RST-Invariant Semi-Fragile Image Authentication Based on DWT-Coefficient Quantization

Mathias Schlauweg and Erika Müller,

Institute of Communications Engineering,
Faculty of Computer Science and Electrical Engineering,
University of Rostock, Rostock 18119, Germany,
{mathias.schlauweg, erika.mueller}@uni-rostock.de

Abstract. In this paper, we propose an image moment-based geometric normalization to be applied before embedding and extracting a digital watermark in the DWT-domain of JPEG2000. A semi-fragile signature, generated from the normalized host signal, afterwards, is embedded for image authentication. The new system is tested extensively and performance results are compared to those of methods proposed by other authors. Our new semi-fragile image authentication is robust against non-malicious modifications, such as lossy compression, noise, image blurring and sharpening, changes of luminance and contrast as well as scaling, rotation, translation, and shearing.

Keywords: Watermarking, discrete wavelet transform-domain, image moment-based geometric normalization, JPEG2000.

1 Introduction

During the last decade, growing applications of digital technologies in the field of multimedia resulted in various advantages. Digital images can be created easily and at a reasonable price. They can be copied without quality loss and changed without special knowledge. But, these properties can also yield disadvantages. For example, it is hard to assert rights of authors and owners and to proof the authenticity of images. For example, every year, there is a spectacular image content manipulation revealed in any famous print media. The repertoire reaches from correction of small blemish to dramatization of war reporting or political campaigns. Hence, images and video are in a credibility crisis.

To verify the authenticity without limiting user's customs additional data can be embedded within images by means of digital watermarks. For embedding, the multimedia signal is slightly changed. At the verification side, these signal changes can be detected and thus the embedded information can be retrieved. By checking the correctness of the extracted watermark a user can infer easily if the image has been tampered with.

Additionally embedded data should be robust against allowed image processing or compression format conversions. But, if the content of an image is tampered with, then an alarm should be raised during verification. Further, image distortions caused

by data embedding should be imperceptible and it should be impossible to manipulate the overall system.

These objectives are not met by any known system, so far. For that reason, in [1], we developed a digital watermarking system for efficient and tamper-proof image authentication. A digital watermark adapted to the image content is embedded imperceptibly by quantization of the coefficients of the discrete wavelet transform domain (DWT). This process is directly integrated into a JPEG2000 image compression and, hence, very efficient. The embedded watermark is robust against a variety of allowed image processing operations, e.g., JPEG and JPEG2000 compression, change of luminance and contrast, filtering, sharpening as well as scaling of image size.

To further enable watermark extraction after changes of image geometry, such as, rotation, translation or shearing, in this paper, an extension is presented using an image moment-based geometric normalization. In section 2, we describe a normalization procedure that is applied before embedding and extracting watermark data. In section 3, we present the integration of this normalization into the authentication framework proposed in [1]. The performance of the extended authentication system is extensively analyzed and compared to data of similar methods by other authors, in section 4. Finally, section 5 concludes our work.

2 Image Moment-Based Geometric Normalization

In [2], Dong *et al*. describe an image moment-based geometric normalization that is applied before embedding and extracting watermark data. Using this normalization the embedded watermark can be extracted even if the host image has been changed by rotation, scaling, translation (RST), or shearing.

Since RST as well as shearing in both x and y directions can all be considered as *affine transformations*, they can be inverted using one affine transformation at watermark extraction side.

To get a fixed orientation and scale of host image $I(x,y)$, that is the same during watermark embedding and extraction, Dong *et al*. calculate *geometric image moments* m_{pq} (see Eq. (1)) and *central moments* μ_{pq} (see Eq. (2)), where $M \times N$ is the size of I.

$$m_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} x^p y^q I(x,y). \tag{1}$$

$$\mu_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x-\bar{x})^p (y-\bar{y})^q I(x,y), \quad \text{where} \quad \bar{x} = \frac{m_{10}}{m_{00}}, \bar{y} = \frac{m_{01}}{m_{00}}. \tag{2}$$

Using these moments image I is centered (step 1) to achieve translation invariance. Afterwards, a shearing transform in the x direction is applied (step 2) followed by a shearing transform in the y direction (step 3). In a fourth step, the image is scaled in both x and y directions so that the resulting image achieves a prescribed standard size.

