SIViP (2012) 6:211–222
DOI 10.1007/s11760-011-0211-2

ORIGINAL PAPER

Toward an interactive poster using digital watermarking and a mobile phone camera

Anu Pramila · Anja Keskinarkaus · Tapio Seppänen

Received: 14 May 2009 / Revised: 13 January 2011 / Accepted: 14 January 2011 / Published online: 6 February 2011

Abstract A method for embedding a watermark in print media, posters or other paper printouts and reading the watermark information blindly with a camera phone is proposed. A subtractive-additive embedding method is applied in which the message is coded with a directed periodic pattern. The message is detected and read by searching regularities in the autocorrelation function of a periodic signal. The robustness to disturbance occurring during printing process due to air interface and camera phone properties is ensured using noise reduction, modified JND model, enhanced peak detection with filtering and shaping and two-level coding of the message. The validity of the approach is proven with tests, and an application example of an interactive poster is examined.

Keywords Camera phones · Digital Watermarking · Print-cam process

1 Introduction

Watermarking has been conventionally used as a way to hide information to the content, and reading the hidden information with a camera phone would make the usage of watermarks independent of time and place. The watermark could be, for example, a link to some website, a piece of music or an initiation for a new service. Most of the proposed watermarking methods are not robust enough for applications where content is recorded from one medium to another. For example, in the print-cam process, a watermark is read with a camera phone from a printed image without a need for special equipment such as readers and scanners.

In this paper, an autocorrelation and directed periodic pattern-based watermarking method, robust to the print-cam process, is presented. A watermark in an poster is utilized as a link to a website. It should be noted that the method does not require any user interaction, cropping or resizing of the image, after capture. The user is assumed to have some knowledge of his/her camera phone but not about watermarking. We focus the research to the scenario in which the watermark on the poster offers valuable application information to the user who, thus, has no intention to attack it and remove it. The scenario differs in this respect from digital rights management scenarios. However, we consider carefully the unintentional attacks on the watermark that are due to the printing and imaging processes of the poster.

A process in which the image is first printed and then scanned is called a print-scan process. The most notable distortions are rotation, scaling, translation and cropping. The printing and scanning operations also flatten the pixel values of the image and induce noise, which makes the extraction of the watermark difficult [1–3]. Various ways to achieve robustness against rotation, scaling and translation have been proposed. These methods include template watermarks [4], self-synchronization [5,6], transform invariant domains [7] and feature point detection [8] based methods.

In the print-cam process, the watermark must, in addition to the print-scan process, be robust against 3-dimensional distortions, such as tilt of the optical axis and variations in the environment like lighting and reflections. The camera itself

The financial support from the National Technology Agency of Finland (Tekes) and the Zirion project consortium is gratefully acknowledged.

A. Pramila (✉) · A. Keskinarkaus · T. Seppänen
Department of Electrical and Information Engineering,
University of Oulu, P.O.Box 4500, 90014 Oulu, Finland
e-mail: anu.pramila@ee.oulu.fi

A. Keskinarkaus
e-mail: anja.keskinarkaus@ee.oulu.fi

T. Seppänen
e-mail: tapio.seppanen@ee.oulu.fi

inflicts various attacks to the watermark, for example, JPEG compression and lens distortions [9,10]. In addition, there may be additional disturbances on top of the images such as logos and writing. The print-cam process can thus be seen as a collection of various attacks.

Print-cam robust watermarking methods are yet few. Na-kamura et al. [11] and Katayama et al. [12] presented a print-cam robust watermark detection method which used a frame around the image to correct geometrical distortions. The locations of the corners of the frame were utilized to approximate and correct the transformations [12]. However, even though frame-based methods are in general very robust to geometrical distortions, they are nevertheless vulnerable to cropping.

Kim et al. [13] embedded the message repeatedly in the form of a pseudorandom vector. In order to read the message, the lattice formed by the message was detected with an autocorrelation function and the message read with crosscorrelation. The results obtained were limited, however: The amounts of rotation, scale, translation and tilt of the optical axis were minimized by using a tripod and resizing and cutting the image by hand to its original dimensions [13].

Unlike the frame, autocorrelation function is invariant to small amounts of cropping, thus offering a viable alternative. The method presented here relies on autocorrelation rather than solely on detecting correlation, in effect preferring detection of the alignment and direction of the autocorrelation peaks. The application area of the method presented here is a special case of data hiding. However, data hiding is a very broad concept and includes also various steganographic techniques. In order to be consistent with the previous literature [9–12], the more specific term, digital watermarking, is used in here.

A multibit watermark message is embedded in the image as sequences of bits which are encoded into a rotational angle and a 2D periodic pattern is rotated according to the angles. The directed 2D periodic patterns are embedded in the image block by block. One of the image blocks serves as a reference, so that the directions of the patterns can be recovered later on and the existence of the watermark detected. If the watermark has not been detected, the message is not read.

The message is read by examining the configuration of the autocorrelation peaks from the image. Hough transform is used in order to gain angle information from the regularly aligned peaks and the angle is interpreted as the message.

We show with various experiments that multiple bits can be read even after the print-cam process when distortions and desynchronization are severe. First, the embedding and extracting processes are explained in detail, respectively, in Sects. 2 and 3 and then the results obtained are shown in Sect. 4. An example application is discussed in Sect. 5.

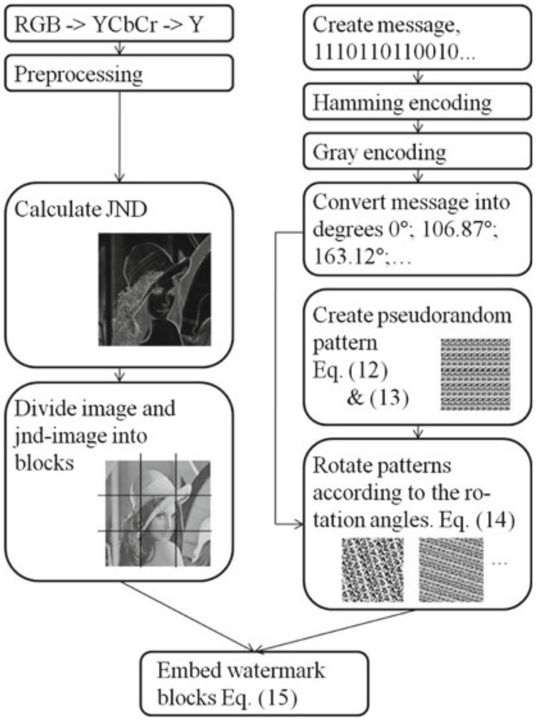

RGB -> YCbCr -> Y
Preprocessing
Calculate JND
Divide image and jnd-image into blocks
Create message, 1110110110010...
Hamming encoding
Gray encoding
Convert message into degrees 0°; 106.87°; 163.12°;...
Create pseudorandom pattern Eq. (12) & (13)
Rotate patterns according to the rotation angles. Eq. (14)
...
Embed watermark blocks Eq. (15)

Fig. 1 The embedding process

2 Message embedding

The order of operations involved is depicted in Fig. 1. The embedding is performed in two tracks: One processes the image and the other prepares the message.

