# Optimized Local Image Watermarking Combining Feature Point and Texture

Ying Huang
Institute of Automation, Chinese Academy of Sciences
Beijing, China
ying.huang@ia.ac.cn

Hu Guan
Digital Content Research Center of Beijing
Beijing, China
hu.guan@ia.ac.cn

Shuwu Zhang
Institute of Automation, Chinese Academy of Sciences
Beijing, China
shuwu.zhang@ia.ac.cn

Baoning Niu
College of Information and Computer
Taiyuan University of Technology
Taiyuan, China
niubaoning@tyut.edu.cn

***Abstract*—The textured regions embedded a watermark have better visual quality than the smooth regions in an image. To take advantage of the image texture being easy to hide the watermark, accurately locating the regions in an image with rich texture is significant. This paper proposes an optimized local image watermarking algorithm combining feature point and texture. The SURF feature points extracted from an image with moderate scales are selected to obtain initial watermark embedding regions. A scoring scheme by comprehensively analyzing texture, scale, and position of a region is proposed to evaluate the regions around each initial embedding region, and select the regions with the highest score from them to constitute the candidate embedding regions. Finally, the same watermarks are embedded in multiple non-overlapping embedding regions to guarantee the imperceptibility and improve the robustness. The simulation experiments on 100 images show the superiority of our proposed method compared with the state-of-the-art method in terms of imperceptibility and robustness.**

***Keywords - image watermark; texture; imperceptibility; robustness***

## I. Introduction

The use and distribution of digital images have brought convenience to people, but it has also caused serious copyright violations. The prevalence of piracy and infringement has harmed the interests of creators and has dampened their enthusiasm for creation. Digital watermarking provides a good solution for this. It is a technique of embedding a mark called a watermark in an image imperceptibly and identifying their ownership by extracting the watermark. In addition to copyright protection, image watermarking can also be applied to broadcast monitoring, forgery detection, image authentication, and covert communication.

Good imperceptibility and robustness are essential attributes of image watermarking. Imperceptibility means the watermark can be embedded into an image without hampering the quality and the utility of the image. Robustness requires the embedded watermark can be extracted correctly even when the image is distorted by various attacks. Under the premise of ensuring the imperceptibility, improving the robustness as much as possible is the goal to the research of watermarking.

A complete image generally contains both smooth regions (the grayscale of the image in the unit space varies slowly) and textured regions (the grayscale of the image in the unit space varies greatly). The smooth and texture regions have different tolerance to the watermark [1]. A watermark embedded in the smooth regions is easier to perceive than it embedded in the texture regions. The global watermarking approaches [2-11] need to utilize every pixel of an image in the embedding process, which means that they are irresistible to cropping attacks, and even impossible to avoid the smooth regions in the image. Generally, they can only avoid affecting the visual quality of the image (depending on the visual quality of the smooth regions) by reducing the embedding strength of the watermark, while the robustness of the watermark is dragged down. The local watermarking approaches [1, 12-14] embeds the watermark into specific multiple regions of an image, and generally locate these regions by the feature points. However, the robustness of these methods is closely related to the stability of the selected feature point. The watermark will not be extracted at all when the embedded regions are not located accurately. Moreover, the feature points do not deliberately capture the textured regions in an image, so these methods are prone to embedding watermark in smooth regions.

This paper addresses these issues by proposing an optimized local image watermarking combining feature point and texture, called FTOL. First, evaluate various feature points (SIFT, SURF, ORB), from which choose one with favorable scales, high efficiency and strong stability. FTOL extract the selected feature points in an image, remain the suitable feature points and make them as reference. Then the sliding window is used to traverse the surrounding regions of each feature point. The window size is set proportional to the scale of feature point to effectively resist scaling attack. Considering comprehensively the texture, grayscale, and position of each window region, select the region with the

978-1-7281-8138-7/20/$31.00 ©2020 IEEE
DOI 10.1109/ICCST50977.2020.00072

highest score, which is most suitable for embedding a watermark, as the embedding region corresponding to the feature point. The same watermarks are embedded in multiple non-overlapping embedding regions to guarantee the imperceptibility and improve the robustness. Combined with the overall design of this algorithm, the embedded watermark is not only have satisfied imperceptibility, but also can effectively resist various image processing attacks and geometric attacks.

