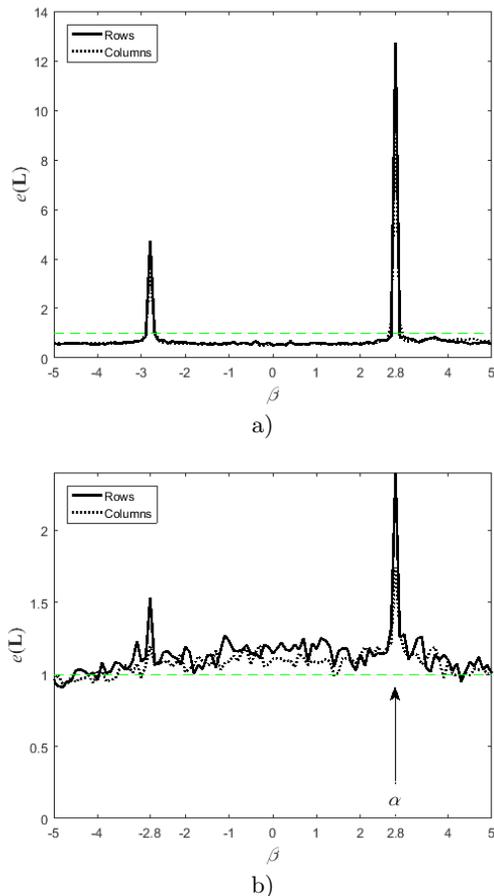
Repeat Steps 3 and 4 for  $i = 1, \dots, n$ .

**Step 5:** Compute the function  $G(\beta) = \mathbf{e}_r(\beta) + \mathbf{e}_c(\beta)$  sampled at  $\beta = \alpha_i$ ,  $i = 1, \dots, n$ .

**Step 6:** Fit a second-order polynomial  $P(\beta)$  through the  $G(\beta)$  data. Find the estimate  $-\hat{\alpha}$  of the image rotation angle  $\alpha$  as the largest positive outlier

$$\hat{\alpha} = \arg \max_{\beta_1 \leq \beta \leq \beta_2} G(\beta) - P(\beta).$$

Notice that we do not rotate the image itself during the search. It is much faster to compute the noise residual just once and rotate only that. We found no loss in the performance of the method compared to rotating the image



**Figure 2.** Linear pattern energy computed for a range of rotation angles for the image previously rotated by  $\alpha = -2.8$  degrees for a) raw image b) HD video frame.

in Step 3. The data regression as part of the peak detection in Step 6 is needed mainly when  $\beta_2 > 5$ . The objective function tends to increase if the image size after cropping decreases.

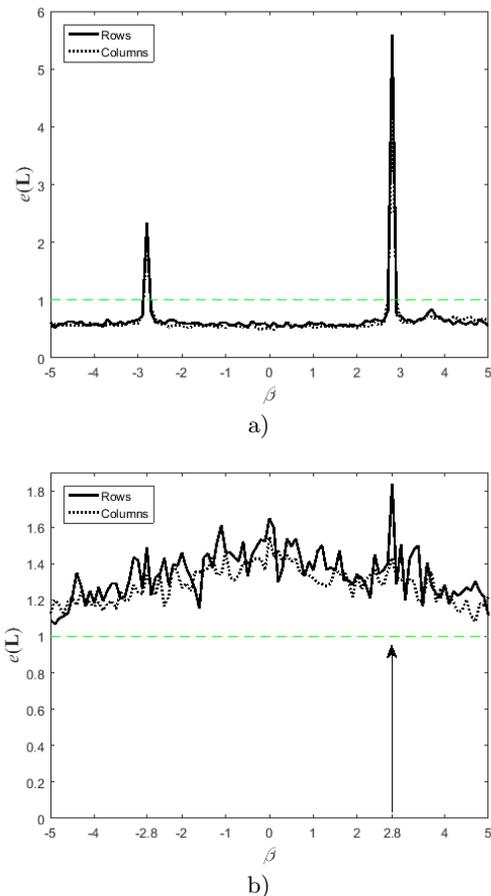
A typical width of the peak is about a quarter degree for images with resolution a few million pixels, suggesting that the maximum size of the sampling step should be much smaller than 0.25 degree. Note that the peak width is proportional to the image width (or height).

### Special cases

Two problems require attention before discussing routine tests. The first is the phenomenon we call the *mirror peak*. The second is a scenario involving double rotation that can be taken as an experimental proof that the LPM, unlike SBM, does not use post-rotation interpolation artifacts.

