WATERMARK-PRESERVING KEYPOINT ENHANCEMENT FOR SCREEN-SHOOTING RESILIENT WATERMARKING

Li Dong[*], *Jiale Chen*[*], *Chengbin Peng*[*], *Yuanman Li*[†], *and Weiwei Sun*[‡]

[*]Ningbo University, China, dongli@nbu.edu.cn, chenoly@outlook.com and pchbin@gmail.com;
[†]Shenzhen University, China, yuanmanli@szu.edu.cn;
[‡]Alibaba Group, China, sunweiwei.sww@alibaba-inc.com.

ABSTRACT

Screen-shooting resilient (SSR) watermark is a special kind of robust watermarking. One can extract the watermark message even the embedded image communicates via a physical screen to the camera channel. The keypoint-based SSR watermarking is one promising solution to realize such screen-to-camera communication. The enhanced keypoints were used to locate the embedding region and then perform watermark embedding. However, the keypoint-based SSR watermarking treats the critical two steps, keypoint enhancement and watermark embedding, independently, neglecting their interplay. This work proposes a watermark-preserving keypoint enhancement algorithm for SSR watermarking. Specifically, we resort to a convex constrained optimization framework to unify keypoint enhancement and watermark embedding. Multiple constraints are imposed to simultaneously ensure the watermark validity and blind synchronization of embedding regions. Our method enables jointly optimizing the watermarking distortion and keypoint enhancement. The proposed method achieves superior watermark extraction accuracy while retaining better watermarked image quality when compared with previous works.

Index Terms— Robust watermarking, screen-to-camera communication, image keypoint enhancement.

1. INTRODUCTION

Screen-shooting resilient (SSR) watermarking aims at embedding watermark data such as hyperlinks into a host image. Afterward, the end-user uses a camera-mounted device to capture the displayed embedded image from the screen, and then correctly extract the watermark. With the widespread use of smartphone and display devices, such cross-media communication is recently becoming a favorable solution to enabling many applications, *e.g.*, copyright protection, confidential image leak tracing, and display information retrieving.

Essentially, SSR watermark is a robust watermark that could survive after various severe distortions, *e.g.*, lens distortion, light source distortion, and screen-shooting Moiré distortion, to name a few. In general, the existing works can be categorized into two main types: image coding-based, and keypoint-based. Many early works mainly fall into the image coding-based category. Typically, these methods attempted to embed watermarks using specially-designed patterns. Nakamura *et al.* [1] devised 2D rotationally orthogonal sinusoidal patterns to encode watermark data, and then adaptively superpose the pattern over the host image. By taking advantage of the imperceptibility to small luminance changes for human eyes, Gugelmann *et al.* [2] designed a symbol-shaped watermark complemented with an adapted convolutional coding system, which could resist screen-shooting distortion and certain image manipulations. Pramila *et al.* [3] suggested encoding the watermark with a directed periodic pattern, and then devised a subtractive-additive embedding algorithm. The watermark can be detected by searching the regularities of a periodic signal. Jia *et al.* [4] proposed an end-to-end neural network, in which an encoder is used to hide the watermark message and a decoder is for watermark extraction.

Instead of encoding watermark over the entire image like image coding-based schemes, keypoint-based methods often work on several carefully selected regions by keypoints. Such image keypoints for locating the watermark embedding regions shall be robust to screen-shooting distortion, and survive at the extractor side, to achieve watermark synchronization. Fang *et al.* [5] thoroughly analyzed the dominating distortions caused by the screen-shooting process, proposed a modified *scale-invariant feature transform* (SIFT) algorithm [6] to locate the embedding regions for template-based watermarking. To improve the accuracy of keypoint localization, the intensity SIFT keypoints were enhanced before watermark embedding. By employing a neural network-based keypoint detector, Li *et al.* [7] embedded the watermark by combining the feature region filtering model with keypoint detection, and a quaternion discrete Fourier transform and tensor

This work was supported by the Natural Science Foundation of China under Grants 61901237 and 62171244, Alibaba Group through Alibaba Innovative Research Program, Natural Science Foundation of Zhejiang Province (LGG20F020011), and Open Project Program of the State Key Laboratory of CAD&CG, Zhejiang University (A2006). (*Corresponding author: Li Dong*.)

decomposition-based watermark procedure. Chen *et al.* [8] offered a local square feature region construction method, and the *speeded-up robust feature* (SURF) keypoint descriptor were used for realizing watermark synchronization.