1) Center image $I(x,y)$ by calculating the coordinates $x^{(1)}$, $y^{(1)}$ of the transformed image $I^{(1)}(x^{(1)}, y^{(1)})$ using Eq. (3).

$$\begin{pmatrix} x^{(1)} \\ y^{(1)} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} - \begin{pmatrix} d_1 \\ d_2 \end{pmatrix}, \quad \text{where} \quad d_1 = \frac{m_{10}}{m_{00}}, d_2 = \frac{m_{01}}{m_{00}}. \tag{3}$$

2) Shear $I^{(1)}\left(x^{(1)}, y^{(1)}\right)$ by calculating the coordinates $x^{(2)}$, $y^{(2)}$ of the transformed image $I^{(2)}\left(x^{(2)}, y^{(2)}\right)$ using Eq. (4) so that the resulting image achieves $\mu_{30}^{(2)} = 0$.

$$\begin{pmatrix} x^{(2)} \\ y^{(2)} \end{pmatrix} = \begin{pmatrix} 1 & \beta \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x^{(1)} \\ y^{(1)} \end{pmatrix}, \quad \text{where} \quad \mu_{30}^{(1)} + 3\beta\mu_{21}^{(1)} + 3\beta^2\mu_{12}^{(1)} + 3\beta^3\mu_{03}^{(1)} = 0\,. \tag{4}$$

3) Shear $I^{(2)}\left(x^{(2)}, y^{(2)}\right)$ by calculating the coordinates $x^{(3)}$, $y^{(3)}$ of the transformed image $I^{(3)}\left(x^{(3)}, y^{(3)}\right)$ using Eq. (5) so that the resulting image achieves $\mu_{11}^{(3)} = 0$.

$$\begin{pmatrix} x^{(3)} \\ y^{(3)} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \gamma & 1 \end{pmatrix} \cdot \begin{pmatrix} x^{(2)} \\ y^{(2)} \end{pmatrix}, \quad \text{where} \quad \gamma = \frac{\mu_{11}^{(2)}}{\mu_{20}^{(2)}} = \frac{\mu_{11}^{(1)} + \beta\mu_{02}^{(1)}}{\mu_{20}^{(1)} + 2\beta\mu_{11}^{(1)} + \beta^2\mu_{02}^{(1)}}\,. \tag{5}$$

4) Scale $I^{(3)}\left(x^{(3)}, y^{(3)}\right)$ by calculating the coordinates $x^{(4)}$, $y^{(4)}$ of the transformed image $I^{(4)}\left(x^{(4)}, y^{(4)}\right)$ using Eq. (6) so that the resulting image achieves a prescribed standard size (e.g., 512×512) and $\mu_{50}^{(4)} > 0$ as well as $\mu_{05}^{(4)} > 0$.

$$\begin{pmatrix} x^{(4)} \\ y^{(4)} \end{pmatrix} = \begin{pmatrix} \alpha & 0 \\ 0 & \delta \end{pmatrix} \cdot \begin{pmatrix} x^{(3)} \\ y^{(3)} \end{pmatrix}. \tag{6}$$

Fig. 1 visualizes this normalization by means of four example images. As can be seen, the normalization of a square image yields a rotated, scaled, sheared, and possibly mirrored image.

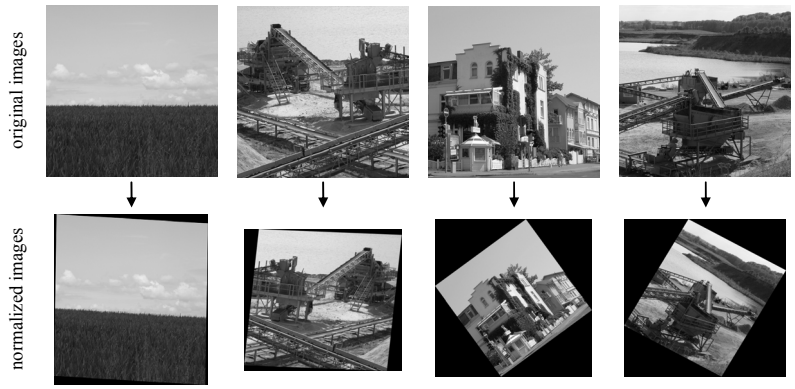

Fig. 1. Examples: image moment-based geometric normalization by Dong *et al*. [2]

The RST-invariant watermarking approach proposed by Dong *et al*. is based on the fact that image I and its affine transforms (geometrically distorted images) all have

the same normalized image. The authors generate a pseudo-random-based 2-D signal with the same size as the normalized image, apply the inverse affine transform to this signal, and add it to the original image (using *spread spectrum watermarking*).