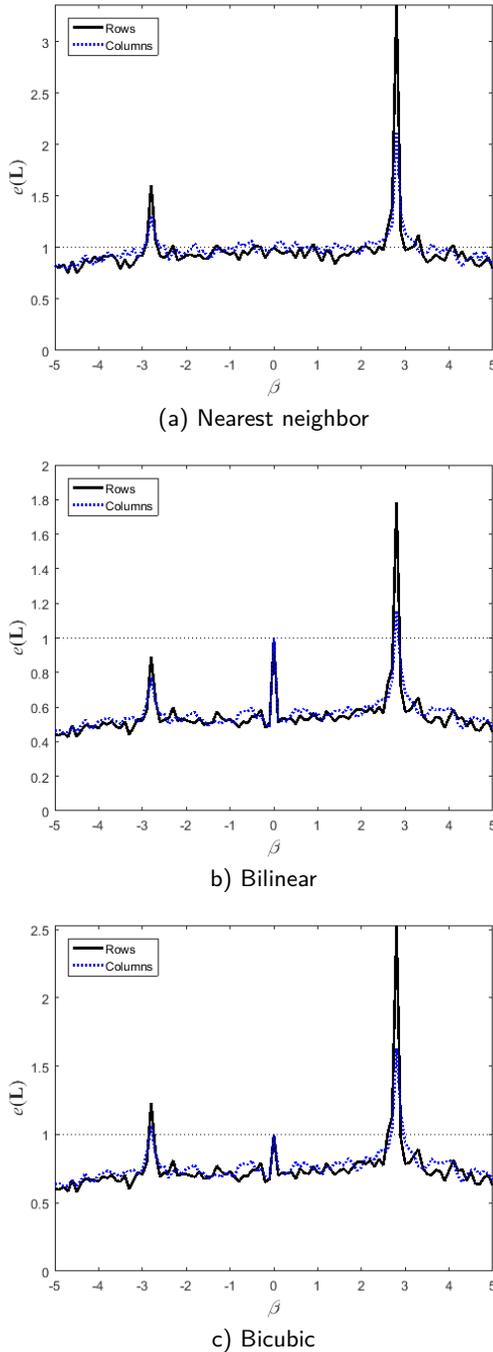
#### Mirror peak problem

Upon closer inspection of Figure 2, one can observe that the plot exhibits an (unexpected) secondary mirror peak. A more detailed investigation reveals that the LP



**Figure 3.** Linear pattern energy computed for a range of rotation angles for an image rotated by -2.8 degrees and JPEG compressed at 95% quality a) raw image b) HD video frame, for a range of rotation angles.

energy suddenly increases after rotating the rotated image by the same angle again. The magnitude of this secondary peak increases with a smaller rotation angle and almost matches in magnitude the main peak as  $\alpha \rightarrow 0$  (not shown in the figure). Thus, with small rotation angles,  $\alpha$  can be confused for  $-\alpha$  if the sampling of  $G(\beta)$  is not sufficiently dense (i. e., the step  $s$  is not sufficiently small) during the exhaustive search. The culprit is a strong periodic LP with a two-pixel period. We illustrate the process of the mirror peak formation on an example with the column part of the LP composed of a perfect periodic signal with period 2 shown in Figure 5(a), top. Black represents -1 and white +1. The plot in the same Figure 5(a) below is of vector  $\mathbf{c}$ , the mean values of the LP columns. A small rotation by just 2 degrees (part (b) in the figure) makes the new calculated LP having a very small amplitude, which is desired. But when rotating by an additional 2 degrees (part (c) in the figure), due to the period 2, the amplitude increases almost to the original value. If the same LP is rotated once by  $2 + 2 = 4$  degrees, the aliasing does not occur and the amplitude stays small. The effect becomes less pronounced if rotating by a larger angle.



**Figure 4.** Normalized LP energy  $e(L)$  of a rotated video frame at angles  $\beta$  using a) nearest neighbor, b) bilinear, c) bicubic interpolation. The the image is a video frame of dimensions  $1080 \times 1920$ . The initial rotation by 2.8 degrees was executed with bicubic interpolation.

Test #	Method	$\alpha^{(1)}$	$\alpha^{(2)} = 2.6$ degrees
1	True angle	13.333	$\alpha^{(1)} + \alpha^{(2)} = 15.933$
	SBM	13.330	2.577
	LPM	13.330	15.950
2	True angle	-1.123	$\alpha^{(1)} + \alpha^{(2)} = 1.477$
	SBM	-1.119	2.577
	LPM	-1.140	1.430

**Table 1.** Double rotation test with an uncompressed image from Canon\_S40 (resolution  $1704 \times 2272$ ).

The mirror peak phenomenon confuses the SBM at angles less than 2 degrees (see the columns with small angles in Table 4). The LPM is resistant to it as long as the sampling step  $s$  is sufficiently small ( $s = 0.025$  in our tests).

It is worth noting that there is an ambiguity once the range of possible rotations exceeds 90 degrees. If the LPM finds the estimate  $\hat{\alpha}$  then other plausible rotations are  $\hat{\alpha} + 90$ ,  $\hat{\alpha} - 90$ , and  $\hat{\alpha} + 180$ . Rotations by a multiple of 90 degrees cannot be distinguished unless a side information about the camera is available.