However, all the existing keypoint-based methods treat the crucial two steps, keypoint enhancement, and watermark embedding, independently, neglecting the interplay between them. On one hand, the embedded watermark distorts the host image, which would, in turn, deteriorate the keypoint localization and make the watermark extraction fail. On the other hand, the keypoint enhancement probably distorts the host image as well, which may harm the effectiveness of the embedded watermark at the extractor side. To resolve this issue, we in this work propose a watermark-preserving keypoint enhancement algorithm for screen-shooting resilient watermarking. The proposed approach resorts to a convex constrained optimization framework and explicitly considers the two steps keypoint enhancement and watermark embedding in a joint way. Multiple constraints are introduced to simultaneously ensure the watermark validity and blind synchronization of embedding regions. Experimental results demonstrate that, compared to previous works, the proposed method could achieve higher watermark extraction accuracy, while retaining reasonably good embedded image quality.

The rest of this paper is organized as follows. In Section 2, the workflow of a typical keypoint-based SSR watermarking is reviewed, along with a discussion of its limitation. Then, Section 3 presents the proposed watermark-preserving keypoint enhancement algorithm. Experimental results are provided in Section 4 and finally Section 5 concludes the work.

2. KEYPOINT-BASED SCREEN-SHOOTING RESILIENT WATERMARKING

In this section, we first briefly review the work [5], a recent representative keypoint-based screen-shooting resilient watermarking scheme, and then point out its inherent limitation. This motivates us to design a watermark-preserving keypoint enhancement framework. The watermarking procedure of [5] consists of three steps: 1) employ a keypoint algorithm to select the several most robust keypoints; 2) enhance the keypoints for better locating the embedding regions blindly; and 3) embed watermark into these carefully selected regions.

Keypoint Selection: The work [5] uses the Intensity-based Scale-Invariant Feature Transform (I-SIFT) [6] for locating the keypoints[1]. Specifically, the given image $\mathbf{I}$ of size $M \times N$ is first convoluted with a series of Gaussian filters,

$$L_{\mathbf{I}}\left(x, y, \sigma_s\right) = \mathbf{I}(x, y) \otimes G\left(x, y, \sigma_s\right), \tag{1}$$

where $L_{\mathbf{I}}$ is the resultant Gaussian blurred image, (x, y) denotes the image coordinate, and G is a 2D Gaussian filter parameterized by σ_s at the s-th scale. Then, Gaussian blurred image $L_{\mathbf{I}}$ is downsampled and repeated another round of Gaussian filtering. This finally forms multiple groups of Gaussian blurred images, *i.e.,* Gaussian blurred image pyramid. Each group of Gaussian blurred images is named as an octave. To identify the keypoint candidates, two consecutive scale images in an octave are subtracted to produce the Difference of the Gaussian (DoG) pyramid,

$$D_{\mathbf{I}}\left(x, y, \sigma_s\right) = L_{\mathbf{I}}\left(x, y, \sigma_{s+1}\right) - L_{\mathbf{I}}\left(x, y, \sigma_s\right). \tag{2}$$

where $D_{\mathbf{I}}$ is the DoG image of the s-th scale. The extreme points of the DoG space are selected as the candidate keypoints. Let $\mathbf{p} \triangleq (x, y, \sigma_s)$ be a point (location) in a DoG image, then we have $D_{\mathbf{I}}(\mathbf{p}) = D_{\mathbf{I}}(x, y, \sigma_s)$, representing the DoG value of this specific point. For each point, test whether it is the maximum or minimum point, in the $3 \times 3 \times 3$ cube centered by this point. The point will be selected as a candidate keypoint if it is the locally maximum or minimum point. I-SIFT suggests that the absolute intensity of a keypoint DoG value can be used as a keypoint robustness indicator, *i.e.,*

$$R_{\mathbf{I}}(\mathbf{p}) = |D_{\mathbf{I}}(\mathbf{p})|, \tag{3}$$

where $R_{\mathbf{I}}(\mathbf{p})$ evaluates the keypoint intensity of $\mathbf{p}$ in the image $\mathbf{I}$, Clearly, a larger R value indicates higher keypoint robustness. Thus, to embed a watermark into the most robust keypoint regions, all the candidate keypoints are sorted in descending order based on their R values, and the top-k keypoints are selected for embedding.

