

# Iterative Filtering for Semi-Fragile Self-Recovery

Paweł Korus, Jarosław Białas and Andrzej Dziech

Department of Telecommunications, AGH University of Science and Technology, Kraków, Poland  
e-mail: pkorus@agh.edu.pl

**Abstract**—In this paper we propose a novel scheme for semi-fragile self-recovery based on iterative filtering of randomly sampled image sections. The scheme exhibits very good robustness against both malicious tampering and lossy JPEG compression with only slight deterioration of the reconstruction quality with attack strength. We describe the operation of the proposed scheme and present the results of its experimental evaluation. We also compare our approach with two state-of-the-art alternatives described in the literature.

**Index Terms**—content authentication; content reconstruction; self-embedding; self-recovery; digital watermarking; inpainting

## I. INTRODUCTION

Self-recovery, also known as self-embedding, is an active image protection technique allowing for the reconstruction of maliciously tampered image content. It exploits an auxiliary *reconstruction reference*, embedded in the image by means of imperceptible digital watermarking [1]. Based on the embedded information, the receiving entity is able to verify the integrity of individual image fragments, and restore an approximate version of the original content.

The necessity to communicate the original appearance of the image to a decoder incurs a high requirement towards embedding capacity. As a result, most of existing schemes do not tolerate global image processing operations, including content preserving ones like lossy compression. Such schemes, referred to as fragile, can deliver accurate tampering identification, high reconstruction quality [2–4], and successful recovery from even extensive modifications [4–6].

However, in many practical applications tolerance for content preserving operations is of principal importance. Robustness against typical correction operations and compression methods is a fundamental design objective for semi-fragile schemes. In contrast to fragile schemes operating with embedding capacity up to 3 bits per pixel (bpp), their semi-fragile counterparts not only can afford to embed payloads several dozen times smaller, but also need to deal with prospective watermark extraction errors.

Reliable authentication and reconstruction requires compact and robust content representation with considerable amount of redundancy. Consequently, semi-fragile schemes can recover only a coarse approximation of the original content. The reconstruction quality oscillates around 25 dB in terms of peak signal to noise ratio (PSNR). When the protected image is subjected to subsequent processing or malicious tampering, reconstruction artifacts emerge and quickly render the restored content indiscernible. As a result, most of existing schemes tolerate only slight compression and small areas affected by malicious tampering. Specific bounds are usually not reported.

In this paper, we aim to adopt the flexible reconstruction paradigm, originally proposed for fragile self-recovery [7, 8]. The principal idea behind flexible reconstruction is to provide high reconstruction quality when malicious tampering is insignificant, and gradually lower the quality as the tampering rate increases. Hence, the quality loss results from the diminishing amount of recovered content details, and not from reconstruction artifacts.

To this end, we propose a novel semi-fragile self-recovery scheme based on iterative filtering. The reference information is constructed by sampling pixels' intensities along random sections of the image. The number of sections that can be correctly extracted from the image will determine the achievable reconstruction quality. Unknown pixels will be inpainted with using iterative filtering [9]. We will show that such an approach successfully achieves reconstruction flexibility with stable image quality. Based on exhaustive experimental evaluation on a test set of natural images, we will demonstrate superior robustness of our scheme against lossy JPEG compression and malicious tampering. The reconstruction quality remains stable even when 50% of the image area becomes tampered and the image is compressed with JPEG of quality 50.

## II. RELATED WORK

In this section we introduce four representative approaches to semi-fragile self-recovery [10–13]. Performance of two recent schemes is evaluated in detail in Section IV-D.

Lin and Chang directly embed a sub-sampled gray-scale version of the image heavily compressed with JPEG [10]. The reconstruction is possible if the tampering affects only a small area, but also in the presence of content preserving operations, e.g., JPEG compression, Gaussian filtering and contrast or brightness changes. The authors do not report any specific operation bounds or any quantitative measurements of image distortion in the restored areas. The typical embedding distortion varies between 33 dB and 42 dB depending on the embedding strength.