3 Image Authentication with Geometric Normalization

Inspired by the above described approach by Dong *et al*. [2], in this paper, we extend our image authentication watermarking system proposed in [1]. For that, we change the normalization procedure (subsection 3.1) to adapt it to the watermark generation and embedding/extraction of the JPEG2000-based image authentication (see Fig. 2), which is described in detail in subsections 3.2 - 3.4.

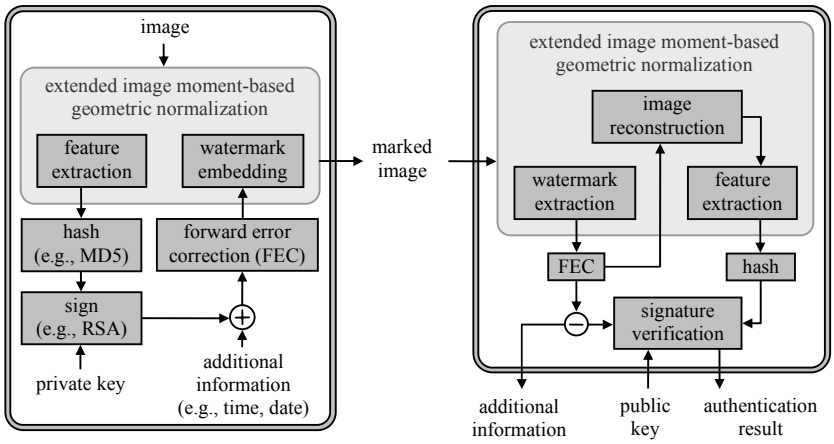

Fig. 2. Digital watermarking system for image authentication

3.1 Extended Normalization Procedure

After applying the calculations of step 1 to step 4 (section 2), we know the parameters α, β, γ, and δ. Using these parameters we can determine the positions of the four corners $\mathrm{P}_i := \left\{ \mathrm{P}_i \left(x_i, y_i \right) : i = 1,...,4 \right\}$ (shown in Fig. 3) of the normalized image $I^{(4)}$.

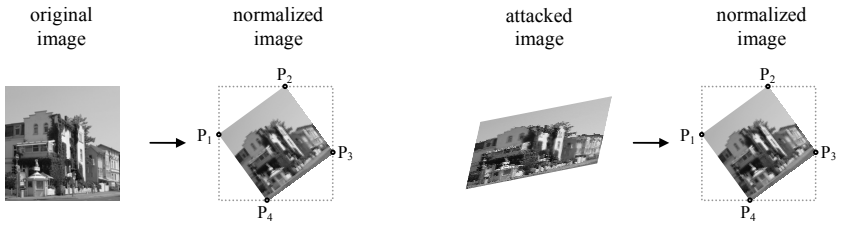

Fig. 3. Determination of the four corners of the host image using the proposed normalization

Our extended normalization procedure is based on stretching the corners P_1, …, P_4 to the corners P_1', …, P_4' of a fixed-size square region during watermark generation/ embedding as well as watermark extraction/verification, as demonstrated in Fig. 4. This stretching operation is a further shearing transform of the image in the *x* direction (step 5) followed by a shearing transform in the *y* direction (step 6) together with an image scaling to the fixed size (step 7).

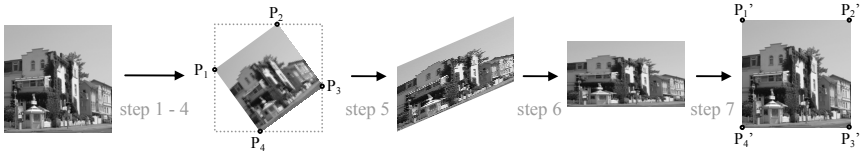

Fig. 4. Extended image moment-based normalization procedure

All these transform steps can be combined into one single normalization procedure (Eq. (7)) to determine the coordinates *x*' and *y*' of the normalized image $I'(x', y')$. Hence, the overall computational efforts as well as the induced image distortions are very low. The inverse transform uses the same parameters $\alpha, \beta, \gamma, \delta$ (see Eq. (9)).