Keypoint Enhancement: To boost the keypoint robustness and facilitate watermark extraction at the extractor side, before watermark embedding, Fang *et al.* suggested a SIFT intensity enhancement scheme, which could enhance the intensity of keypoints that used for locating embedding regions, and weaken the intensity of the other keypoints that not selected for embedding watermark. Basically, the proposed enhancement method established a constrained optimization framework, by modifying the SIFT keypoint editing algorithm proposed by Li *et al* [9], which is originally devised for SIFT keypoint removal. Due to space limit, we omit the detailed formulation here. More relevant details of the optimization model can be referred to Section IV of the work [5].

Watermarking Embedding: The watermark is embedded into the regions centered around the selected and enhanced keypoints. As many previous works practiced, Fang *et al.* also suggested performing embedding in the transform domain, which could well balance the robustness and transparency of the embedded watermark. The core idea is to encode one watermark bit by modulating the relationship between paired transformed coefficients. To be more specific, for an image patch of size 8×8, the 2D discrete cosine transform (DCT) is first applied, obtaining 8×8 DCT coefficient matrix $\mathbf{C}$. A pair of coefficients c_1, c_2 is selected from low-middle frequency bands, *e.g.,* $c_1 = \mathbf{C}(4, 5)$ and $c_2 = \mathbf{C}(5, 4)$. Then, the embedding process can be written as

[1]I-SIFT primarily works on grayscale images. The color images shall be first converted into YCbCr space, and then apply I-SIFT in the Y channel.

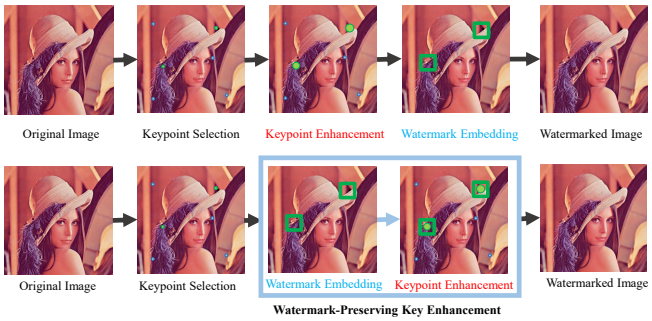


Fig. 1. Comparison of the watermarking workflow between Fang *et al.* [5] (top) and our proposed method (bottom). The key difference is that the proposed method swap the order of keypoint enhancement and watermark embedding, and jointly optimize these two steps via an optimization model.

$$\begin{cases} \hat{c}_1 = \max(c_1, c_2) + \Delta, \hat{c}_2 = \min(c_1, c_2) - \Delta, & \text{if } w = 0 \\ \hat{c}_1 = \min(c_1, c_2) - \Delta, \hat{c}_2 = \max(c_1, c_2) + \Delta, & \text{if } w = 1 \end{cases} \tag{4}$$

where $\hat{c}_1$ and $\hat{c}_2$ are the resultant embedded coefficients. $w \in \{0, 1\}$ is the watermark bit to be embedded, and Δ is the predefined or adaptively chosen embedding strength parameter, aiming to enlarge the differences between $\hat{c}_1$ and $\hat{c}_2$. As can be seen, after embedding, the inequality $\hat{c}_1 \geq \hat{c}_2$ holds when $w = 0$, and $\hat{c}_1 < \hat{c}_2$ holds when $w = 1$. Therefore, at the receiver side, one can blindly extract the watermark bit $\hat{w}$: $\hat{w} = 0$ if $\hat{c}_1 \geq \hat{c}_2$ and $\hat{w} = 1$ otherwise.

It is worth noting that, the correctness of extracted watermark bit is primarily determined by the keypoint localization. This is because, on the extractor side, the embedding regions are located using the keypoints. Inaccurate keypoint localization would severely degrade the watermark extraction accuracy. However, we in experiments found that, for certain images (*e.g.*, simple textured images such document), the keypoint localization would suffer severe performance degradation, and the located embedding regions are drift dramatically compared with the correct ones (refer to Section 4). In fact, this issue was also frankly pointed by Fang *et al.* [5], being an open problem to be resolved. With careful examination of the framework of the work [5] (see Fig. 1), we notice that the keypoints for locating embedding regions are enhanced before watermark embedding, and these two steps are considered independently. The embedding operation certainly distorts the image content, which inevitably distort the previous enhanced keypoint localization step to some extent. To this end, we in this next section propose a watermark-preserving keypoint enhancement algorithm to tackle this issue.