### Double rotation

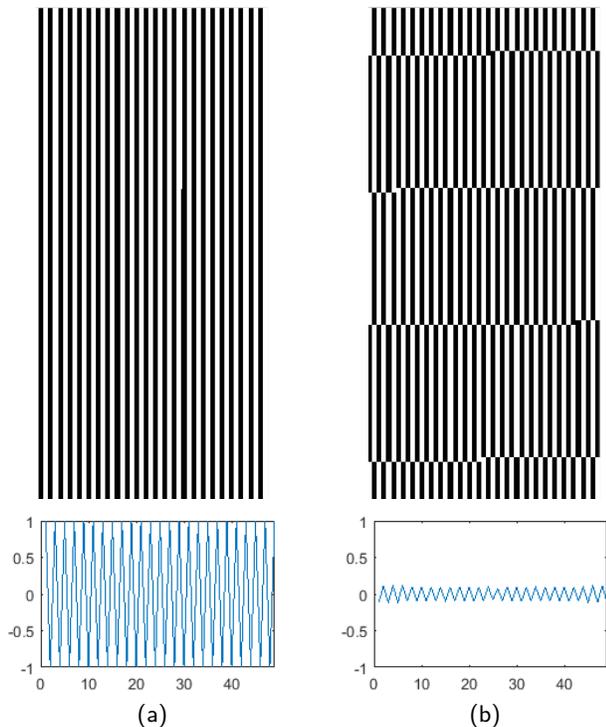
An interesting scenario, although not very common in practice, involves images rotated twice, first by angle  $\alpha^{(1)}$  and subsequently by  $\alpha^{(2)}$ . Experimenting with an uncompressed image rotated by  $\alpha^{(1)}$ , both methods correctly detect it. After the second rotation, the SBM, however, only finds this second angle while the LPM determines the total rotation. The two methods thus represent two different tools. If camera identification is the primary purpose for undoing the rotation then the LPM is the right choice.

## Experiments

Having reproducibility of experiments in mind, we choose the FAU dataset of 48 original test images provided by Friedrich-Alexander-Universität, Erlangen-Nürnberg, Germany.<sup>1</sup> These images are in the 8-bit PNG format. The origin of the images is unknown to us. Their resolution spans from small  $533 \times 800$  to the largest  $3900 \times 2613$ . We list the image dimensions along with normalized energy of their LP for the first 20 test images in Table 2.

Camera identification using PRNU works best when pixel-to-pixel synchronization is achieved. Estimating the rotation to stay within one pixel difference at a distance of 1200 pixels from the image center requires better than 0.05 degree precision for the angle estimate. We take this value as one of two thresholds (the second is 0.1 degrees) for evaluating the success rate of our estimation. For improved accuracy, two stages of the search were implemented. The search interval in all experiments with small angles below 5 degrees is set to  $b = [-5, 5]$ ,  $s = 0.025$ , while for larger angles  $b = [0, 45]$ ,  $s = 0.05$ . The step is then set to  $s = 0.01$  during the second, refining stage, searching within 1 degree

<sup>1</sup><https://www5.cs.fau.de/research/data/image-manipulation/>



**Figure 5.** Origin of the “mirror peak” in the plot in Figure 2. (a) Original (simulated) LP, (b) LP rotated by 2 degrees, (c) LP rotated by 2 degrees twice, (d) LP rotated by 4 degrees. (All rotations are done with the nearest neighbor interpolation.)

Image filename	Rows	Columns	$e_r$	$e_c$
'barrier'	2592	3888	3.99	12.81
'beach_wood'	2448	3264	7.71	6.50
'berries'	2014	3039	4.78	4.10
'bricks'	2592	3888	4.53	6.72
'cattle'	854	1280	1.28	1.31
'central_park'	2448	3264	0.70	2.15
'christmas_hedge'	2014	3039	18.63	14.85
'clean_walls'	754	1024	1.16	1.05
'dark_and_bright'	2592	3888	7.15	8.09
'disconnected_shift'	3888	2592	6.79	7.51
'egyptian'	2014	3039	1.74	1.42
'extension'	2592	3888	2.14	6.10
'fisherman'	2448	3264	0.54	0.96
'fountain'	2448	3264	6.52	5.66
'four_babies'	2000	3008	1.25	0.63
'giraffe'	533	800	2.07	2.83
'hedge'	2448	3264	13.89	13.68
'horses'	2014	3039	9.40	5.96
'japan_tower'	2448	3264	6.87	5.29
'jellyfish_chaos'	2592	3888	4.54	17.49
...	...	...	...	...

**Table 2.** Dimensions and LP energy of test images from FAU dataset (the first 20).

around the peak of  $G(\beta)$ .

The SBM implementation<sup>2</sup> lets us choose the size of the image middle square portion that enters the computations. We tested two options,  $512 \times 512$  for which processing of one image took 160 sec, and  $720 \times 720$ , taking 316 sec when run on Intel Dual Xeon 2.93 GHz (6 core). In comparison, running the LPM for a  $720 \times 1280$  image took 17.2 sec (with  $n = 500$  in Step 2).