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & \beta \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ \gamma & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & \beta' \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ \gamma' & 1 \end{pmatrix} \cdot \begin{pmatrix} \alpha & 0 \\ 0 & \delta \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}. \tag{7}$$

$$\beta' = \arctan\left(\frac{x_4 - x_1}{y_4 - y_1}\right),\ \gamma' = \arctan\left(\left(\frac{y_4 - y_3}{x_3 - x_4} + \tan\beta\right)^{-1}\right),$$
$$\alpha = \frac{512}{x_3 - x_4 + (y_4 - y_3)\cdot\tan\beta},\ \delta = \frac{512}{y_4 - y_1}. \tag{8}$$

$$\begin{pmatrix} x \\ y \end{pmatrix} = \left(\begin{pmatrix} 1 & \beta \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ \gamma & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & \beta' \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ \gamma' & 1 \end{pmatrix} \cdot \begin{pmatrix} \alpha & 0 \\ 0 & \delta \end{pmatrix}\right)^{-1} \cdot \begin{pmatrix} x' \\ y' \end{pmatrix}. \tag{9}$$

Now, watermark generation/embedding as well as extraction/verification can take place using the normalized image $I'(x', y')$ as described in what follows.

3.2 Watermark Generation and Embedding

As opposed to the approach by Dong *et al*., our new authentication system is based on quantization of the coefficients of the host image in the *discrete wavelet domain* (DWT). It is directly integrated in the process of a JPEG2000 compression.

Construction of a Secure Image-Dependent Hash. If $x := \{x_j \in \mathbb{R} : 1 \le j \le J\}$ are the coefficients of an image in DWT-domain and q_j is a quantized value using quantizer $Q(\bullet)$ and step-size Δ, then $\hat{x}_j = Q^{-1}(q_j)$ is the reconstructed value of q_j, as in Eq. (10) and Eq. (11).

$$q_j = Q(x_j) = \operatorname{sign}(x_j)\left\lfloor \frac{|x_j|}{\Delta} \right\rfloor. \tag{10}$$

$$\hat{x}_j = Q^{-1}(q) = \begin{cases} 0 & q = 0 \\ \operatorname{sign}(q)(|q| + 0{,}5)\Delta & q \neq 0 \end{cases}. \tag{11}$$

In numerous simulations, we found out that if we quantize and, afterwards, hash all coefficients $x := \{x_n \in \mathbb{R} : 1 \le n \le N\}$ of the LL_4-subband of the DWT-decomposition a secure and robust image-dependent hash-value can be constructed.

As long as the quantized coefficients $\hat{x}$ after changes due to image processing operations or attacks remain within the range $[\Delta\mathbb{Z}; \Delta(\mathbb{Z}+1))$ they yield the same hash-value during verification. If a forger moves just one single LL_4-coefficient out of its quantization interval this manipulation can be detected and alarm is raised.

A digital signature is generated from the hash-value by the use of asymmetric encryption (e.g., *RSA*) with a key of length 512 bits. Additionally to the hash-value also time, date, etc. can be integrated to make the shot unique (see Fig. 2).

Afterwards, the signature is encoded using forward error correction. We apply *convolutional coding* (code rate r = 1/2). Hence, the watermark $w := \{w_n \in \pm 1 : 1 \le n \le N\}$ to be embedded has a length of 1024 bits.

Signature Embedding by Quantization. For our semi-fragile authentication approach it is sufficient not to embed the signature watermark as robust as possible but as robust as necessary. That means, if an image processing operation or an attack yields a different hash during verification it doesn't matter if the signature can be extracted correctly. Signature and hash-value don't match, and hence, verification fails.

For that reason, we embed the data within the same host signal locations the signature is generated from using scalar *dither modulation* [3]. Hence, the embedding locations are secured by the hash process in turn.

Since JPEG2000 applies quantization with dead-zone, our watermark embedding is adapted to this dead-zone as in Eq. (12), where y is the watermarked host signal.

$$y_n = \begin{cases} x_n & -\Delta/4 \le x_n \le \Delta/4 \\ & \text{and } w_n = +1 \\ \Delta \cdot \left(\operatorname{sign}(x_n) \left\lfloor \frac{|x_n|}{\Delta} \right\rfloor + w_n \cdot \Delta/4 \right) & \text{otherwise} \end{cases}. \tag{12}$$

Data is embedded by quantizing every LL_4-coefficient to a closest quantization lattice point of one of two subsets of lattices $\Lambda_{w_n} = \Delta\mathbb{Z} + w_n\Delta/4$. In Fig. 5, these lattice points are marked by either × or ○.

Luminance and Contrast Normalization. Since we use LL-subband coefficients for signature generation as well as embedding, the host image and the quantization step-

size have to be normalized prior watermarking to allow luminance and contrast adjustment operations.