3. PROPOSED WATERMARK-PRESERVING KEYPOINT ENHANCEMENT

As shown in Fig. 1, we suggest swapping the step *watermark embedding* with *keypoint enhancement*, and jointly performing these two steps through a constrained optimization framework. The underlying motivation behind this strategy is as follows. By moving the *watermark embedding* one step forward, one can embed watermarking embedding process into the keypoint enhancement optimization framework, which can jointly control the watermarking distortion, keypoint enhancement distortion in an optimal way, without harming the effectiveness of watermarking and keypoint localization.

More specifically, let $\mathbf{B}_E$ be the original image region for watermark embedding, which is located by the selected keypoints, and let $\widetilde{\mathbf{B}}_E$ denotes its corresponding region after jointly watermark embedding and keypoint enhancement. Clearly, $\widetilde{\mathbf{B}}_E$ is the optimization variable. It is expected that, after keypoint enhancement, the distortion between image region $\widetilde{\mathbf{B}}_E$ and $\mathbf{B}_E$ should be as small as possible. The Frobenius norm is adopted here to evaluate the loss to be minimized, which can be expressed as

$$\arg\min_{\widetilde{\mathbf{B}}_E} \left\| \mathbf{B}_E - \widetilde{\mathbf{B}}_E \right\|_F^2. \tag{5}$$

The constraints for the proposed constrained optimization problem shall consist of the following three constraints sets.

Constraint $\mathcal{C}_1$: Guarantee the validity of the embedded watermark. Recall that, for the watermark embedding procedure, one watermark bit is embedded via modulating the magnitude relationship between two DCT coefficients. However, the following keypoint enhancement may cause the coefficient pair to violate this relationship. Thus, we shall impose constraints on the DCT coefficients of $\widetilde{\mathbf{B}}_E$ to preserve the original magnitude relationship. This can be expressed as

$$\mathcal{C}_1 : \begin{cases} \tilde{c}_1 \geq \hat{c}_1,\ \tilde{c}_2 \leq \hat{c}_2, & \text{if } w = 0. \\ \tilde{c}_1 \leq \hat{c}_1,\ \tilde{c}_2 \geq \hat{c}_2, & \text{if } w = 1. \end{cases}$$

where $\tilde{c}_1$ and $\tilde{c}_2$ are the DCT coefficient pair to be optimized. These two variables can be expressed via DCT transform over $\widetilde{\mathbf{B}}_E$. $\hat{c}_1$ and $\hat{c}_2$ are the DCT coefficients after embedding via (4). One can see that when the watermark bit $w = 0$, the resultant coefficient $\hat{c}_1 \geq \hat{c}_2$. Combined the above constraint $\tilde{c}_1 \geq \hat{c}_1, \hat{c}_2 \geq \tilde{c}_2$, we have $\tilde{c}_1 \geq \hat{c}_1 \geq \hat{c}_2 \geq \tilde{c}_2$, leading to $\tilde{c}_1 \geq \tilde{c}_2$. This maintains the magnitude relationship of the DCT coefficient pair after embedding.

Constraint $\mathcal{C}_2$: Enhance the intensity of the selected keypoints for embedding. As stated in [5], the keypoint selected for locating embedding regions would undergo a series of complex distortions, including keypoint enhancement distortion, watermarking distortion, and the following screen-shooting channel distortions. To ensure the robustness of watermark extraction, it is suggested to enhance the intensity of the selected keypoints. In this light, a constraint for increasing the intensity of the keypoint $\mathbf{p}_k$ can be posed as follows

$$\mathcal{C}_2 : R_{\tilde{\mathbf{I}}}(\mathbf{p}_k) - R_{\mathbf{I}}\left(\mathbf{p}_k^{\max}\right) \geq \xi, \quad \text{if } \mathbf{p}_k \text{ is a extrema},$$

where $\mathbf{p}_k$ is the keypoint to be enhanced, and $\mathbf{p}_k^{\max}$ is the keypoint with the largest keypoint intensity over the entire original image. $\xi > 0$ is a pre-specified intensity increment parameter. $\tilde{\mathbf{I}}$ is the watermarked image, in which the embedding