2.1 Preprocessing determination of JND for message embedding

In order to enhance watermark robustness, the image is preprocessed. When the watermark is read, the image is Wiener filtered and the noise component is handled as the watermark information. The preprocessing is conducted in order to increase the amount of watermark in the noise component and to break periodicities originally existing in the image.

$$I_{pre}(x,y) = \begin{cases} I_{wiener}(x,y), & \text{when } r(x,y) = 1 \text{ and} \\ & (\mathrm{I(x,y)} - \mathrm{I_{wiener}(x,y)}) > 0, \\ I(x,y), & \text{otherwise} \end{cases} \tag{1}$$

where I_{wiener} is a Wiener-filtered version of the image, r is a random sequence with the same size as the image and consists of 80% 1s and 20% 0s. In order to break periodicies, 100% filtering would often soften the details too much. Therefore, less than 100% filtering was used.

In the print-cam process, the quality of the image is evaluated after printing, and therefore, the conventional JND (just noticeable difference) methods may not work well here.

The print-cam process flattens the intensity values of the images and thus also blurs the details in the images.

Chou and Li [14] proposed a JND method that calculates JND values with background luminance and spatial masking. The strongest component of the two was chosen as the JND value. This method, however, was designed for image coding and focuses on maintaining visual quality in digital form.

In order to achieve robustness against print-cam process, we modify the method by Chou and Li. The spatial masking component that affects the embedding strength around the details of the images is emphasized, and the spatial masking and background luminance components are summed together instead of selecting the stronger component. This is done in order to take into account the small details of the image. The JND is calculated with equations

$$JND(x,y) = \lambda_1 * (f_1(bg(x,y), mg(x,y)) + \lambda_2) + f_2(bg(x,y)) \quad (2)$$

$$f_1(bg(x,y), mg(x,y)) = mg(x,y) * \alpha(bg(x,y)) + \beta(bg(x,y)) \quad (3)$$

$$f_2(bg(x,y)) = \begin{cases} T_0 * (1 - (bg(x,y)/127)^{1/2}) + 3 \\ \text{for } bg(x,y) \leq 127 \\ \gamma * (bg(x,y) - 127) + 3 \\ \text{for } bg(x,y) > 127 \end{cases} \quad (4)$$

$$\alpha(bg(x,y)) = bg(x,y) * 0.0001 + 0.115 \quad (5)$$

$$\beta(bg(x,y)) = \lambda - bg(x,y) * 0.1, \quad (6)$$

where f_1 is the spatial masking component, $bg(x,y)$ and $mg(x,y)$ are average background luminance and maximum weighted average of luminance differences around the pixel at (x,y), respectively, as in [14]. $\alpha(x,y)$ and $\beta(x,y)$ specify the slope of the f_1 and the intersection with the visibility threshold axis. The visibility threshold due to background luminance is given by function f_2. Chou and Li found the T_0, γ and λ to be 17, 3/128 and 1/2, respectively. λ_1 and λ_2 are scaling factors. In this paper, we have used values 2.0 and 3.0, respectively. The values were determined experimentally by watermarking the image with some values, printing the image and examining the image quality visually. The resulting values of the JND image are then scaled between 0 and 1. The $mg(x,y)$ and $bg(x,y)$ is determined with

$$mg(x,y) = \max_{k=1,2,3,4} \{|grad_k(x,y)|\} \quad (7)$$

$$grad_k(x,y) = \frac{1}{16} \sum_{i=1}^{5} \sum_{j=1}^{5} p(x-3+i, y-3+j) * G_k(i,j) \quad (8)$$

$$bg(x,y) = \frac{1}{32} \sum_{i=1}^{5} \sum_{j=1}^{5} p(x-3+i, y-3+j) * B(i,j) \quad (9)$$

$$G_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 3 & 8 & 3 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ -1 & -3 & -8 & -3 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} G_2 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 8 & 3 & 8 & 0 \\ 1 & 3 & 0 & -3 & -1 \\ 0 & 0 & -3 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 2 & 2 & 1 \\ 1 & 2 & 0 & 2 & 1 \\ 1 & 2 & 2 & 2 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$G_3 = transpose(G_2) \qquad G_4 = transpose(G_1), \quad (10)$$

where $p(x,y)$ denotes a pixel at (x,y) and $B(i,j)$ a weighted low-pass operator. More information about the equations is found in [14].

The grayscaled image and the JND image are then divided into blocks, and several bits are embedded in each block. The following section explains the message encoding and embedding.

2.2 Message encoding and embedding processing

The watermark is read and processed in blocks, and the capacity of the method depends on the number of bits embedded in each block. In our experiments, the image was divided into nine blocks and four bits were embedded in each block. One of the blocks was a sychronization block, and thus, the watermark capacity is here 32 bits.

The message is protected with Hamming (32, 6) error correction coding that is capable of correcting two bits. The error-coded message is then divided into blocks of four bits and coded with 4-bit Gray coding. Each Gray coded sequence of message is transformed into rotation angles by assigning each sequence a value between 0 to 180 degrees. The value is chosen by quantizing the rotation angles and assigning each of the values a number of bits, as illustrated in Fig. 2. The quantization angle is determined with equation

$$\alpha = 180^\circ/(2^m), \quad (11)$$

where m is the amount of bits embedded in each of the blocks.

When the direction of the periodic pattern is detected and if the direction is erroneously interpreted to the adjacent quantization step, the gray coding and Hamming coding together ensure that the message can be decoded correctly because only one bit chances between adjacent quantization steps.

Periodic patterns are formed for each block. Each pattern is formed by first repeating a small rectangular pseudorandom sequence until the sequence covers an area double the size of the block and forms a 2D pattern. Each pattern is then rotated according to the message and cut to the size of the

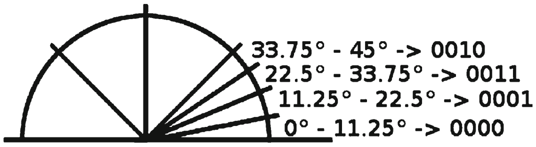


Fig. 2 Creating encoding table

block. The process does not affect periodicity. The pattern is then embedded in the image block. The first block is defined as a reference block, which is later used for watermark detection and as a reference for the other blocks when calculating the amount of rotation.

A periodic pattern satisfies the Eqs. (4) and (5)

$$W(x + q_0N_0, y) = W(x, y);\ q_0, N_0 > 1 \tag{12}$$

$$W(x, y + q_1N_1) = W(x, y);\ q_1, N_1 > 1, \tag{13}$$

where N_0 and N_1 determine the periodicity of repetitions and q_0 and q_1 a repetition number on the horizontal and vertical directions. Generation from pseudorandom values $\{-1, 1\}$ produces a rectangular, binary valued pattern. A directed periodic pattern is defined by

$$W^{\theta} = W(u', v') + \varepsilon = \Pi\left\{\begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} * W(u, v)\right\}, \tag{14}$$

where Π defines the interpolation method and ε describes the error caused by interpolation. Here, bilinear interpolation is used. θ defines the rotation angle of the Gray coded bit sequence.