Self-recovery can also be modeled as an irregular sampling problem with restoration by iterative projections onto convex sets [11]. The reference information is obtained by exclusive disjunction on cosine transform coefficients' polarity and pseudo-random bit sequences. The watermark is embedded by modulating middle frequency components of the pinned sine transform. The scheme is robust against JPEG compression, Gaussian filtering, unsharping, and contrast changes. The main limitations are the low reconstruction fidelity, and low resistance to tampering. With sufficiently small tampered areas

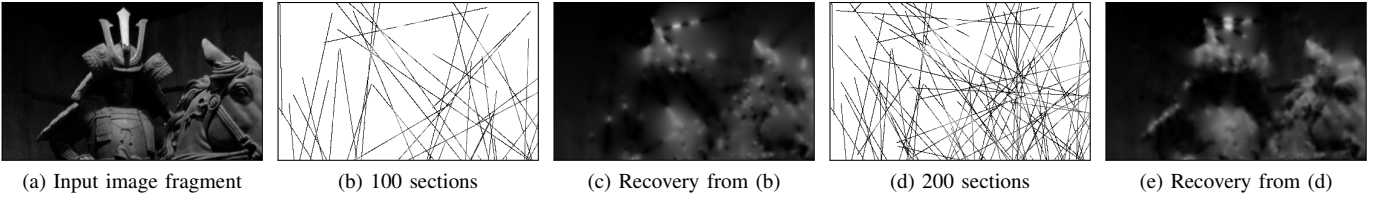


Fig. 1: Content recovery from randomly chosen sections of pixels' intensities; unknown values are marked as white.

and absence of global attacks, the reconstruction quality can reach 27 dB, but typically varies between 12 dB and 24 dB. The typical embedding distortion is approx. 35 dB.

Another approach is to construct the reconstruction reference as a traditional binary watermark by half-toning a sub-sampled grayscale version of the original image [12]. The reconstruction is performed using inverse half-toning, e.g., by Gaussian filtering. The watermark is embedded by custom quantization of the coefficients in the LL1 sub-band of a discrete wavelet transform (DWT) spectrum. No authentication mechanism is used and tampering is to be detected by visual comparison of the restored and the watermarked images. Emerging watermark extraction errors introduce noise into the restoration result, which gradually becomes indiscernible. With error-free extraction of the watermark, the scheme delivers reconstruction quality around 27 dB. Our implementation uses quantization index modulation (QIM) [14] for watermark embedding, which gives better results and allows for better control over the embedding distortion.

Wang et al. use linear regression to predict first 4 discrete cosine transform (DCT) coefficients of the tampered blocks from the embedded mean image block intensities [13]. These intensities are embedded in the image by replacing selected low and medium frequency coefficients in the image's spectrum. The reported reconstruction quality reaches approx. 25 dB, but drops very quickly with increasing JPEG compression strength. The scheme is susceptible to reconstruction dependencies as it simply shifts the reference information between randomly chosen blocks. Hence, it tolerates only limited tampering but no specific bound is reported. In order to mitigate this problem, we supplemented the scheme with inpainting to complete emerging holes in the restored content.

### III. OPERATION OF THE PROPOSED SCHEME

The goal of the proposed scheme is to allow for approximate reconstruction of the original content of protected digital images despite malicious tampering and lossy compression. The scheme involves two entities: an encoder and a decoder. The encoder is responsible for preparing protected images prior to their storage or distribution. The decoder is responsible for image authentication upon its retrieval.

The principal idea behind the proposed approach is to construct the reference information by sampling pixels' intensities along randomly chosen lines on the image plane. We refer to these intensity profiles as *reference sections*. The decoder extracts as many valid sections as possible, restores the pixels along their lines and approximates unknown pixels with the

use of inpainting. Example recovery results from 100 and 200 reference sections are shown in Fig. 1. The more valid sections, the higher the reconstruction quality.

#### A. Operation of the Encoder

Operation of the encoder is shown in Fig. 2. The input image is divided into non-overlapping macro-blocks, and the following steps are repeated for each of them.

1) *Draw line coordinates*: The first step is to draw four line coordinates  $(x_1, y_1, x_2, y_2)$  from a uniform distribution spanning the dimensions of the image. The coordinates are regarded as side-information shared by the encoder and the decoder. In practical implementation, a pseudo-random number generator is initialized by a shared secret key  $\kappa$  to guarantee proper synchronization.

2) *Sample intensities*: The next step is to sample pixels' intensities along the chosen line and transform the sampled signal using 1-dimensional DCT. In this study, we use a constant number of 1000 samples per section.