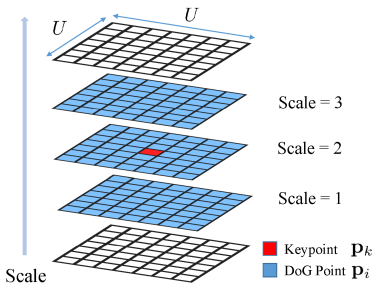


Fig. 2. The scale of keypoints.

region is $\widetilde{\mathbf{B}}_E$, and non-embedding regions remain the same as the original image $\mathbf{I}$. It is worth noting that the intensity of a keypoint, *i.e.*, R value in (3), is non-negative, while DoG value could be negative, and $R_{\tilde{\mathbf{I}}}(\mathbf{p}_k) = |D_{\tilde{\mathbf{I}}}(\mathbf{p}_k)|$. Thus the constraint $\mathcal{C}_2$ equivalent to increasing the DoG value towards positive infinity by at least ξ when the keypoint is a maximum, and decrease the negative DoG value towards negative infinity when the keypoint is a minimum.

Constraint $\mathcal{C}_3$: The intensity of newly generated keypoints shall be smaller than that of the enhanced keypoints. In the work [5], to correctly locate the embedding regions, despite enhancing the intensity of the selected keypoints, one shall ensure the enhancement will not cause new keypoints as well. The work [5] inherits Li *et al.* optimization formulation [9], making a constraint by bounding the DoG values of the embedded local region between local minimum and local maximum. However, such constraint is quite strong, shrinking the feasible regions of optimization variable $\widetilde{\mathbf{B}}_E$. Instead, we in this work relax this constraint to permit new keypoints generation, but the intensity of newly generated shall be smaller than the keypoint to be enhanced. Specifically, this constraint is designed as follows

$$\mathcal{C}_3 : R_{\tilde{\mathbf{I}}}(\mathbf{p}_i) < R_{\mathbf{I}}\left(\mathbf{p}_k^{\max}\right), \text{ for } \mathbf{p}_i \in \mathcal{S}_k \backslash \{\mathbf{p}_k\},$$

where $\mathbf{p}_i \in \mathcal{S}_k \backslash \{\mathbf{p}_k\}$ is the DoG points around the keypoint $\mathbf{p}_k$. As shown in Fig. 2, the set $\mathcal{S}_k$ as the location collection of $U \times U \times 3$ centered on the keypoint $\mathbf{p}_k$ in the DoG image

$$\mathcal{S}_k = \{(x, y, \sigma_s) \mid |x - x_k| \le d, |y - y_k| \le d, 1 \le s \le 3\},$$

where d is the width and height of DoG plane and set to $\frac{U-1}{2}$, and s is the scale index. Note that, in constraint $\mathcal{C}_3$, $\mathbf{p}_k^{\max}$ is the keypoint with the largest keypoint intensity over the entire original image, rather than a local maximum-intensity keypoint. Thus, a new keypoint may emerge after joint watermark embedding and keypoint enhancement. However, we in the next prove that the newly generated (if possible) keypoints would not affect the embedding region localization at the extractor side.

Theorem 1. *Let* $\mathcal{P} = \{\mathbf{p}_j\}_{j=1}^n$ *be set consisting of the* n *newly generated keypoints after watermark-preserving keypoint enhancement procedure, and* $R^{\max}$ *be the maximum intensity of the keypoint from set* $\mathcal{P}$*, i.e.,* $R^{\max} = \max\left(R_{\tilde{\mathbf{I}}}(\mathcal{P})\right)$*. Then, the inequality* $R^{\max} < R_{\tilde{\mathbf{I}}}(\mathbf{p}_k)$ *holds.*

Proof. According to constraint $\mathcal{C}_3$, one can see that

$$R_{\tilde{\mathbf{I}}}(\mathbf{p}) < R_{\mathbf{I}}\left(\mathbf{p}_k^{\max}\right), \text{ for } \mathbf{p} \in \mathcal{P}, \tag{6}$$

which means that

$$\max\left(R_{\tilde{\mathbf{I}}}(\mathcal{P})\right) = R^{\max} < R_{\mathbf{I}}\left(\mathbf{p}_k^{\max}\right). \tag{7}$$

Further consider the constraint $\mathcal{C}_2$, for the keypoint $\mathbf{p}_k$ to be enhanced, we have

$$R_{\mathbf{I}}\left(\mathbf{p}_k^{\max}\right) + \xi \le |D_{\tilde{\mathbf{I}}}(\mathbf{p}_k)| = R_{\tilde{\mathbf{I}}}(\mathbf{p}_k). \tag{8}$$