The embedding of the message in the host image is realized in spatial domain utilizing equation

$$Y_i(x, y) = X_i(x, y) + \delta_1 * JND(x, y) * W_i^{\theta i}(x, y) + \delta_2 * (1 - JND(x, y)) * W_i^{\theta i}(x, y), \tag{15}$$

where Y_i is the ith watermarked block of the image, X_i is the preprocessed image, and the W_i^{θ} represents the coded watermark information in the form of directed periodic pattern, as in Fig. 1 and δ_1 and δ_2 are scaling factors for JND values.

3 Message detection and extraction

After capturing a picture with a camera, the captured image is divided into blocks. The existence of a watermark is tested by processing only the reference block, and if a watermark is detected, the message is read by processing the rest of the blocks.

The blocks are processed by calculating autocorrelation function for each of the blocks and examining the alignment of the autocorrelation peaks as illustrated in Fig. 3. The processing (P) needed for robust extraction due to distortions caused by print-cam process, as well as the message decoding from the attained angle information, is explained next.

The order of operations in reading message from a captured image is shown in Fig. 4. First, the captured image is downsampled by using bilinear interpolation. It was necessary to compromise between high processing time and amount of data processed. A square area is sectioned of the image by cutting the sides of the image and the consequent square image is divided into blocks. The watermark is detected and the message read block by block.

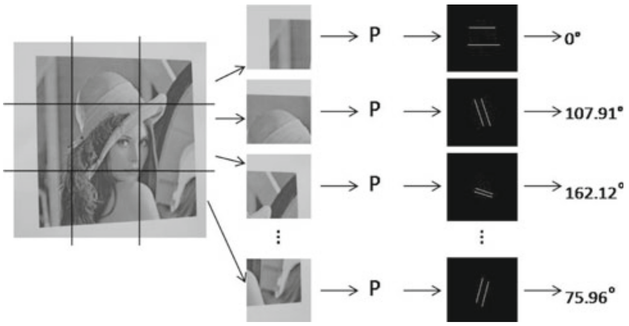


Fig. 3 The message extraction idea

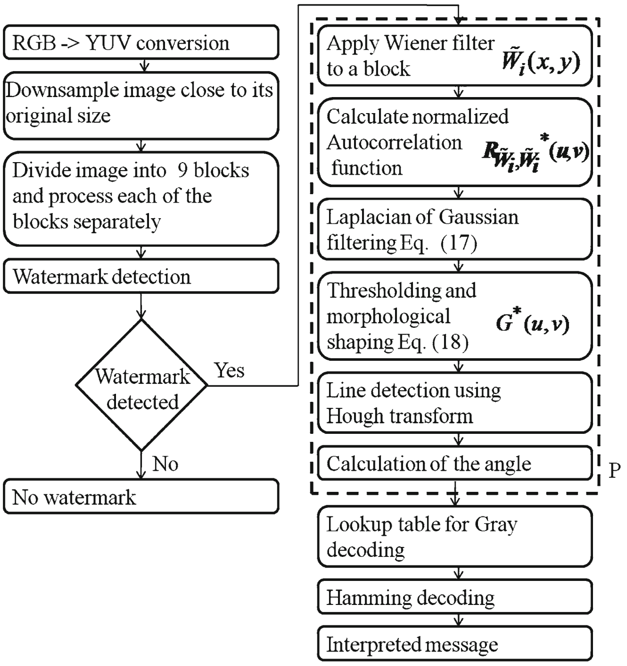

RGB -> YUV conversion
Downsample image close to its original size
Divide image into 9 blocks and process each of the blocks separately
Watermark detection
Watermark detected
Yes
No
No watermark
Apply Wiener filter to a block $\tilde{W}_i(x,y)$
Calculate normalized Autocorrelation function $R^*_{\tilde{W}_i\tilde{W}_i}(u,v)$
Laplacian of Gaussian filtering Eq. (17)
Thresholding and morphological shaping Eq. (18) $G^*(u,v)$
Line detection using Hough transform
Calculation of the angle
P
Lookup table for Gray decoding
Hamming decoding
Interpreted message

Fig. 4 Extraction process in detail

For each block, a Wiener estimate $\tilde{W}_i(x, y)$ of the periodic watermark structure is calculated

$$\tilde{W}_i(x, y) = Y_i^*(x, y) - h_w(k) * Y_i^*(x, y), \tag{16}$$

where $Y_i^*(x, y)$ is the ith watermarked block and $h_w(k)$ represents the adaptive Wiener filtering.

The autocorrelation function is calculated of the Wiener estimate, and this operation doubles the size of the processing block. The autocorrelation function $R^*_{\tilde{W}_i,\tilde{W}_i}(u, v)$ of Wiener estimate is severely distorted due to the combined effect of all the attacks (DA/AD, Jpeg compression, 3D distortions). In order to enhance peak detection, rotationally symmetric

Laplacian of Gaussian filtering operation is performed

$$R^{**}_{\tilde{W}_i \tilde{W}_i}(u, v) = \frac{\partial^2}{\partial(u, v)^2} h^T * \left(\frac{\partial^2}{\partial(u, v)^2} h * R^{*}_{\tilde{W}_i \tilde{W}_i}(u, v) \right) \qquad (17)$$

The enhanced autocorrelation peaks are then thresholded, and a binary grid is formed with equation

$$G^*(u, v) = \begin{cases} 1, \text{ when } M(u, v) \times R^{**}_{\tilde{W}_i \tilde{W}_i}(u, v) \geq \gamma \\ 0, \text{ when } M(u, v) \times R^{**}_{\tilde{W}_i \tilde{W}_i}(u, v) < \gamma \end{cases}, \qquad (18)$$

where $M(u, v)$ is a circular masking operation. The central area of the autocorrelation image contains noise, which causes errors in line detection. Therefore, the center of the grid is masked out. The threshold was chosen such that approximately 99.7% of the data are below the threshold.

Resampling while the picture is taken distorts the shape of the peaks. Therefore, the thresholded image is dilated with a small disk-shaped structuring element (radius 1) in order to reshape the peaks. This improves line detection because small errors are allowed in the peak locations while determining line directions.

Lines are searched from the binary grid using Hough Transform. The dominating direction, shown in Fig. 5, is found by evaluating the number of peaks allocated to the same bin in the Hough transform matrix. Due to the properties of the Hough transform, the possible false peaks in the autocorrelation function have little or no effect. However, it is important to locate as many of the correct peaks as possible for reliable determination of the angle.