Rotation angles in the experiments are unevenly distributed within the interval  $[0, 45]$ , with an added angle of  $-1.22$  degrees for rotation in opposite direction. More tests are performed for small angles as we anticipate them to be more frequently occurring in practice. The purpose of the unconventional choice of the set of angles is to make it convincing that the estimation methods do not work for special values only, such as for integer degrees.

### Uncompressed images

In the first test, uncompressed images are rotated by  $\alpha$  degrees clockwise using bicubic interpolation, rounded to 8-bit and cropped to its valid rectangular part removing all padded areas. As shown in Table 3 (shortened to fit the first 20 images), the LPM is highly reliable and very precise for most images. This table shows the success rate with which the angle was estimated with 0.1-degree accuracy, i.e. the estimation error  $\delta = |\hat{\alpha} - \alpha| \leq 0.1$ . The LPM can not correctly estimate rotation angles for image 'clean\_walls' because  $\max e(\mathbf{L})$  is close to 1.

Taking the statistical significance of the peak that lead to the estimate  $\hat{\alpha}$ , we can for example discard the estimates that are not backed up by significant peaks. The success rate “With confidence” in Table 3 is based on the signifi-

<sup>2</sup>Available for download from github.com

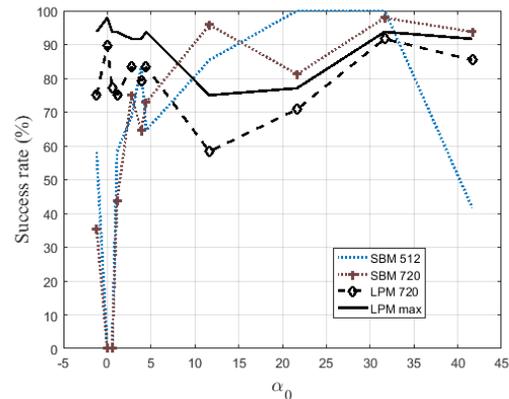
$\alpha \rightarrow$	-1.22	0	0.55	1.11	2.77	3.88	4.44	11.66	21.66	31.66	41.66
'barrier'	-1.22	-0.01	0.55	1.11	2.76	3.89	4.44	11.66	21.66	31.68	41.66
'beach_wood'	-1.22	0.01	0.55	1.11	2.76	3.88	4.44	11.66	21.66	31.65	41.65
'berries'	-1.22	-0.01	0.55	1.11	2.78	3.88	4.44	11.66	21.66	31.66	41.65
'bricks'	-1.22	-0.01	0.55	1.11	2.76	3.88	4.44	11.66	21.66	31.66	41.66
'cattle'	-1.24	-0.04	0.55	1.13	2.93	3.88	4.43	11.67	<i>37.60</i>	31.66	<i>28.07</i>
'central_park'	-1.21	0.01	0.55	1.11	2.78	3.89	4.45	<i>30.10</i>	21.66	31.65	41.67
'christmas_hedge'	-1.22	-0.01	0.55	1.11	2.78	3.89	4.44	11.66	21.66	31.66	41.65
'clean_walls'	<i>4.35</i>	<i>0.19</i>	<i>-0.07</i>	<i>-0.08</i>	<i>-1.20</i>	<i>-2.80</i>	<i>3.63</i>	<i>44.25</i>	<i>36.91</i>	<i>31.15</i>	<i>41.78</i>
'dark_and_bright'	-1.22	-0.01	0.55	1.11	2.78	3.88	4.44	11.66	21.66	31.66	41.66
'disconnected_shift'	-1.21	-0.01	0.55	1.11	2.78	3.89	4.44	11.66	21.66	31.66	41.65
'egyptian'	-1.21	-0.01	0.55	1.11	2.76	3.88	4.44	11.66	21.66	31.66	41.63
'extension'	-1.22	0.01	0.55	1.10	2.78	3.88	4.45	11.66	21.66	31.69	41.66
'fisherman'	-1.22	-0.01	0.55	1.11	2.78	3.89	4.43	<i>38.20</i>	<i>40.09</i>	31.67	41.66
'fountain'	-1.21	0.01	0.55	1.11	2.78	3.88	4.44	11.65	<i>40.10</i>	31.65	41.66
'four_babies'	-1.22	0.01	0.55	1.12	2.78	3.88	4.45	11.66	21.67	31.66	41.66
'giraffe'	-1.21	-0.07	0.55	1.13	2.75	3.84	4.45	11.63	21.67	31.69	41.75
'hedge'	-1.21	-0.01	0.55	1.11	2.78	3.89	4.44	11.66	21.66	31.65	41.66
'horses'	-1.21	-0.01	0.55	1.11	2.76	3.89	4.44	11.66	21.66	31.66	41.64
'japan_tower'	-1.21	-0.01	0.55	1.11	2.78	3.88	4.44	11.66	21.66	<i>5.09</i>	41.65
'jellyfish_chaos'	-1.22	0.01	0.55	1.11	2.78	3.88	4.44	11.66	21.66	31.66	41.65
...	...	...	...	...	...	...	...	...	...	...	...
Success count	45/48	47/48	45/48	45/48	44/48	44/48	45/48	36/48	37/48	45/48	44/48
With confidence	43/43	45/45	42/43	43/43	43/43	43/44	43/44	33/41	36/42	44/45	42/45

**Table 3.** LPM applied to full dimension images (the first 20 test images shown). Estimates with  $\delta > 0.1$  are in italic.

cance level of  $10^{-5}$  (assuming a Gaussian distribution for  $G(\beta) - P(\beta)$ ). This rate for example excludes estimates for the test image 'clean\_walls' due to lack of peaks. There are still a few errors due to occasional occurrence of false peaks.