Note that $\xi > 0$, and further combined with (7), we arrive at $R^{\max} < R_{\tilde{\mathbf{I}}}(\mathbf{p}_k)$. This completes the proof. □

From **Theorem** 1, one can readily conclude that, for any newly generated keypoint $\mathbf{p}$, its keypoint intensity will be strictly smaller than that of the enhanced keypoint $\mathbf{p}_k$ that for locating embedding region, *i.e.*, $R_{\tilde{\mathbf{I}}}(\mathbf{p}) \le R^{\max} < R_{\tilde{\mathbf{I}}}(\mathbf{p}_k)$. Remind that, on the extractor side, the embedding regions are localized by finding the keypoints with the largest intensities. The embedding region can thus be correctly located.

In a short summary, the proposed constrained optimization for watermark-preserving keypoint enhancement can be formulated as follows

$$\widetilde{\mathbf{B}}_E^* = \arg\min_{\widetilde{\mathbf{B}}_E} \left\| \mathbf{B}_E - \widetilde{\mathbf{B}}_E \right\|_F^2$$

$$\text{s.t.:} \begin{cases} \mathcal{C}_1 : \begin{cases} \tilde{c}_1 \ge \hat{c}_1,\ \tilde{c}_2 \le \hat{c}_2, & \text{if } w = 0. \\ \tilde{c}_1 \le \hat{c}_1,\ \tilde{c}_2 \ge \hat{c}_2, & \text{if } w = 1. \end{cases} \\ \mathcal{C}_2 : R_{\tilde{\mathbf{I}}}(\mathbf{p}_k) - R_{\mathbf{I}}\left(\mathbf{p}_k^{\max}\right) \ge \xi, \text{ if } \mathbf{p}_k \text{ is an extrema.} \\ \mathcal{C}_3 : R_{\tilde{\mathbf{I}}}(\mathbf{p}_i) < R_{\mathbf{I}}\left(\mathbf{p}_k^{\max}\right), \text{ for } \mathbf{p}_i \in \mathcal{S}_k \backslash \{\mathbf{p}_k\}. \end{cases}$$

It is easy to see that the objective function is quadratic, and the constraint sets are all linear. Consequently, both the objective function and the constraints are all convex. Therefore, the posed optimization problem is indeed a convex optimization. One can employ an off-the-shelf toolbox to tackle the above optimization problem. In the implementation, we use the `fmincon` solver provided by MATLAB to solve.

4. EXPERIMENTAL RESULTS

4.1. Experimental Setup

The size of the watermark is 64 bits. The top-2 strongest keypoints sorted by keypoint intensity are selected for locating the embedding regions. The size of one embedding region associated with the selected keypoint is 64×64. Following the embedding protocol of [5], the watermark is embedded into these two located regions repeatedly for cross-validation. Mobile device iPhone 12 Mini is mounted on a tripod for screen shooting, and the refresh rate of the display screen is 75 Hz. The experiments are all conducted in indoor. The keypoint enhancement parameter $\xi = 10$, and the embedding strength parameter $\Delta = 25$. Four test images with different levels of texture are used, *i.e.*, `Lena`, `Baboon`, `Peppers` and `Words`, as shown in Fig. 3.

Fig. 3. Test images with different levels of texture.

Fig. 4. Comparison of the watermarked image quality (PSNR). Fang *et al.* [5] (top row) and proposed (bottom row).

4.2. Comparison of the Watermarked Image Quality

To validate the effectiveness of the proposed method, We first compare the quality of watermarked images that not suffered the complex screen-shooting distortion. The representative keypoint-based method Fang *et al*. [5] is included for comparison. Note that, for a fair comparison, both the proposed method and the work [5] are tested under the same hyper-parameter setting. The watermarked image quality is evaluated in the metric PSNR, computing between the original image with the watermarked image. Clearly, a higher watermarked image quality indicates better watermark transparency. As can be seen from Fig 4, for all test images, the PSNR values produced by our method are greater than that of the work [5] at a large margin. The average PSNR gain over [5] is 1.28 dB. We believe this quality gain can be attributed to the joint optimization of *keypoint enhancement* and *watermark embedding* step. Thus, the distortion incurred by these two steps can be controlled in an optimal way, especially the low-texture regions, *e.g.*, the top of hat for `Lena`. Instead, Fang *et al.* treat these two steps independently, losing the distortion control. To facilitate better visual comparison, exemplar 64×64 local embedding regions are enlarged and shown in Table 1. As one can see, the embedding distortion of our method is dramatically reduced (the averaged PSNR gain is 1.66 dB) when compared with Fang *et al*. [5].