3) *Generate section descriptor*: The next step is to generate a compact representation of the reference section by retaining a small number of DCT coefficients. The number and the precision of included coefficients is chosen automatically to minimize the distortion within a given budget of available payload bits. The DC coefficient is represented with a uniform 6-bit code-book and the AC coefficients with a family of optimal Lloyd-Max code-books [15] with precision of 4-6 bits per coefficient. The precisions for individual coefficients are determined by solving the following optimization problem:

$$\min \theta(\mathbf{v}), \quad (1a)$$

$$\text{s.t. } \forall_{i \in \{2, \dots, 1000\}} 4 \leq v_i \leq 6, \quad (1b)$$

$$\forall_{i \in \{2, \dots, 1000\}} v_i \in \mathbf{Z}, \quad (1c)$$

$$v_1 = 6, \quad (1d)$$

$$\sum_{i \in \{1, \dots, 1000\}} v_i = b. \quad (1e)$$

where  $b$  is the total number of bits,  $\mathbf{v}$  is a coefficient precision vector, and  $\theta(\mathbf{v})$  is the corresponding distortion of the coefficients. In this study, we used the expected L2 distance between the quantized and the original spectrum coefficients based on a training set of 8,000 sections. Successive improvements of the distortion for individual coefficients are monotonically decreasing with their precision. As a result, the optimization problem in (1) can be efficiently solved with a simple algorithm given below.

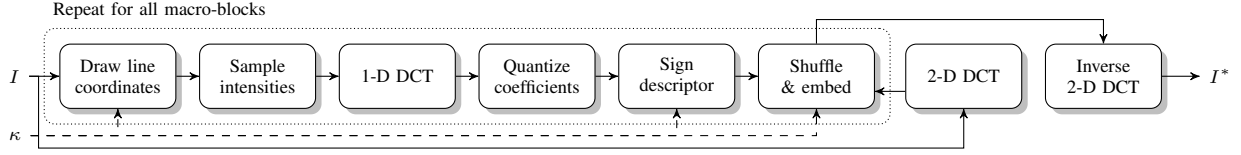


Fig. 2: Operation of the considered encoder:  $I$  - input image,  $\kappa$  - secret key,  $I^*$  - protected image.

The procedure starts with the following initial vector  $\mathbf{v}^{(0)}$ :

$$\mathbf{v}^{(0)} : v_i^{(0)} \in \begin{cases} 6 & \text{for } i = 1, \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

In each step  $k$ , the algorithm improves the precision for a single coefficient whose promotion maximizes the improvement of the objective function (per single allocated bit):

$$\arg\max_i \begin{cases} \frac{1}{4} (\theta(\mathbf{v}^{(k-1)}) - \theta(\mathbf{v}^{(k)})) & \text{for } v_i^{(k-1)} = 0, \\ \theta(\mathbf{v}^{(k-1)}) - \theta(\mathbf{v}^{(k)}) & \text{otherwise.} \end{cases} \quad (3)$$

The algorithm terminates when  $\sum_{i=1}^{1000} v_i = b$ . Coefficients whose promotion would exceed  $b$ , either directly in the current step or indirectly in the following ones, are not considered.

4) *Sign Section Descriptor*: The final step of section descriptor preparation is to append a short signature for validation by the decoder. Due to negligible impact of prospective errors on certain positions, the signature is calculated only from the most important bits (MSB) of the coefficients; we consider 4 MSBs for 6-bit coefficients and 3 MSBs for lower precisions. The secret key  $\kappa$  and macro-block index are also included in the signature. We use 8-bit signatures obtained by shortening MD5 hashes.

5) *Shuffle & Embed*: The final step is to embed the obtained watermark. The embedding is performed on selected coefficients  $\{x_i\}$  of a block-wise calculated 2-dimensional DCT spectrum. The block size for DCT calculation is chosen to match JPEG, i.e.,  $8 \times 8$  px. The coefficients are modified using quantization index modulation:

$$x_i^* = \text{sign}(x_i) \cdot \Delta \cdot Q_{m_i}(|x_i| \cdot \Delta^{-1}) \quad (4)$$

where  $x_i$  is the original spectrum coefficient,  $x_i^*$  is the watermarked coefficient,  $\Delta$  is the quantization step. The  $\text{sign}(x) = x|x|^{-1}$  function returns the sign of the coefficient, and  $Q_{m_i}(\cdot)$  is a quantizer for message bit  $m_i$ , i.e.,  $Q_0(x) = 2 \cdot \lfloor \frac{x}{2} + 0.5 \rfloor$  and  $Q_1(x) = 2 \cdot \lfloor \frac{x}{2} \rfloor + 1$ .