The SBM works best and better than the LPM for large angles with the exception of  $\alpha = 41.66$  in which case the larger block size (720) helps. The SBM always fails to find 0-degree and 0.55 degree rotations. This suggests that rotations by less than 1 degree is beyond the capability of this method. Then, we see a high number of incorrect estimations involving digits 1, 7, and 9, particularly angles 7.17, 7.19, and 17.09 (and 1.79 for image 'fountain'). Incorrect "guesses" are not random, they tend to fall on a small subset of angles that somehow depend on the true rotation angle. We hypothesize that these errors are due to misinterpreting CFA interpolation peaks in the frequency domain. In their original work [1], test images from the Dresden database were downsampled by a factor of 2 in order to remove CFA artifacts that would otherwise cause the confusion. The test set is also narrowed down by choosing just one camera, the Nikon D200, from which all the test images were sampled from.

A summary of the results are in Table 5 and Figure 6. The value in the first column by "SBM" means the block size, "LPM 720" denotes the LPM applied to the cropped rotated image of size  $720 \times 1280$  (whenever the original image size was not smaller) and "LPM max" denotes the LPM applied to the maximum rectangular image size with padded areas removed. The LPM always performed better with the largest cropped size and yet much faster than "SBM 512." The success rate is evaluated for  $\delta \leq 0.1$  and



**Figure 6.** Success rates of the LPM and SBM ( $\delta \leq 0.1$ ).

for  $\delta \leq 0.05$  degrees.

In order to emphasize the ability of the LPM to produce precise estimations for small angles  $|\alpha| \leq 3$ , more tests in this range are shown in Table 6. The average success count with confidence (as above) was 42.8. The success rate with confidence is shown in the table after dividing the success counts by the number of all cases with statistically significant peaks.

### Down-sampling by 2

This downsampling factor was chosen for two reasons. First, it is commonly applied to decrease the filesize of digital photos. The number of pixels matches the number of green pixels in the most common RRGB-type CFA. Sec-

$\alpha \rightarrow$	-1.22	0	0.55	1.11	2.77	3.88	4.44	11.66	21.66	31.66	41.66
'barrier'	<i>1.23</i>	<i>24.86</i>	<i>-0.67</i>	<i>-1.12</i>	2.80	3.92	4.48	11.62	21.66	31.67	41.60
'beach_wood'	-1.23	-3.47	-3.47	1.12	2.80	3.92	4.48	11.62	21.66	31.67	<i>17.09</i>
'berries'	<i>1.23</i>	<i>39.98</i>	<i>9.83</i>	1.12	2.80	3.92	4.48	11.62	21.66	31.67	<i>17.09</i>
'bricks'	-1.23	<i>35.05</i>	<i>7.14</i>	1.12	2.80	3.92	4.48	11.62	21.66	31.67	<i>17.09</i>
'cattle'	<i>1.23</i>	<i>7.14</i>	<i>7.14</i>	<i>7.14</i>	<i>7.17</i>	<i>7.18</i>	<i>7.19</i>	<i>21.68</i>	21.66	31.67	41.68
'central_park'	-1.23	-5.60	-2.80	1.12	2.80	3.92	4.48	11.73	21.66	31.67	<i>17.09</i>
'christmas_hedge'	-1.23	<i>-14.50</i>	<i>-43.01</i>	1.12	2.80	3.92	4.48	11.62	21.66	31.67	<i>17.09</i>
'clean_walls'	-1.23	-2.80	-0.67	1.12	2.80	3.92	4.48	11.73	21.66	31.67	41.60
'dark_and_bright'	<i>1.23</i>	<i>-5.72</i>	<i>0.67</i>	1.12	2.80	3.92	4.48	11.73	21.66	31.67	41.60
'disconnected_shift'	-1.23	<i>1.46</i>	<i>-0.67</i>	1.12	2.80	3.92	4.48	11.73	21.66	31.67	41.60
'egyptian'	-1.23	<i>-0.781</i>	<i>7.39</i>	1.12	2.80	3.92	4.48	11.62	21.66	31.67	<i>17.09</i>
'extension'	-1.23	3.36	3.36	-1.12	2.80	3.92	4.48	11.62	21.66	31.67	<i>17.09</i>
'fisherman'	-1.23	<i>-26.36</i>	<i>7.14</i>	1.12	<i>7.17</i>	3.92	<i>7.19</i>	11.62	21.66	31.67	<i>17.09</i>
'fountain'	<i>1.79</i>	<i>-1.79</i>	<i>1.79</i>	<i>1.79</i>	<i>1.79</i>	<i>1.79</i>	<i>1.80</i>	11.73	21.66	31.67	41.60
'four_babies'	-1.23	4.24	7.14	-1.12	<i>7.17</i>	3.92	<i>7.19</i>	11.62	21.66	31.67	<i>17.09</i>
'giraffe'	<i>1.23</i>	<i>7.25</i>	<i>7.14</i>	1.12	<i>7.17</i>	3.92	<i>7.19</i>	11.75	21.66	31.67	41.67
'hedge'	-1.23	-7.14	7.14	7.14	<i>7.17</i>	<i>7.18</i>	<i>7.19</i>	11.73	21.66	31.67	41.60
'horses'	-1.23	-2.46	6.25	-1.12	2.80	3.92	4.48	11.62	21.66	31.67	<i>17.09</i>
'japan_tower'	-1.23	-3.02	-3.13	1.12	2.80	3.92	4.48	11.62	21.66	31.67	41.67
'jellyfish_chaos'	<i>1.23</i>	<i>3.58</i>	<i>7.14</i>	1.12	<i>7.17</i>	3.92	<i>7.19</i>	<i>21.68</i>	21.66	31.67	<i>-24.18</i>
...	...	...	...	...	...	...	...	...	...	...	...
Success count	28/48	0/48	0/48	28/48	33/48	40/48	31/48	41/48	48/48	48/48	20/48