4.3. Comparison of the Watermark Extraction Accuracy

In this subsection, we compare the accuracy of watermark extracted from the screen-shot images, which are captured at different distances and with different angles. The evaluation metric is the average erroneous bits (AEB), which can be computed by averaging accumulated erroneous bit over the number of trials. Two more image coding-based methods, Pramila *et al*. [3] and Nakamura *et al*. [1], are also included for comparison. We would like to note that, as pointed by [5], the method of Fang *et al.* often fails watermark extraction for the document-type images. Thus, to make a fair comparison, the image `Words` is excluded.

Table 1. Comparison of the exemplar embedding region of the watermarked image (in terms of PSNR) between Fang *et al.* [5] and our method.

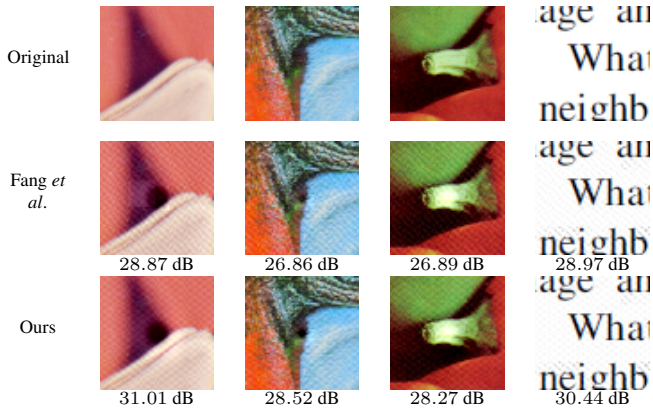


Table 2. Comparison of the average erroneous bits of the watermark for screen-shot images at various distance (cm).

Distance	Pramila [3]	Nakamura [1]	Fang [5]	Ours
45	32.00	10.50	0.50	**0.43**
55	30.19	9.44	0.69	**0.33**
65	28.62	17.81	1.69	**0.35**
75	29.94	8.75	0.88	**0.67**
Average	30.19	11.63	0.94	**0.45**

Different Distances: The experiment is conducted with the front-view camera of the mobile phone fixed on a tripod. The test distances are set from 45 cm to 75 cm with a step of 10 cm. The results are tabulated in Table 2. One can see that for all the cases, our method constantly achieves the lowest average erroneous bits 0.45 bits, significantly outperforms the competing methods [3]and [1]. Compared with the work [5], the AEB is further reduced, which can be attributed to the watermark-preserving keypoint enhancement.

Different Horizontal Angles: The experiment is conducted with the front-view camera of the mobile phone from different horizontal angles. The distance between the camera and the center of the screen is fixed as 60 cm, and the angle ranges from 15° to 60° with a step 15°. Before watermark extraction, the captured image is first geometrically corrected following the procedure suggested in [5]. Table 3 shows the shooting scenario and the perspectively-corrected images. The AEB results for different methods are provided in Table 4. Not surprisingly, the proposed method reaches the lowest AEB, except for the case Right 15° reported by [5]. Nevertheless, compared with [5], the AEB of the proposed method still drop 1.15 bits. Finally, we would like to note that, the results for the vertical angle case are similar to the

Table 3. Screen-shot images under different horizontal angles, along with the geometrically-corrected images. Top row: screen-shot images. Bottom row: corrected images.

Table 4. Comparison of the averaged erroneous bit (AEB) of the watermark extracted from images screen-shot with different horizontal angle (°).