The repetition number on both directions in the periodic pattern defines the robustness to cropping and perspective distortions. Our tests showed that by using a pseudorandom sequence of dimensions 28 × 7 during embedding, two desired properties can be achieved. First, the number of repetitions in each block is enough to conquer a substantial amount of translation and cropping. Second, the non-square rectangular form of the pseudorandom sequence reduces orthogonal errors in the line detection.

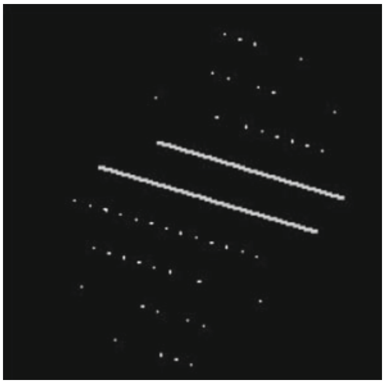


Fig. 5 The dominating direction found using Hough transform

Prior processing all the blocks, the watermark is detected. The reference block is processed until the lines and consequently the angle of the periodic pattern has been found. The autocorrelation peak pattern is then projected in a direction perpendicular to the angle of the periodic pattern. If the watermark exists, the projected pattern forms distinct peaks at regular intervals. The watermark is assumed to be present if

$$\sum_{i=1}^{5} p_i > a * \sum_{i=6}^{P} p_i, \qquad (19)$$

where p contains all the peaks sorted by their size, a is a threshold set here to 0.7, and P is the amount of peaks in total. The threshold was set experimentally and the results are shown later in Sect. 4.

The rest of the blocks are processed, and the final estimate of the angle is calculated using $\tilde{\theta}_k = \tilde{\theta}_r - \theta_i^*$, where θ_i^* is the calculated angle of the lines and $\tilde{\theta}_r$ is the angle of the reference block. The process is repeated for each block and the angle information is quantized. The same quantization step size and encoding table as during embedding is used. Each quantized angle value represents a coded bit sequence which is interpreted using a coding table and Gray decoding. Finally, Hamming (32, 6) error decoding is used to decode the message.

4 Experiments

4.1 Test images

The method was realized as a Matlab program, and five images shown in Fig. 6 with the size of 512 × 512 pixels were watermarked by using a different message for each image. The watermarked images were pinned on the wall, and all the following experiments were conducted in a normal office lightning. The processing time of the unoptimized code of the extraction algorithm per image was about 5 seconds with 2.66-GHz processor. The physical sizes of the printed images were 13.5 × 13.5 cm.

The images were watermarked with strengths $\delta_1 = 50$, $\delta_2 = 5$ and $\delta_1 = 60$, $\delta_2 = 6$. The qualities of the watermarked

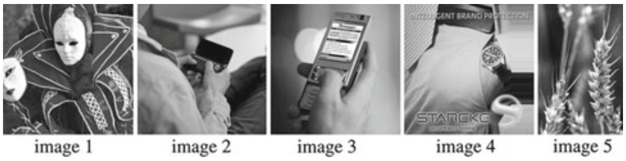


Fig. 6 The test images

Table 1 Subjective difference grades of the watermarked images

	$\delta_1 = 60, \delta_2 = 6$	$\delta_1 = 50, \delta_2 = 5$
Camou	−0.4	−0.5
Capricode	−0.7	−0.7
Starcke	0.1	0.3
Tekes	−0.2	−0.6
Nokia	−0.9	−0.5

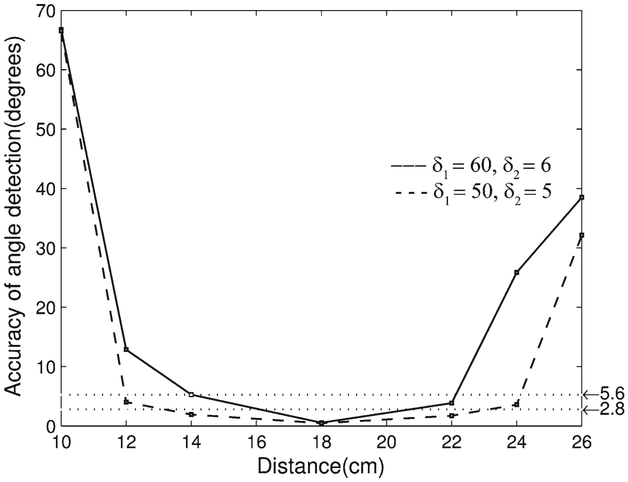


Fig. 7 Effect of distance on the accuracy of the angle detection

images were evaluated with subjective tests. The images were placed on a poster background, as in Fig. 10. and 10 persons were asked to evaluate the images without them knowing which images were watermarked. Subjects were asked to rate the images using a 5-point impairment scale: (5: imperceptible, 4: perceptible but not annoying, 3: slightly annoying, 2: annoying, 1: very annoying). Subjective difference grade (SDG) was calculated using the test results as:

$$SDG = Score_{image\_under\_test} - Score_{reference\_signal}, \tag{20}$$

where the ratings of the unwatermarked images were used as reference signals. Resulting SDG scores are shown in Table 1. The obtained results show that the qualities of the watermarked posters are sufficient for the considered use case.

4.2 Robustness against distance variations

In order to find out the shooting range at which the extraction is robust, the distance of the camera phone from the wall was gradually changed. The experiments were conducted with Nokia N82 mobile phone equipped with a 5 megapixel CMOS camera. Focal length of the camera is 5.6 mm and aperture F2.8. Tests were done on image printed on a white paper, using HP Color LaserJet 4650 PCL 6 printer.

The results are illustrated in Fig. 7 where the curves are calculated by processing one picture from watermarked and printed image at each distance. The image is divided into blocks, and the average accuracy of the rotation angles between blocks is calculated. The data are raw data, i.e. no error correction or rounding was applied. The angle 5.6° is marked to the figure, and it represents the theoretical limit at which the reading of the watermark is still reliable if 4 bits per block were embedded in the image. The 2.8° is the theoretical limit if 5 bits were embedded in each block. The limits are obtained by applying equation 3 where m is replaced with 4 and 5, respectively, and dividing the results with 2 in order to obtain the maximum acceptable error.

The tests illustrate that embedding more bits per block decreases the quantization step size and therefore the robustness because, as the quantization angle decreases, the possibility for misinterpreting the rotation angle information increases as well. The optimum shooting range for all the test images when 4 bits were embedded per block was from 12 − 24 cm, when watermark strength $\delta_1 = 60$ and $\delta_2 = 6$ and 14 − 22 cm with watermark strength $\delta_1 = 50$ and $\delta_2 = 5$. When the picture is taken from 12 cm, a large part of the image is cropped, as shown in Fig. 8a), and when taken from the distance of 24 cm, significant amount of background is captured along with the actual watermarked image area, as shown in Fig. 8b). Although the original block division is not maintained, due to periodicity, the message can still be retrieved correctly.