The descriptor is scrambled prior to embedding. The most important bits and the least important bits are shuffled separately, so that the most important bits would be embedded in lower frequencies characterized by better robustness against incidental distortion.

## B. Operation of the Decoder

Operation of the decoder is shown in Fig. 2. The following steps are used to recover the coarse version of the original content. Steps 2 - 6 are repeated for all macro-blocks; the remaining ones are performed only once.

1) *Recover the Quantization Step*: The first step is to estimate the quantization step  $\Delta$  used by the encoder. We use a linear search in a range of feasible values to find the one maximizing the number of correctly extractable sections.

2) *Extract the watermark*: The next step is to extract the embedded watermark using:

$$m_i = \lfloor 0.5 + |x_i^*| \cdot \Delta^{-1} \rfloor \bmod 2 \quad (5)$$

where  $x_i^*$  denotes the watermarked coefficient, and  $m_i$  a single bit of the watermark. The extracted payload is then descrambled to obtain valid sections' descriptors.

3) *Draw line coordinates*: In order to maintain synchronization with the coordinates chosen by the encoder, the decoder draws  $(x_1, y_1, x_2, y_2)$  from a pseudo-random number generator initialized by the secret key  $\kappa$ .

4) *Validate signature*: In the next step the decoder recalculates the signatures of the extracted section descriptors and compares them with their extracted counterparts. If the signatures do not match, the current section is discarded.

5) *Reconstruct and validate sections' spectra*: Validated section descriptors are parsed and their spectrum coefficients are read from the corresponding code-books. Due to limited length of the embedded signatures, occasional false negative errors are to be expected as a result of signature verification. In order to improve the classification accuracy, the decoder analyzes the spectra of the previously accepted sections. A support vector machine (SVM) is used to distinguish between valid and anomalous sections. In this study, we used a linear kernel to train a classifier on four features extracted from the sections' spectra: the variance of the coefficients' magnitudes, parameters of a linear regression fit to the AC coefficients, and the maximum deviation of the AC coefficients from the linear fit. The penalty for false positive errors was 10-fold the penalty for false negative errors. The training was performed on a set of 30 representative natural images (Section IV). We verified the accuracy using 4-fold cross validation, and obtained the average true positive rate of 99.96% and the average true negative rate of 91.72%.

6) *Inverse 1-D DCT*: In the final step of reference section descriptors' extraction, the decoder performs inverse 1-dimensional DCT to recover their spatial representation.

7) *Fuse reference sections*: Once the reference sections are validated, the next step is to restore known pixels. Since the lines are likely to overlap, a single pixel can be estimated from multiple values originating from different sections. While many strategies could be adopted to fuse candidate values, it is important to consider shorter sections with greater weights, due to their relatively better representation accuracy. In our

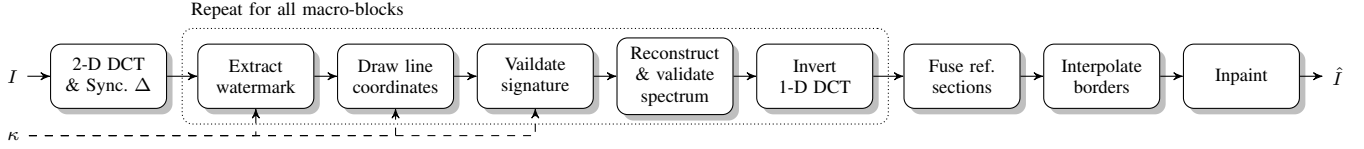


Fig. 3: Operation of the considered decoder:  $I$  - input image,  $\kappa$  - secret key,  $\hat{I}$  - restored original image.

experiments, we observed no visible differences between choosing the best candidate (from the shortest section) and weighting the candidates inversely proportional to their sections' lengths. The presented results were obtained with the former strategy.