Angle °		Pramila [3]	Nakamura [1]	Fang [5]	Ours
Right	60	31.56	12.56	2.94	**2.68**
	45	28.62	10.44	**0.69**	1.00
	30	27.50	8.44	1.12	**0.67**
	15	28.88	14.25	2.19	**0.33**
Left	15	24.94	8.88	**1.00**	**1.00**
	30	23.37	10.44	3.81	**0.66**
	45	18.31	10.13	3.31	**1.43**
	60	11.37	9.44	3.88	**2.00**
Average		24.32	10.57	2.37	**1.22**

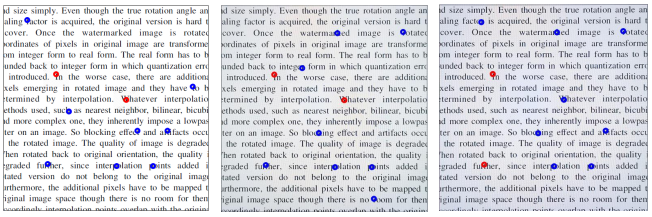
horizontal one, and thus omitted here due to space limit.

4.4. Results on Simple Textured Image

Recall that, the representative keypoint-based work Fang *et al.* [5] cannot handle the simple textured image case. This is because the two independent steps, keypoint enhancement, and watermark embedding, would have mutually adverse effects. The watermark embedding operation could degrade the keypoint localization performance. In Fig.5, we provide the keypoint localization results yielded by our method and Fang *et al.* [5], respectively. As one can see, the embedding region localization results (denoted by red circle) are almost the same as the original ones. In contrast, one keypoint for locating embedding region of Fang *et al.* drifts dramatically. This mislocalization consequently fails the following watermark extraction. Quantitatively, the erroneous bits for our method are 3 bits, while the erroneous bits for the work [5] is 28 bits, almost being the random guess (*i.e.,* 32 bits out of 64 bits are incorrect).

5. CONCLUSION

This work revisits the keypoint-based screen-shooting resilient watermarking. We argue that the limitation inherited to this framework is neglecting the interaction between keypoint enhancement and watermark embedding. Then, a watermark-preserving keypoint enhancement algorithm is proposed to resolve this issue. A convex constrained optimization model is established to unify keypoint enhancement and watermark embedding. Multiple constraints are imposed to ensure the watermark validity and blind synchronization of embedding regions. This strategy enables us jointly optimize the watermarking distortion and keypoint enhancement. Experimental results demonstrate that the proposed method achieves higher watermark extraction accuracy and better watermarked image quality simultaneously.



(a) Original (b) Ours (c) Fang *et al.*

Fig. 5. Comparison of the keypoint localization. (a) is the keypoint localization for the original image, (b) and (c) are the keypoint localization on the screen-shot images for ours and [5], respectively. Each circle denotes a keypoint, where the red ones are the keypoints for locating embedding regions.

6. REFERENCES

[1] T. Nakamura, N. Katayama, M. Yamamuro, et al., "Fast watermark detection scheme from analog image for camera-equipped cellular phone," *IEICE Trans. Inf. Syst.*, vol. 87, no. 12, pp. 2145–2155, 2004.

[2] D. Gugelmann, D. Sommer, et al., "Screen watermarking for data theft investigation and attribution," in *Proc. Int. Conf. Cyber Confl.*, 2018, pp. 391–408.

[3] A. Pramila, A. Keskinarkaus, et al., "Toward an interactive poster using digital watermarking and a mobile phone camera," *Signal Image Video Process.*, vol. 6, pp. 211–222, June 2012.

[4] J. Jia, Z. Gao, K. Chen, et al., "RIHOOP: Robust invisible hyperlinks in offline and online photographs," *IEEE Trans Cybern*, pp. 1–13, Dec. 2020 (In Press).

[5] H. Fang, W. Zhang, H. Zhou, et al., "Screen-shooting resilient watermarking," *IEEE Trans. Inf. Forensics Secur.*, vol. 14, no. 6, pp. 1403–1418, Oct. 2018.

[6] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. IEEE Int. Conf. Comput. Vis.* IEEE, 1999, vol. 2, pp. 1150–1157.

[7] L. Li, R. Bai, S. Zhang, et al., "Screen-shooting resilient watermarking scheme via learned invariant keypoints and QT," *Sensors*, vol. 21, no. 19, pp. 6554, Sept. 2021.

[8] W. Chen, N. Ren, C. Zhu, et al., "Screen-cam robust image watermarking with feature-based synchronization," *Appl. Sci*, vol. 10, no. 21, pp. 7494, July 2020.

[9] Y. Li, J. Zhou, A. Cheng, et al., "SIFT keypoint removal and injection via convex relaxation," *IEEE Trans. Inf. Forensics Secur.*, vol. 11, no. 8, pp. 1722–1735, Apr. 2016.