For that reason, in a first step, the host signal is normalized to the mean pixel luminance (subtraction of gray-value pixel mean). In a second step, the quantization step-size is normalized to contrast. As in Eq. (13), a factor g is computed from the pixel values of image $I := \{I_j \in \mathbb{N} : 0 \le I_j \le 255, 1 \le j \le J\}$. Prior to hashing and signature embedding, the step-size is divided by factor g, where the same process takes place during signature verification.

$$g = \frac{1}{256}\left(\frac{1}{J}\sum_{j=1}^{J} I_j^{\,2}\right)^{1/2} . \tag{13}$$

A contrast change, now, becomes a scaling of factor g, and hence, can be reversed similar to the normalization proposed by Pérez-González *et al*. in [4].

Further, we embed g as a second watermark in the HL_4-, LH_4-, and HH_4-coefficients using the same strategy as for the LL_4-subband. Thereby, g is represented by 32 bits and encoded using *repeat-accumulate coding* with a code rate of $r = 1/96$. The resulting 3072 bits are embedded using a small step-size, whereby there occur no further perceptual embedding distortions.

3.3 Watermark Extraction and Hash-Intervall Error Correction

To extract the watermark data the host signal is quantized to the nearest neighbor lattice point of one of the two quantizer subsets. Afterwards, the extracted signature has to be compared with the hash-value generated from the received image for content integrity verification.

As mentioned before, the hash-value remains constant as long as the quantized LL_4-coefficients don't leave the interval $\left[\Delta\mathbb{Z};\Delta(\mathbb{Z}+1)\right)$. But, due to embedding the coefficients are moved to the lower or upper half of the quantization interval, respectively. Hence, even image processing operations changing the LL_4-coefficients more than $\Delta/4$ yield the verification to fail.

To solve this problem we extended the watermark bit error correction as follows. If $\hat{w} = [-1;+1]$ denotes the watermark data extracted from the received host signal $\hat{y} \in \mathbb{R}$ and $\tilde{w} = \{-1,+1\}$ is the corrected watermark after FEC-decoding, then Eq. (14) can be applied to correct the hash intervals.

$$q_n = \operatorname{sign}(\hat{y}_n) \cdot \left\lfloor \frac{|\hat{y}_n|}{\Delta} + \frac{\hat{w}_n - \tilde{w}_n}{8} \right\rfloor . \tag{14}$$

As demonstrated in Fig. 5, in that way, the hash interval is expanded to the range $\left[\Delta(\mathbb{Z}-1/4);\Delta(\mathbb{Z}+3/4)\right)$ or $\left[\Delta(\mathbb{Z}+1/4);\Delta(\mathbb{Z}+5/4)\right)$, respectively, depending on the watermark bit at the appropriate location. Hence, despite data embedding the coefficients can be changed up to $\Delta/2$ without affecting images authenticity. That way, the overall robustness is gained by a factor of two.

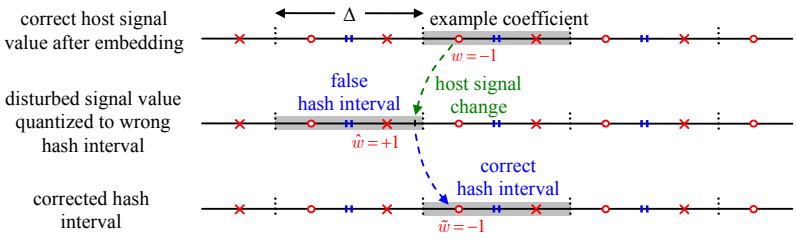

Fig. 5. Example: reconstruction of hash interval by combining hash-value quantization and watermark bit error correction

3.4 Adaptation of Step-Size Δ Based on Image Content

The choice of embedding strength (step-size Δ), and hence, the robustness of the hash as well as the signature are limited by the visual perception of embedding induced distortions. As shown in Fig. 6, if the same step-size is used for all LL_4-coefficients watermark embedding is not optimal.

The *human visual system* is less sensitive to changes in textured regions than in smooth regions of an image. That means, the choice of embedding strength is mainly limited by the visual perception of distortions in homogenous regions such as the cloud-free sky in the example image.