Table 4. SBM, block size 512. Estimates with  $\delta > 0.1$  are in italic.

$\alpha \rightarrow$	-1.22	0	0.55	1.11	2.77	3.88	4.44	11.66	21.66	31.66	41.66
SBM 512	28 (28)	0 (0)	0 (0)	28 (28)	33 (33)	40 (40)	31 (31)	41 (41)	48 (48)	48 (48)	20 (20)
SBM 720	17 (17)	0 (0)	0 (0)	21 (21)	36 (36)	31 (27)	35 (34)	46 (45)	39 (38)	47 (46)	45 (45)
LPM 720	36 (35)	43 (40)	37 (36)	36 (35)	40 (39)	38 (38)	40 (40)	28 (28)	34 (34)	44 (43)	41 (41)
LPM max	45 (45)	47 (44)	45 (44)	45 (45)	44 (44)	44 (44)	45 (45)	36 (36)	37 (38)	45 (45)	44 (41)

Table 5. The number of correct angle estimations with error  $\delta \leq 0.1$  degree and with  $\delta \leq 0.05$  (in parentheses), out of 48.

ond, all images in the original work [1] were downsampled this way, yet before rotation. The success count of the SBM dropped on average more than twice, see Table 7.

### Resizing by factor 0.8

This is the most damaging post-rotation processing among all presented in this work. Coincidentally, signs of resampling by factor 4/5 with bicubic interpolation kernel are the hardest to detect among all resampling factors (shown by Pasquini and Böhme in 2017 [8]). Most incorrect estimates obtained from the SBM carried systematic errors. This confirms that resizing can be considered as an effective attack on the SBM [1]. The LPM succeeds in roughly 50% of tests. One exception is at zero (no rotation) where the LPM dealt really well with it at success rate 47/48 (Table 8 and Figure 7(b)).

### JPEG compression

JPEG compression suppresses the LP and decreases its energy. Therefore, a drop in performance of both tested methods is expected.

Setup: Images are first JPEG compressed at 90% quality, then rotated by a small angle of 2.77 degrees (with bicubic interpolation) and finally JPEG compressed with quality factors between 80 and 100 or saved uncompressed. This is a single angle test.

The LPM performs better than the SBM only for JPEG quality factors larger than 90 (Table 9). When the SBM estimate is incorrect it is mostly confused for the angle of  $-14.62$  degrees, which could be caused by the first compression interpolation peaks in DFT.

### Discussion

The main limitation of the LPM could be the lens distortion correction automatically performed in long zoom compact cameras at the short end of the zoom. This correction is a geometric transform that distorts the linear pattern. If the distortion is severe, the LPM may fail if the lens distortion transform is not undone.

Although it is possible to implement the LPM using Radon transform, it may not work correctly or be faster, depending on what interpolation the transform does and how it handles padding. For example, the Matlab (R2016a) implementation is not suitable.

### Conclusions

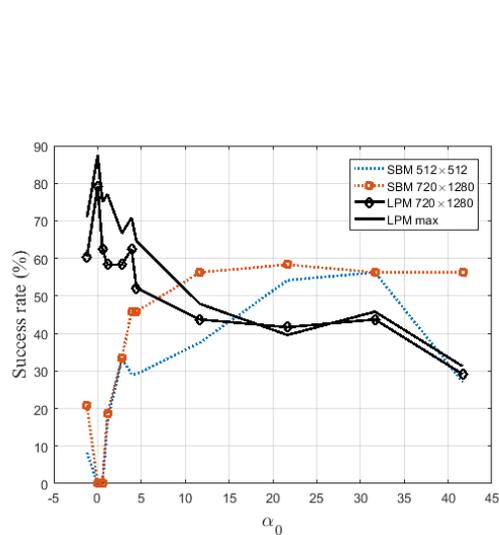
Estimating rotation angle using the proposed LP-based method without any reference signal or side information is possible and efficient in terms of computation speed and the likelihood of obtaining the correct estimate with high precision. The LPM is robust to image resizing and rather higher quality JPEG compression.