8) *Interpolate borders*: Randomly chosen lines will rarely provide enough information about the content on the borders which leads to irregular black frame around the image. In order to mitigate this issue, the content of the borders is predicted from the closest known pixels. On each border 64 pixels are estimated this way; the remaining ones are estimated using B-Spline interpolation.

9) *Inpaint unknown pixels*: The last step is to predict the values of the unknown pixels with the use of inpainting. We use a simple and fast approximation algorithm for iterative filtering [9] inspired by the push and pull approach [16]. The algorithm performs iterative down-sampling and up-sampling using a local kernel at multiple resolutions. In this study we used 5 decomposition levels and a 2-D Gaussian-like kernel  $\mathbf{k}^T \mathbf{k}$  with  $\mathbf{k} = [0.05 \ 0.25 \ 0.40 \ 0.25 \ 0.05]$ .

#### IV. EXPERIMENTAL EVALUATION AND ANALYSIS

The behavior of the proposed scheme can be controlled with a number of parameters. The reconstruction quality is controlled by the number of reference sections and by the precision of their representation. The embedding distortion and robustness are controlled by the quantization step  $\Delta$  and the selection channel. In the performed experiments, we use the coefficients 7-14 in the zig-zag order from each  $8 \times 8$  px block which results in a fixed embedding capacity of 0.125 bpp. Hence, each macro-block carries a 128-bit descriptor of a reference section, including a 8-bit signature.

Due to insufficient image resolution in commonly used test sets, we chose 100 photographs with diverse content spanning various brightness levels and the number of details. The images (grayscale 1 Mpx) can be obtained at [17].

##### A. Embedding Distortion Assessment

In this experiment we change the quantization step  $\Delta$  and measure the PSNR between the original and the protected images. The measurement is performed for  $\Delta$  starting from 0.035 up to 0.085 with a 0.005 step. Each test image is evaluated 10 times, with different security keys  $\kappa$ . The obtained results are shown in Fig. 4. The plot shows the average distortion for each quantization step along with prediction intervals for new observations with 95% confidence.

Fig. 5 shows an example fragment of a protected image for increasing embedding strength. The watermark is visible

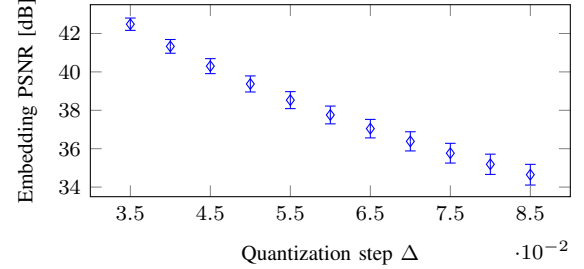


Fig. 4: Embedding distortion for various embedding strengths; error bars represent prediction intervals with 95% confidence.

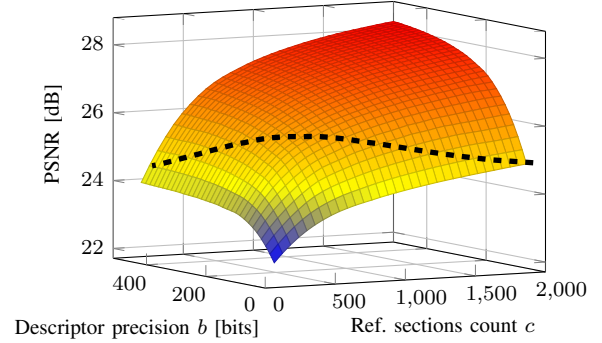


Fig. 6: Average reconstruction quality for various numbers and precision of reference sections; the dashed line shows the embedding capacity constraint for 1 Mpx images.

as a noise-like pattern uniformly distributed over the entire image. The noise is well masked in textured areas, but becomes perceptible in low-texture areas. This limitation could be addressed by modulating the embedding strength accordingly; the issue is, however, out of scope of this study.

##### B. Reconstruction Quality Assessment

Let  $c$  denote the number of reference sections, and  $b$  the precision of their descriptors. Fig. 6 shows a surface fitted to the average reconstruction quality (3 repetitions for 100 images) for various configurations of  $(b, c)$ . The fit was obtained using locally weighted smoothing quadratic regression ( $R^2 = 0.995$ ). Embedding capacity constraint for 1 Mpx images is shown with a dashed curve. The optimal parameters correspond to a conditional maximum on the curve. We numerically obtained  $(b = 174, c = 753)$  which gives the PSNR of 25.7 dB.