4.3 Robustness against 3D geometric distortions

In practice when the picture is taken freehandedly, some tilting will occur and thus the robustness of the method to tilting was estimated. The estimation was realized using Matlab to computationally simulate tilting with bilinear interpolation and perspective projection. A picture was taken with the camera perpendicularly to the printed image and tilted backward, forward, left and right gradually with the program and the percentage of correctly extracted bits measured. Results of these tests are depicted in Fig. 9 for image 1 and the results for the other images in "Appendix".

The tests showed that watermark strength of $\delta_1 = 50$ and $\delta_2 = 5$ is enough to extract the message correctly from all of the images when the picture is taken perpendicularly. The watermark strength of $\delta_1 = 60$ and $\delta_2 = 6$ instead allows more tilting to occur. For all the images and directions, extraction was robust to the average of 10 degrees, 6 degrees at the minimum and 20 degrees at the maximum.

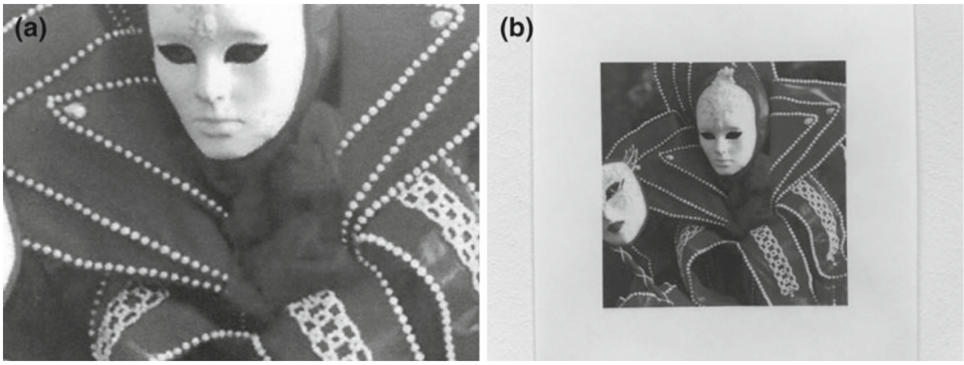


Fig. 8 **a** A picture taken from the distance of 12 cm. **b** A picture taken from the distance of 24 cm

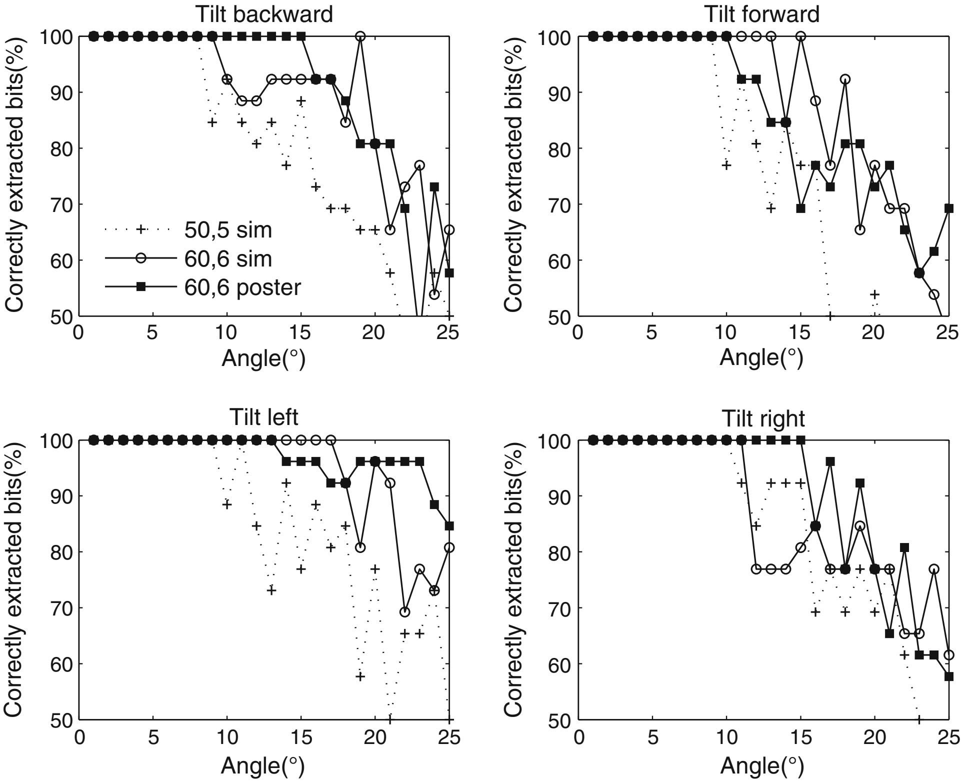
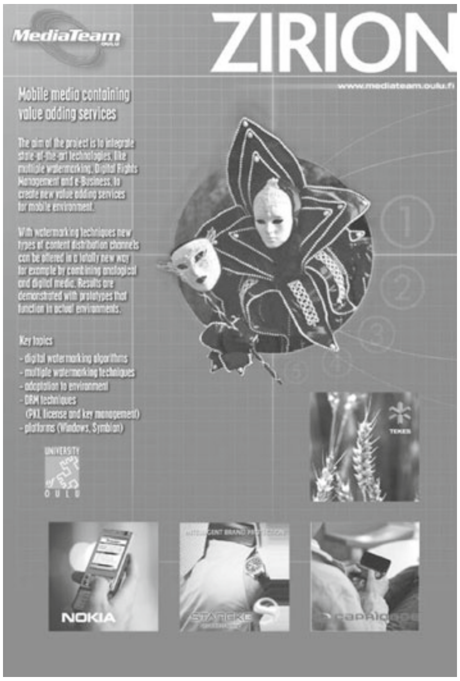


Fig. 9 The robustness to tilting for image 1

4.4 Robustness on poster layout

In a practical case, images rarely reside against white background, and what is more, there might be additional elements like logos on top of the images. To experiment the robustness of the algorithm in a more realistic situation, a poster layout was designed. All the images in the poster, shown in Fig. 10, contain a watermark, including the uppermost image in which the watermark was embedded in a rectangular area in the middle of the image. According to the results explained in Sect. 4.2, the watermark strength of $\delta_1 = 60$ and $\delta_2 = 6$ was selected. Notice that the visible lattice structure in the image background is just an artistic feature and not utilized at all in the extraction algorithm. The paper was common poster paper with somewhat shiny surface and the poster was printed with a Canon IPF 8000. The size of the poster was 56.5 × 84 cm, and the images in the poster were 13.5 × 13.5 cm as in the previous tests with plain office paper.

The robustness to perspective distortions was measured, similarly as for images with white background. The results are depicted in Fig. 9. For all the images and directions, extraction was robust to the average of 15 degrees, and 10 degrees at the minimum and 20 degrees at the maximum. The results showed that despite the additional distortions and shiny paper which easily causes unwanted reflections, performance of the method is better than the results explained



Mobile media containing value adding services

The aim of the project is to integrate state-of-the-art technologies, like multiple watermarking, Digital Rights Management and e-Business, to create new value adding services for mobile environment.