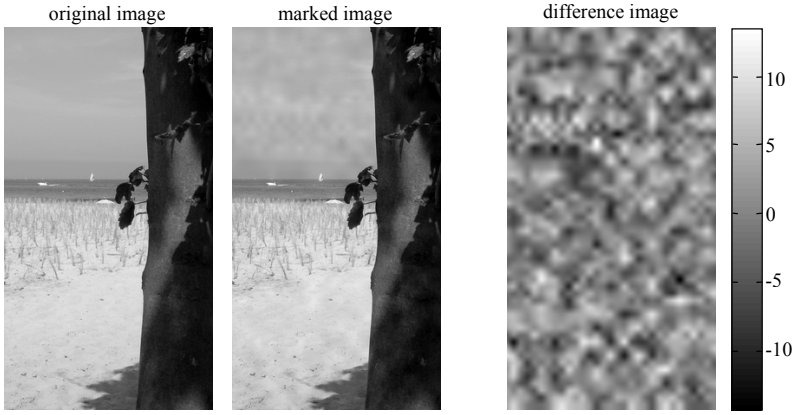

Fig. 6. Example: image distortions caused by signature generation and embedding using the same step-size Δ = 8 for all LL_4-subband coefficients

To improve the performance of our authentication system we use different step-sizes. We separate the image into homogenous regions and stronger textured regions. For signature generation and embedding within the LL_4-coefficients representing the former regions we use step-size Δ_1. For all the rest we use Δ_2.

In Fig. 7, marked images are shown using non-adaptive as well as adaptive embedding. Although the PSNR-values are similar for the left and middle image, distortions cannot be seen for the adaptively marked image in the middle.

Fig. 7. Example: (a) marked image using non-adaptive embedding, where $\Delta_1 = \Delta_2 = 6$, resulting in PSNR = 40.89 dB, (b) marked image using texture-based step-size adaptation, where $\Delta_1 = 3$ and $\Delta_2 = 9$, resulting in similar PSNR = 40.98 dB, and (c) contrast-enhanced difference of (b) to the original image

For the texture-based image region separation we use the coefficients of the third DWT-decomposition level. As visualized in Fig. 8, except for the LL_3-subband all these coefficients are compared to a threshold τ. Afterwards, the three matrices are added and 2×2 block-wise averaged. Finally, the known morphologic operations *closing* and *erosion* are applied to refine the separation. The resulting matrix $F \in \mathbb{R}$ we call feature mask.

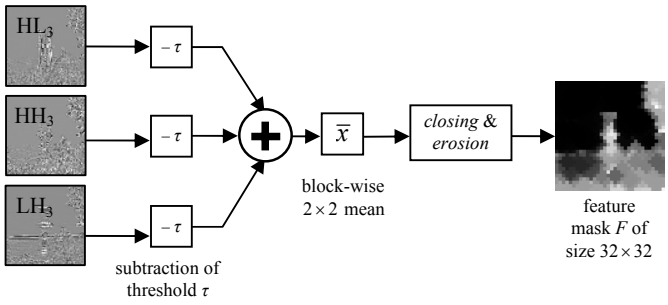

Fig. 8. Texture-based feature mask generation

Compared to the original image, homogenous regions yield negative values. For stronger textured regions feature F is positive. Hence, during watermark embedding, we apply Δ_1 for all locations where $F < 0$, otherwise, we apply Δ_2, if $F \geq 0$.

During watermark extraction, we apply adaptive decoding. We use the separation feature $\hat{F}$ computed from the received image to weight the extracted watermark signal during FEC-decoding. We use the certainty of how close the texture feature is to the feature threshold τ. If the feature is close to the decision threshold ($\hat{F}$ tending to zero), it is uncertain which quantization lattice has to be used during extraction. In this case, the certainty tends to zero. If the feature is far from the threshold and it is sure which lattice was chosen during embedding, then the certainty is high.

At the decoding side, we separate the received host signal into two sub-signals $\hat{w}_1 = Q'(\hat{y})$ and $\hat{w}_2 = Q''(\hat{y})$, where $Q'(\bullet)$ denotes the quantizer that uses the step-

size Δ_1 and $Q''(\bullet)$ denotes the quantizer that uses the step-size Δ_2. Afterwards, $\hat{w}_1$ and $\hat{w}_2$ are weighted using the two functions $f_1(\hat{F})$ and $f_2(\hat{F})$. Details can be found in [1].

$$f_1(\hat{F}) = \begin{cases} 1 & , -\infty < \hat{F} < -\alpha \\ \frac{1}{2}\left(1+\cos\left(\frac{\hat{F}+\alpha}{\alpha}\cdot\frac{\pi}{2}\right)\right) & , -\alpha \le \hat{F} < +\alpha \\ 0 & , +\alpha \le \hat{F} < +\infty \end{cases}. \quad (15)$$

$$f_2(\hat{F}) = \frac{\Delta_2}{\Delta_1}\cdot\begin{cases} 0 & , -\infty < \hat{F} < -\alpha \\ \frac{1}{2}\left(1-\cos\left(\frac{\hat{F}+\alpha}{\alpha}\cdot\frac{\pi}{2}\right)\right)\cdot\beta(\hat{F},\hat{w}_2) & , -\alpha \le \hat{F} < +\alpha \\ 1 & , +\alpha \le \hat{F} < +\infty \end{cases}, \quad (16)$$

$$\text{where} \quad \beta(\hat{F},\hat{w}_2) = 1-\sin\left(\frac{\hat{F}+\alpha}{\alpha}\cdot\cos\left(\hat{w}_2\cdot\frac{\pi}{2}\right)\right). \quad (17)$$