The obtained results clearly show that it is pointless to increase the precision  $b$  until a sufficient number of sections

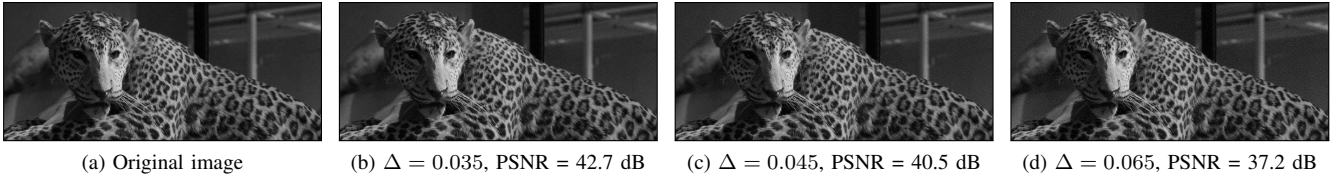


Fig. 5: Example fragments of a protected image for increasing embedding strength.

can be used for reconstruction. The choice also needs to take into account that both malicious tampering and incidental processing will reduce the number of useful sections. Considering this, we chose the precision of  $b = 120$  bits, which gives the expected PSNR only 0.3 dB worse than the optimum. For higher-resolution images other values might be preferred.

The reconstruction quality is expected to fluctuate based on the selection of reference sections on the image plane. For the selected configuration of  $b = 120$  bits, we performed 30,000 repetitions (300 seeds for 100 images) of the protection process. 99% of the obtained PSNR scores line within a  $\pm 0.38$  dB range around the mean value for individual images. Greater improvement can be expected by carefully designing a strategy for optimal selection of the reference sections. With proper control over the distribution of drawn lines' lengths, we managed to obtain visibly better representation of finer image details. However, the issue is out of scope of this study.

### C. Impact of Malicious Tampering

The number of reference sections available for reconstruction is proportional to the number of authentic image blocks. Sections extracted from tampered image blocks will ideally be removed during descriptor validation. The goal of this experiment is to assess the effective false negative classification rate. For this purpose, we perform watermark extraction and reconstruction attempts for unwatermarked images compressed with JPEG of various quality levels. The obtained effective false negative rate is  $3.799 \pm 0.66 \cdot 10^{-5}$  95% confidence (Wilson confidence interval). Signature verification alone leads to false positive rate of  $3.9 \cdot 10^{-3}$ , which means that adoption of SVM classifier reduced the error rate approximately 100-fold.

### D. Comparison with Existing Schemes

In this experiment we compared the performance of the proposed scheme with two state-of-the-art alternatives: the scheme by Cheddad et al. [12] based on inverse halftoning of a binary reference image, and the scheme by Wang et al. [13] based on prediction of low-frequency DCT coefficients from the embedded mean block intensities.

Our comparison focused on the reconstruction quality and robustness. We performed a full protection  $\rightarrow$  tampering  $\rightarrow$  reconstruction cycle for various combinations of compression levels and tampering rates; each configuration was repeated 10 times for different security keys for each of the 100 images. The tampering was implemented as content replacement with unwatermarked image data in randomly chosen image blocks.