With watermarking techniques new types of content distribution channels can be offered in a totally new way for example by combining analogical and digital media. Results are demonstrated with prototypes that function in actual environments.

Key topics

- digital watermarking algorithms
- multiple watermarking techniques
- adaptation to environment
- DRM techniques
  (PKI, license and key management)
- platforms (Windows, Symbian)

Fig. 10 Poster with watermarked images

Table 2 False negatives and false positives of the watermark detection

	Watermark present	Watermark not present	Total
Detected	96	1	97
Not detected	4	99	103
Total	100	100	200

Table 3 BER when images were captured by tilting the camera randomly

	5 MPX N82	2 MPX N82	2 MPX N90
Camou	0.6	1.1	4.8
Capricode	1.7	0.3	2.7
Starcke	0.3	0.8	2.3
Tekes	2.5	5.6	3.1
Nokia	3.2	4.7	6.1

in Sect. 4.2. This is probably due to better printing quality of Canon IPF 8000 compared with HP Color LaserJet 4650 PCL 6.

4.5 Robustness with cameras

Two different camera phones were used, Nokia N82 released in 2007 and Nokia N90 released in 2005. N90 has a 2MPX CMOS camera, and for comparison, images were taken with two resolutions with N82, namely 5MPX and 2MPX. Based on the earlier results, the test images were watermarked with strength $\delta_1 = 60$ and $\delta_2 = 6$. The same images were used as in Sect. 4.2, so that the effects of the background of the images and the surface of the paper would be minimized.

Watermark detection results are shown in Table 2. Hundred images were captured randomly of the five watermarked images and 100 images of the unwatermarked images.

Robustness tests were conducted by rotating camera randomly ±10 degrees around the optical axis. Each of the watermarked test images was captured 50 times with each of the camera settings and each image contained 26 bits and error coding. This resulted in 1,300 bits per each image and camera setting. The obtained BER (Bit Error Rate) values are shown in Table 3.

The obtained results show that the camera quality has some effect on the reading of the watermark. The newer camera performed better when the same resolution was used. Probably, the lens distortions were the most effective factor.

5 Application example

An application prototype was implemented with Symbian C++ in order to gain information about the performance under a combination of attacks. The test phone used was Nokia N82, which has a 330-MHz processor. Due to autocorrelations calculations, the processing time was about 40 s if all the calculations were made in the phone and 15 s if the image was sent to a server for calculations.

As in [11–13], the application aim is to bridge the analog printout and Internet, enabling the users to get relevant web services without typing or searching.

Fifteen persons took pictures with the test phone from the poster and additionally tested the installed application. The users were guided regarding how to use the application. They were told to take the image straight and close enough but freehandedly. Feedback was gathered from the participants with an interview form. In Fig. 11, examples of the pictures taken are shown of cases when the whole message was retrieved correctly.

Most of the users were inexperienced with the camera phone model used and had rarely taken images with any camera phone. This was also visible from the resulting images as some of the images were not correctly focused. This seemed to be the largest individual problem. It is reasonable to assume that, in practice, application of this type will be used with the user's own mobile phone, and therefore, the user is experienced in using his/her own camera phone.

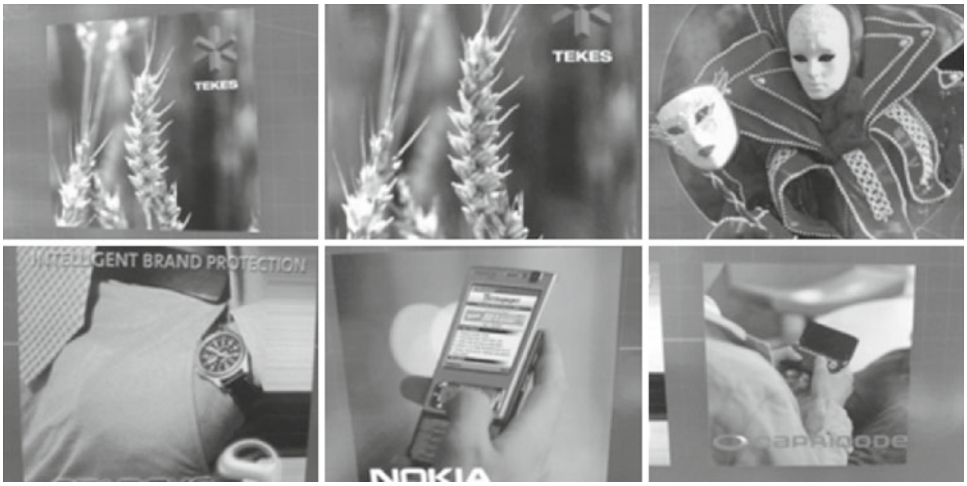


Fig. 11 Examples of pictures taken by users

While testing the application, 14 out of 15 were directed to the correct web page at the first attempt. All the users gave feedback that the application would be easy to learn and use.

6 Conclusion

In this paper, a watermarking method resilient to the print-cam process was proposed. Autocorrelation function was applied in order to find the alignment of the pseudo-random pattern and reading the multibit watermark message. The method was applied to the print-cam process successfully. The previously proposed methods that are robust against the print-cam process require either a frame around the image or human interaction. It should be noted that our method does not require frame or human interaction after the image has been captured. Furthermore, the method is blind and thus does not need the original message.

Comparison with the literature is challenging due to the different nature of the methods and evaluations. Katayama et al. and Nakamura et al. [11,12] focused on watermark detection and asked their test subjects to capture the image from a position right in front of the watermarked image to decrease the projection distortion, as well as capture the entire frame. These kinds of constraints were not applied in our tests but the camera was rotated 10° around the optical axis, sometimes resulting in cropping out part of the image.

The tests showed robustness to the print-cam process when shooting range was from 12 to 24 cm along with the consequent cropping and translation. Also for all the images and directions, extraction was robust to the average of 10° tilting from the final poster.

We showed that the method is robust against additional distortions such as logos placed on top of the images, reflections on shiny surface of the poster and background variations. The watermarks can be read even off a round image, despite that there is no clear boundary between the watermarked and unwatermarked area. If it can be ensured that there is no significant rotation present in the captured image, then the method is still able to find the message correctly and the reference block is not required.

An application prototype was implemented and feedback from users gathered. Although some uncertainty about handling an unfamiliar camera, no severe problems hindering the usage of the method in practical applications were detected. All the users found the application easy to use.

In the future, we hope to improve computational performance of the method. For example, threading can be applied as the method is well suited for concurrent processing. However, it can be expected that the mobile phone processors keep getting more powerful, and in few years, the method will offer real-timeliness to applications. In addition, robustness against second-generation capturing will be researched. At the moment, the method is not robust against attacks where the watermarked image is printed, scanned, printed again and then captured.