By applying Eq. (18), the two sub-signals are joint resulting in watermark signal $\tilde{w}$, which is the input to the soft-decision FEC-decoder (e.g., *Viterbi algorithm*).

$$\tilde{w} = \frac{\hat{w}_1\cdot f_1(\hat{F}) + \hat{w}_2\cdot f_2(\hat{F})}{2}. \quad (18)$$

4 Experimental Results

4.1 Robustness Simulations

In Fig. 9, we present the results of robustness simulations for our new semi-fragile image authentication system. For these simulations we used a set of 52 different gray-scale images of size 512×512 pixels. For LL_4-coefficient hashing we applied the *message digest algorithm 5* (MD5) yielding a hash-value of length 128 bits. We used RSA for signing the hash (512 bits key length) and convolutional coding for error correction (code-rate r = 1/2). Hence, 1024 bits were embedded within the LL_4-sub-band (32×32 coefficients) of every host image.

As can be seen, the developed image authentication is robust against a variety of image processing operations. Using the extended image moment-based normalization procedure it is also robust against rotation as well as shearing.

By the use of subjective tests and simulations, we found out that choosing Δ_1 = 3 and Δ_2/Δ_1 = 3 yield the best compromise between perceptual image quality and robustness.

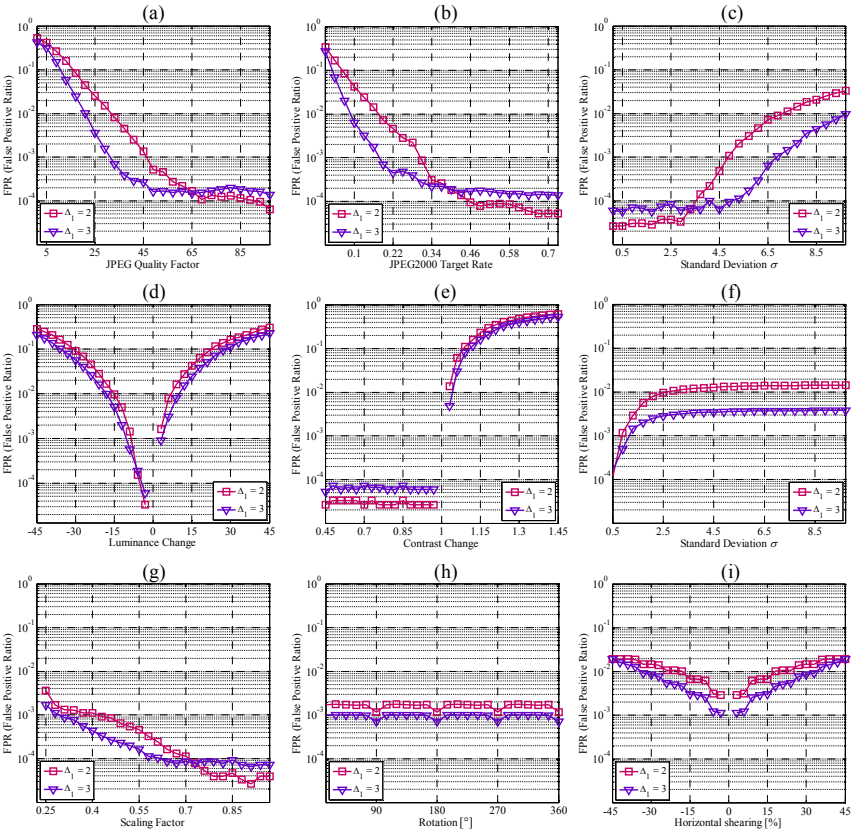

Fig. 9. Results of simulations for the overall extended semi-fragile image authentication system - robustness against: (a) JPEG compression, (b) JPEG2000 compression, (c) Gaussian noise, (d) luminance change, (e) contrast change, (f) Gaussian low-pass filtering (g) scaling of image size, (h) rotation, and (i) shearing of the image in x direction. Parameters: $\Delta_2/\Delta_1 = 3$, $\tau = 1.5$, $\alpha = 5$.