$\alpha \rightarrow$	-3	-2.6	-2.2	-1.8	-1.4	-1	-0.6	-0.2	0	0.2	0.6	1	1.4	1.8	2.2	2.6	3
$\delta \leq .1$	45	44	45	45	46	44	46	43	47	43	45	44	45	46	44	45	43
$\delta \leq .05$	44	44	44	44	46	44	45	42	44	43	45	44	45	46	44	44	43
$\delta \leq .2$	46	45	46	46	47	44	47	45	48	44	45	44	46	47	45	47	45
With confidence	.98	.98	1	1	.98	1	1	.95	1	.98	.98	1	1	1	1	1	1

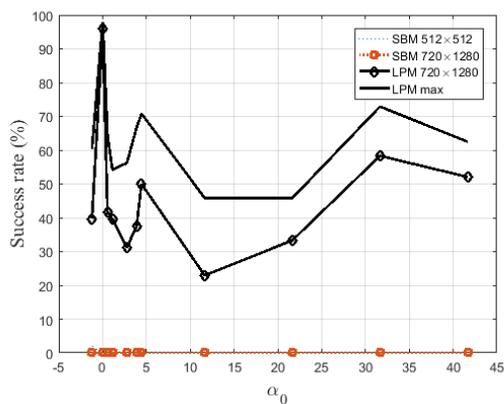
Table 6. The number of correct angle estimations (out of 48) by the LPM at maximum available image size for small angles.

$\alpha \rightarrow$	-1.22	0	0.55	1.11	2.77	3.88	4.44	11.66	21.66	31.66	41.66
SBM 512	4 (4)	0 (0)	0 (0)	8 (8)	16 (15)	14 (13)	14 (14)	18 (13)	26 (26)	27 (26)	13 (7)
SBM 720	10 (10)	0 (0)	0 (0)	9 (9)	16 (16)	22 (18)	22 (22)	27 (27)	28 (28)	27 (27)	27 (26)
LPM 720	29 (24)	38 (34)	30 (26)	28 (27)	28 (27)	30 (28)	25 (23)	21 (21)	20 (12)	21 (15)	14 (14)
LPM max	34 (32)	42 (38)	36 (34)	37 (36)	32 (31)	34 (32)	31 (30)	23 (20)	19 (16)	22 (19)	15 (12)

Table 7. The number of correct angle estimations with error  $\delta \leq 0.1$  degree and with  $\delta \leq 0.05$  (in parentheses), out of 48. Resized factor  $d = 0.5$ , i. e., downsampling by 2.



(a)



(b)

Figure 7. Success rates for resized images, a) downsampling by factor of 2, b) resizing at 0.8 ( $\delta \leq 0.1$ ).

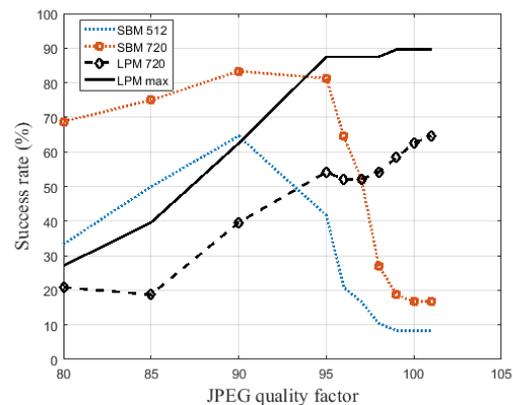


Figure 8. Success rates for JPEG images rotated by 2.77 degrees and recompressed at varying JPEG quality or uncompressed (shown at quality 101) ( $\delta \leq 0.1$ ).

The main advantage of LPM is in estimating small rotation angles where methods based on analysis of DFT signal (spectral methods) mostly fail. Because the LPM works with the signal that has already been present in the image before rotation it is independent from the spectral based methods that rely on artifacts introduced at the final stage of image rotation. Therefore weaknesses of these different methods do not coincide. They could be combined for best performance.

Beside the general forensic purpose, reversing rotation of rotated frames in a hand-shake stabilized video will likely increase the quality of camera PRNU-based fingerprint extraction for source camera identification [9]. We believe that a combination of LP-based and PRNU-based approaches can improve and/or speed-up re-synchronization of traces of PRNU from individual video frames before averaging them into the final PRNU estimate.