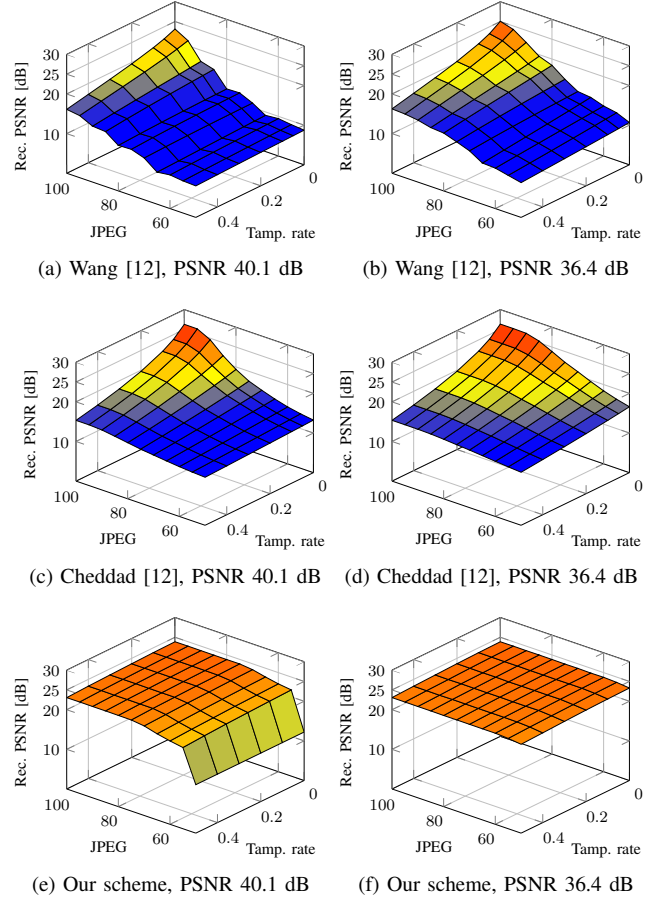


Fig. 7: Average reconstruction quality for tampered and compressed images; embedding strengths 40.1 dB and 36.4 dB.

We considered four embedding strengths, corresponding to average embedding distortion of 42.5 dB, 40.1 dB, 38.5 dB and 36.4 dB. We controlled the embedding strength by adjusting normalization factors of the mean pixel values [13] and by adjusting the quantization step for the remaining schemes.

Fig. 7 shows the average quality of the restored images. The best quality (PSNR of 26.3 dB) was observed for the scheme by Cheddad et al. [12]. However, it quickly deteriorates with increasing tampering and compression rates. The scheme by Wang et al. [13] reveals similar behavior yet with higher susceptibility to lossy compression. Despite worse representation of fine image details, the proposed scheme delivers superior robustness against both malicious tampering and JPEG compression. The obtained PSNR scores remain stable



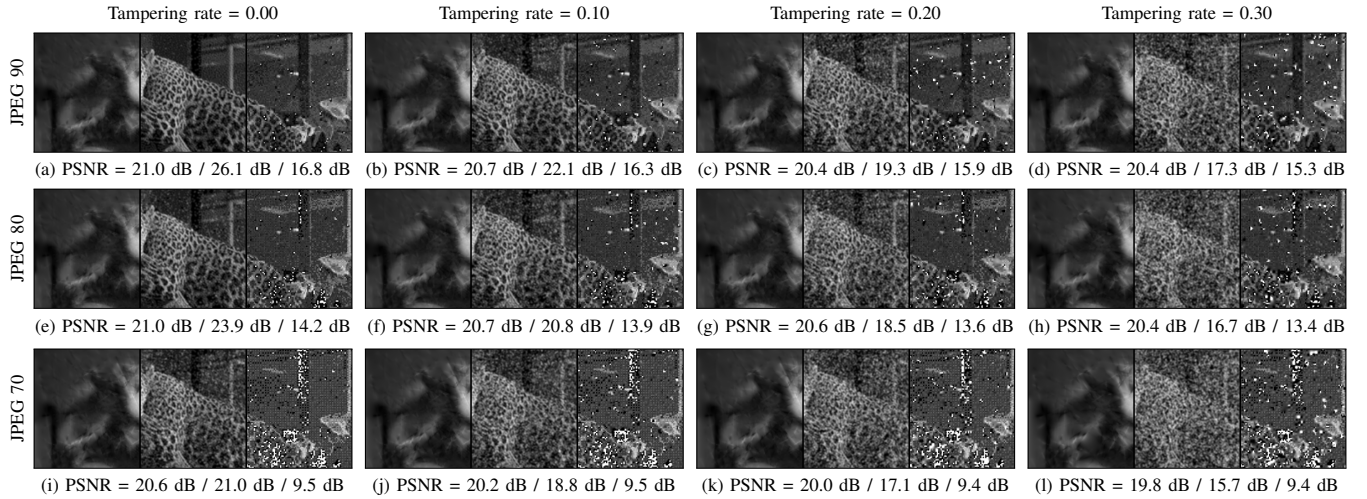


Fig. 8: Comparison of restored images in presence of both compression and tampering; embedding strength 36 dB; from left: proposed scheme, Cheddad's scheme [12], Wang's scheme [13]; for clarity, only top half of the image is shown.

for a very wide range of attack strengths. The observed quality deterioration for lower embedding strength and increasing compression rates stems from a rapid decline of watermark extraction performance for some of the test images.