Appendix

See Figs. 12, 13, 14, and 15.

Fig. 12 The robustness to tilting for image 2

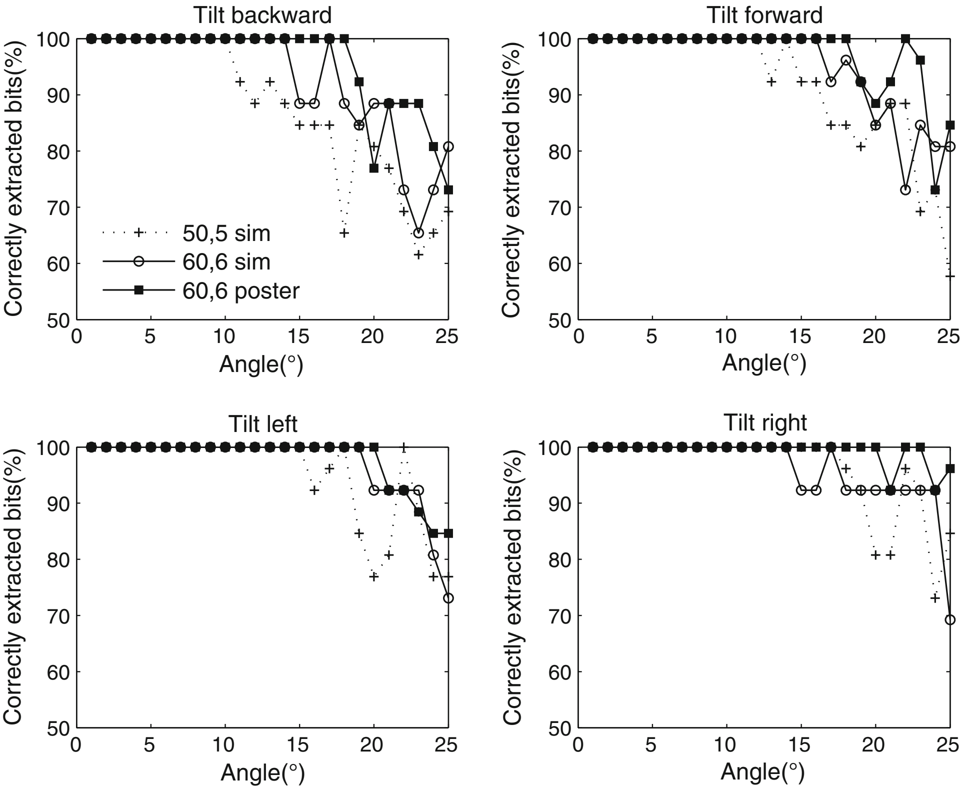


Fig. 13 The robustness to tilting for image 3

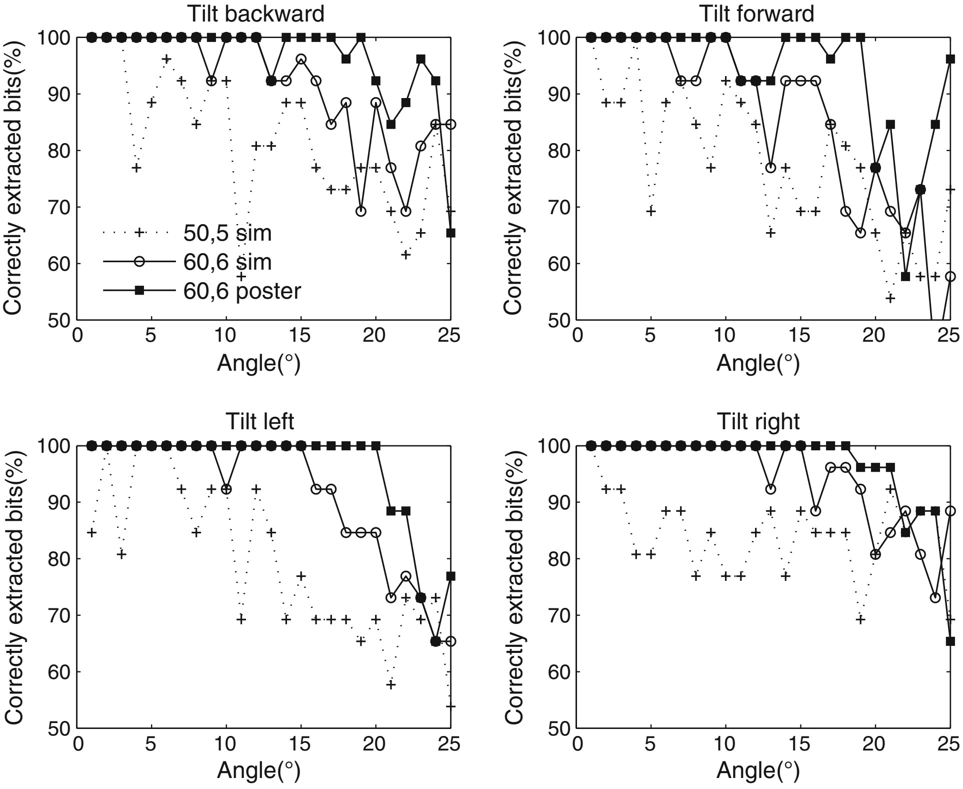


Fig. 14 The robustness to tilting for image 4

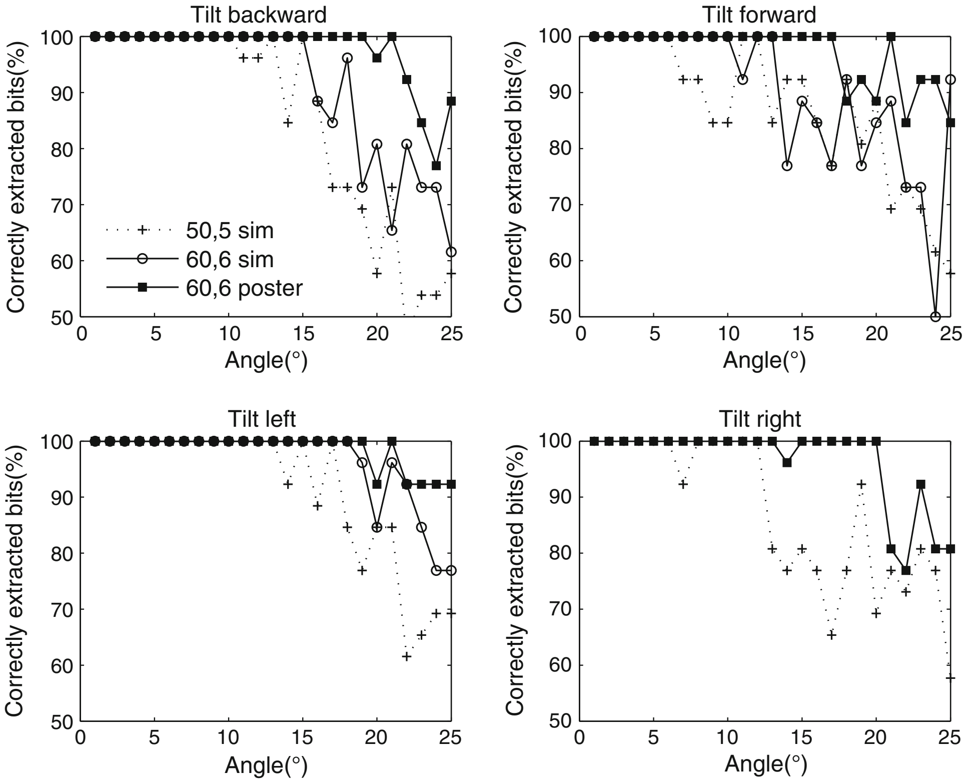
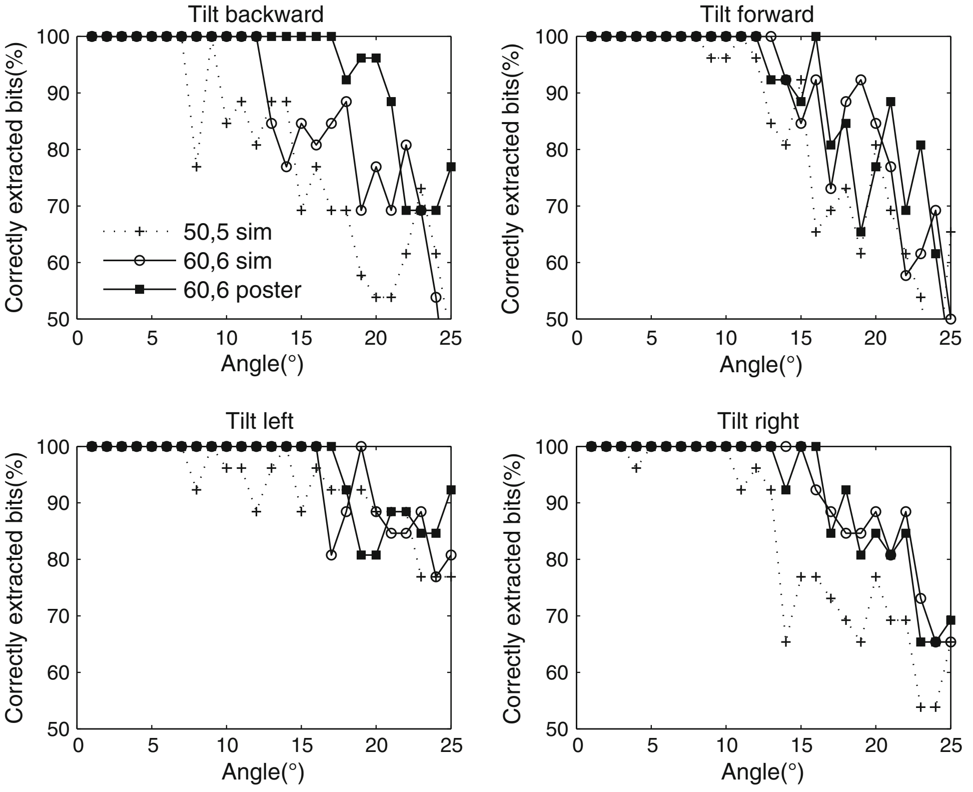


Fig. 15 The robustness to tilting for image 5



References

1. Lin, C.Y., Chang, S.F.: Distortion modeling and invariant extraction for digital image print-and-scan process. In: International Symposium on Multimedia Information Processing, December, Taiwan (1999)
2. He, D., Sun, Q.: A practical print-scan resilient watermarking scheme. IEEE Int. Conf. Image Process. **1**, I-257-60 (2005)
3. Solanki, K., Madhow, U., Manjunath, B.S., Chandrasekaran, S.: Estimating and undoing rotation for print-scan resilient data hiding. IEEE Int. Conf. Image Process. **1**, 39–42 (2004)
4. Pereira, S., Pun, T.: Robust template matching for affine resistant image watermarks. IEEE Trans. Image Process. **9**(6), 1123–1129 (2000)
5. Kutter, M.: Watermarking resisting to translation, rotation and scaling. In: Proceedings of SPIE Multimedia Systems and Applications, Boston, MA, USA, vol. **3528**, pp. 423–431 (1998)
6. Deguillaume, F., Voloshynovskiy, S., Pun, T.: Method for the estimation and recovering from general affine transforms. In: Proceedings of SPIE, Electronic Imaging 2002, Security and Watermarking of Multimedia Contents IV, vol. **4675**, pp. 313–322 (2002)
7. O'Ruanaidh, J.J.K., Pun, T.: Rotation, scale and translation invariant digital image watermarking. In: IEEE Proc. of Int. Conf. on Image Processing, Santa Barbara, CA, USA, vol. **1**, pp. 536–539 (2007)