## Acknowledgments

This material is based on research sponsored by FBI Forensic Audio, Video and Image Analysis Unit (FAVIAU) under contract number DJF-17-1200-P-0007116. The U.S. Government is authorized to reproduce and distribute

$\alpha \rightarrow$	-1.22	0	0.55	1.11	2.77	3.88	4.44	11.66	21.66	31.66	41.66
SBM 512	1 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)
SBM 720	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)
LPM 720	19 (17)	46 (43)	20 (19)	19 (19)	15 (15)	18 (17)	24 (21)	11 (11)	16 (14)	28 (25)	25 (25)
LPM max	29 (29)	47 (44)	31 (30)	26 (26)	27 (27)	32 (31)	34 (34)	22 (21)	22 (21)	35 (34)	30 (30)

**Table 8.** The number of correct angle estimations with error  $\delta \leq 0.1$  degree and with  $\delta \leq 0.05$  (in parentheses), out of 48. Resized factor  $d = 0.8$ , nearest neighbor interpolation.

JPEG quality $\rightarrow$	80	85	90	95	96	97	98	99	100	Uncompressed
SBM, 512	16 (16)	24 (24)	31 (31)	20 (19)	10 (10)	8 (8)	5 (5)	4 (4)	4 (4)	4 (4)
SBM 720	33 (33)	36 (36)	40 (40)	39 (39)	31 (31)	25 (25)	13 (13)	9 (9)	8 (8)	8 (8)
LPM 720	10 (10)	9 (8)	19 (15)	26 (25)	25 (24)	25 (22)	26 (25)	28 (27)	30 (29)	31 (29)
LPM max	13 (13)	19 (18)	30 (29)	42 (41)	42 (41)	42 (41)	42 (42)	43 (43)	43 (43)	43 (43)

**Table 9.** The number of correct angle estimations (out of 48) with error  $\delta \leq 0.1$  degree (with  $\delta \leq 0.05$  in parentheses) for JPEG images compressed at 90% quality before rotation and recompressed at a range of qualities after rotation by 2.77 degrees.

reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of FAVIAU or the U.S. Government.

Special thanks belong to Chenglong Chen for providing public access to the research code for their spectral method (SBM).

## Author Biography

Miroslav Goljan received the Ph.D. degree in Electrical Engineering from Binghamton University in 2002 and the M.S. in Mathematical Informatics from Charles University in Prague, Czech Republic, in 1984. He is Research Scientist at the Dept. of Electrical and Computer Engineering at Binghamton University. His research focuses on digital image and digital camera forensics, steganography, steganalysis, and reversible data hiding in digital media.

## References

- [1] C. Chen, J. Ni, and Z. Shen. Effective estimation of image rotation angle using spectral method. *IEEE Signal Processing Letters*, 21(7):890–894, July 2014.
- [2] Mo Chen, Jessica Fridrich, and Miroslav Goljan. Digital imaging sensor identification (further study). In *Proceedings of SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, pages 0P 1–13, January 2007.
- [3] T. Filler, J. Fridrich, and M. Goljan. Using sensor pattern noise for camera model identification. In *Proc. ICIP 2008*, pages 1296–1299, San Diego, CA, October 12–15, 2008.
- [4] M. Goljan and J. Fridrich. Camera identification from scaled and cropped images. In E. J. Delp et al., editor, *Proceedings of SPIE, Electronic Imaging, Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, volume 6819, pages 0E 1–13, San Francisco, CA, January 2008. SPIE.
- [5] M. Goljan and J. Fridrich. Estimation of lens distortion correction from single images. In *Proceedings of SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2014*, volume 9028, pages N 01–13, San Francisco, CA, February 2014. International Society for Optics and Photonics.
- [6] M. Goljan, J. Fridrich, and T. Filler. Large scale test of sensor fingerprint camera identification. In N.D. Memon, E.J. Delp, P.W. Wong, and J. Dittmann, editors, *Proceedings of SPIE, Electronic Imaging, Media Forensics and Security XI*, volume 7254, pages 0I 1–12, San Jose, CA, January 2009. SPIE.
- [7] M. K. Mihcak, I. Kozintsev, and K. Ramchandran. Spatially adaptive statistical modeling of wavelet image coefficients and its application to denoising. In *Proceedings IEEE, International Conference on Acoustics, Speech, and Signal Processing*, volume 6, pages 3253–3256, Phoenix, AZ, March 15–19, 1999.
- [8] C. Pasquini and R. Böhme. Information-theoretic bounds of resampling forensics: New evidence for traces beyond cyclostationarity. In *Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security, IH&MMSec '17*, pages 3–14, New York, NY, USA, 2017. ACM.
- [9] S. Taspinar, M. Mohanty, and N. Memon. Source camera attribution using stabilized video. In *2016 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6, Dec 2016.
- [10] D. Vázquez-Padín, C. Mosquera, and F. Pérez-González. Two-dimensional statistical test for the presence of almost cyclostationarity on images. In *2010 IEEE International Conference on Image Processing*, pages 1745–1748, Sept 2010.
- [11] W. Wei, S. Wang, X. Zhang, and Z. Tang. Estimation of image rotation angle using interpolation-related spectral signatures with application to blind detection of image forgery. *IEEE Transactions on Information Forensics and Security*, 5(3):507–517, Sept 2010.