4.2 Comparison with Methods by Other Authors

An authentication watermark should be robust against non-malicious image processing but fragile against image content attacks. Most authentication approaches focus too much on robustness and neglect security. Good comparative overviews of different semi-fragile image authentication methods can be found in [5]-[7].

To compare the performance of our system with those of methods by other authors we use the results collected by Ekici *et al*. in [5]. Table 1 shows that our authentication system (Schlauweg *et al*.) performs better in most cases. Since it was not possible to find any image authentication that tested robustness against rotation, translation, scaling, or shearing, we cannot compare our performances for these operations.

We think that for applicability of an authentication system it is important that the system is secure. Hence, we highlight that P_{miss} (forgery attack) is zero for our system.

Semi-fragile method	Forgery attack P_{miss}	Signal-processing attacks P_f								
		No attack	Smooth	Histog. equal.	S and P 1%	AWGN 35 dB	JPEG 70	Sharpen	Random errors
Chang *et al.*	0,0 %	0,0 %	100 %	99,0 %	100 %	32,3 %	0,0 %	100 %	0,0 %
Delp *et al.*	0,1 %	2,3 %	54,5 %	3,4 %	6,5 %	2,7 %	2,4 %	0,3 %	14,1 %
Eggers *et al.*	0,0 %	0,0 %	41,4 %	91,0 %	2,6 %	0,0 %	0,0 %	65,6 %	2,5 %
Fridrich	1,0 %	1,6 %	62,0 %	5,5 %	19,5 %	2,5 %	25,8 %	21,0 %	2,5 %
Kundur *et al.*	0,1 %	0,0 %	77,7 %	99,5 %	51,9 %	10,0 %	2,9 %	98,1 %	0,1 %
Queluz	0,01 %	0,01 %	27,8 %	94,3 %	42,7 %	0,01 %	0,01 %	100 %	1,1 %
Liao *et al.*	8,7 %	3,0 %	34,3 %	80,7 %	43,3 %	1,7 %	1,5 %	79,9 %	4,2 %
Schlauweg *et al.*	0,0 %	0,0 %	0,0 %	100 %	100 %	0,0 %	0,0 %	43,7 %	0,0 %

Table 1. False alarm and miss probabilities for comparison of performance of our approach with results of other authentication methods as given in [5] (embedding induced PSNR = 41dB)

5 Conclusion

This paper presents the embedding of a digital watermark for image authentication within images normalized using geometric moments. During JPEG2000 compression, a semi-fragile signature is generated from image content and embedded by quantization of the coefficients in the DWT-domain. Generation as well as embedding of the signature is adapted to the image content using texture-based image region separation. Our image authentication is tested extensively and performance results are compared to those of methods proposed by other authors. The semi-fragile authentication is robust against non-malicious modifications, such as lossy compression, noise, image blurring and sharpening, changes of luminance and contrast as well as scaling, rotation, translation, and shearing.

References

1. Schlauweg, M. and Müller, E.: Content-adaptive semi-fragile image authentication based on JPEG2000 compression. In: Proc. of 16th IEEE International Conference on Digital Signal Processing, Santorini, Greeece, (2009)
2. Dong, P., Brankov, J. G., Galatsanos, N. P., Yang, Y., and Davoine, F.: Digital watermarking robust to geometric distortions. In: IEEE Transactions on Image Processing, vol. 14 (12), pp. 2140--2150, (2005)
3. Chen, B. and Wornell, G.: Quantization index modulation: a class of provably good methods for digital watermarking and information embedding. In: IEEE Transactions on Information Theory, vol. 47 (4), pp. 1423--1443, (2001)
4. Pérez-González, F., Mosquera, C., Barni, M., and Abrardo, A.: Rational dither modulation: a novel data hiding method robust to value-metric scaling attacks. In: Proc. of 6th IEEE Workshop on Multimedia Signal Processing, Siena, Italy, pp. 139--142, (2004)
5. Ekici, Ö., Sankur, B., Coşkun, B., Naci, U., and Akcay, M.: Comparative evaluation of semifragile watermarking algorithms. In: Journal of Electronic Imaging, vol. 13 (1), pp. 209--216, (2004)
6. Zhu, B. B., Swanson, M. D., and Tewfik, A. H.: When seeing isn't believing. In: IEEE Transaction on Signal Processing, vol. 21, pp. 40--49, (2004)
7. Rey, C and Dugelay, J.-L.: A survey of watermarking algorithms for image authentication. In: EURASIP Journal of Applied Signal Processing, vol. 6, pp. 613--621, (2002)