8. Bas, P., Chassery, J.M., Macq, B.: Geometrically invariant watermarking using feature points. IEEE Trans. Image Process. **11**(9), 1014–1028 (2002)
9. Perry, B., MacIntosh, B., Cushman, D.: Digimarc MediaBridge—the birth of a consumer product, from concept to commercial application. In: Proceedings of SPIE Security and Watermarking of Multimedia Contents IV, San Jose, CA, USA, vol. **4675**, pp. 118–123 (2002)
10. Stach, J., Brundage, T.J., Hannigan, B.T., Bradley, B.A., Kirk, T., Brunk, H.: On the use of web cameras for watermark detection. In: Proceedings of SPIE Security and Watermarking of Multimedia Contents IV, San Jose, CA, USA, CA, vol. **4675**, pp. 611–620 (2002)
11. Nakamura, T., Katayama, A., Yamamuro, M., Sonehara, N.: Fast watermark detection scheme for camera-equipped cellular phone. In: Proceedings of the 3rd International Conference on Mobile and Ubiquitous Multimedia, College Park, Maryland, USA, October ACM, vol. **83**, pp. 101–108 (2004)
12. Katayama, A., Nakamura, T., Yamamuro, M., Sonehara, N.:New highspeed frame detection method: side trace algorithm (STA) for i-appli on cellular phones to detect watermarks. In: Proceedings of the 3rd International Conference on Mobile and Ubiquitous Multimedia, College Park, Maryland, USA, vol. **83**, pp. 109–116 (2004)
13. Kim, W.-G., Seon, H.L., Seo, Y.-S.: Image fingerprinting scheme for print-and-capture model. In: Advances in Multimedia Information Processing-PCM 2006, LNCS, vol. **4261**, pp. 106–113 (2006)
14. Chou, D.-H., Li, Y.-C.: A perceptually tuned subband image coder based on the measure of just-noticeable-distortion profile. IEEE Trans. Circuits Syst. Video Technol. ACM **5**(6), 467–476 (1995)