Fig. 8 shows example reconstruction results for a grid of various tampering and compression rates. The figures allow to see gradual degradation of the reconstruction fidelity. White artifacts in the scheme by Wang et al. [13] stem from overflowing mean block intensities, most frequent in dark image fragments and for low embedding strengths. The illustrated example corresponds to the embedding PSNR of 36.4 dB. A complete set of results, including wider ranges of the grid and all test images is available as supplementary materials [17].

## V. CONCLUSIONS

While the adopted approach is not well suited for the reconstruction of fine image details, it exhibits remarkable resiliency against both malicious tampering and lossy compression. We verified that the reconstruction quality remains nearly unaffected even when 50% of the image area becomes tampered, and the image is compressed to a JPEG of quality 50 (Fig. 7). We believe our scheme provides an interesting alternative, especially for applications where coarse reconstruction is sufficient or high robustness is required.

## ACKNOWLEDGMENT

The research leading to these results received funding from the European Regional Development Fund under INSIGMA project no. POIG.01.01.02-00-062/09. We would like to thank Hui Wang, Anthony Ho, and Xi Zhao for providing source code of their scheme.

## REFERENCES

- [1] J. Fridrich and M. Goljan, "Images with self-correcting capabilities," in *Proc. of IEEE International Conference on Image Processing*, 1999.
- [2] X. Zhang, S. Wang, Z. Qian, and G. Feng, "Reference sharing mechanism for watermark self-embedding," *IEEE Trans. Image Process.*, vol. 20, no. 2, pp. 485–495, 2011.
- [3] X. Zhang and S. Wang, "Fragile watermarking with error free restoration capability," *IEEE Trans. Multimedia*, vol. 10, no. 8, 2008.
- [4] P. Korus and A. Dziech, "Efficient method for content reconstruction with self-embedding," *IEEE Trans. Image Process.*, vol. 22, no. 3, pp. 1134–1147, March 2013.
- [5] X. Zhang, S. Wang, and G. Feng, "Fragile watermarking scheme with extensive content restoration capability," in *Proc. of International Workshop on Digital Watermarking*, 2009.
- [6] P. Korus and A. Dziech, "Adaptive self-embedding scheme with controlled reconstruction performance," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 2, pp. 169–181, Feb 2014.
- [7] X. Zhang, Z. Qian, Y. Ren, and G. Feng, "Watermarking with flexible self-recovery quality based on compressive sensing and composite reconstruction," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 4, pp. 1223–1232, 2011.
- [8] X. Zhang, S. Wang, Z. Qian, and G. Feng, "Self-embedding watermark with flexible restoration quality," *Multimedia Tools and Applications*, vol. 54, pp. 385–395, 2011.
- [9] I. Drori, D. Cohen-Or, and H. Yeshurun, "Fragment-based image completion," in *ACM T Graphic*, 2003, vol. 22, pp. 303–312.
- [10] C-Y. Lin and S-F. Chang, "Sari: self-authentication-and-recovery image watermarking system," in *ACM Int. Conf. on Multimedia*, 2001.
- [11] X. Zhu, A. T. S. Ho, and P. Marziliano, "A new semi fragile image watermarking with robust tampering restoration using irregular sampling," *Signal Processing : Image Communication*, vol. 22, no. 5, 2007.
- [12] A. Cheddad, A. Condell, K. Curran, and P. Mc Kevitt, "A secure and improved self-embedding algorithm to combat digital document forgery," *Signal Processing*, vol. 89, pp. 2324–2332, 2009.
- [13] H. Wang, A. T. S. Ho, and X. Zhao, "A novel fast self-restoration semi-fragile watermarking algorithm for image content authentication resistant to jpeg compression," in *Proc. 10th Int. Conf. on Digital-Forensics and Watermarking*, 2012.
- [14] B. Chen and G. W. Wornell, "Quantization index modulation: a class of provably good methods for digital watermarking and information embedding," *IEEE Transactions on Information Theory*, vol. 47, no. 4, pp. 1423–1443, 2001.
- [15] S. Lloyd, "Least Squares Quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129 – 137, mar 1982.
- [16] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. ACM, 1996, pp. 43–54.
- [17] "Supplementary materials," <http://kt.agh.edu.pl/~korus>.