## II. Related Work

The difference between the global watermarking approaches [2-11] and the local watermarking approaches [1, 12-14] depend on where the watermark is embedded. The global watermarking approaches embed the watermark into a complete image, which requires to use each pixel of this image. In contrast, the local watermarking approaches only use part of the image when embedding the watermark.

The global watermarking approaches can be performed in either a spatial domain or a transform domain. Spatial domain watermarking [4, 10] modifies the pixels of the host image directly for watermark embedding. In comparison, the transform domain watermarking [2, 3, 5-9, 11] carries out cross-domain transformations and embeds the watermark by modifying the coefficients of the transform domain. Discrete Fourier Transform (DFT) [2], Discrete Cosine Transform (DCT) [6, 8, 11], Discrete Wavelet Transform (DWT) [5, 9] and Singular Value Decomposition (SVD) [7] are commonly used transformations. The transform domain watermarking boasts better imperceptibility and robustness, although it is more complex than spatial domain watermarking and requires extra computation. The inter-block correlation of the DCT is applied to embed watermark by modifying difference between DCT coefficients of adjacent blocks in [3, 6]. This method can have a high watermark capacity, but it has limited resistance to attacks since only one level of DCT is used. Huynh-The et al. [5, 9] propose a channel selection mechanism to embed the watermark into the optimal channel of the DWT and determine the segmentation threshold using the Otsu's algorithm during extraction. It has a good compromise between imperceptibility and robustness, but sensitive to rotation and cropping attacks. By combining the stability of certain coefficients in the DCT and the advantages of spread spectrum and quantization schemes, the methods in [8, 11] can resist not only common image processing attacks, but also geometric distortions. However, the global watermarking approaches utilize all the pixels of the image when embedding the watermark, which means that they are difficult to resist cropping attacks, and even impossible to avoid the smooth regions in the image.

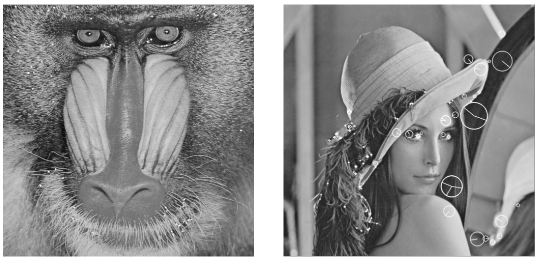
Currently, the common local watermarking approaches are the feature point-based watermarking approaches [1, 12-14]. They embed a watermark in some regions, which are located by the feature points being invariant to rotation, scaling, translation. Since the watermarked image may be distorted after geometric attack, their purpose is to take advantage of the invariance of feature points to make the embedding regions remain approximately consistent before and after the attacks. Since each feature region is only part of the image, these methods are effective against cropping attacks. Bas et al. [12] use Harris detector to extract feature points and Delaunay tessellation is applied to divide the image into a set of disjoint triangles for embedding watermark. But if the feature points extracted from the attacked image are different from the original image, the embedding watermark will not be found. Mexican-Hat wavelet [13] is utilized to located feature regions, where the watermark is embedding in after image normalization. This method is vulnerable to rotation and scaling. Ye et al. [14] embed the watermark into the circular regions centered at the extracted SIFT. However, the robustness of the feature point-based watermarking approaches is closely related to the stability of the selected feature point. The watermark will not be extracted at all when the embedded regions are not located accurately. Moreover, the feature points do not deliberately capture the texture regions in an image, so these methods also difficult to avoid embedding the watermark in smooth regions. Besides, TBAQT [1], the state-of-the-art local watermarking algorithm, only embedding the watermark in the texture regions, but since it uses the sliding window to locate the texture regions, there is a trade-off between efficiency and accuracy.

## III. Optimized Local Image Watermarking

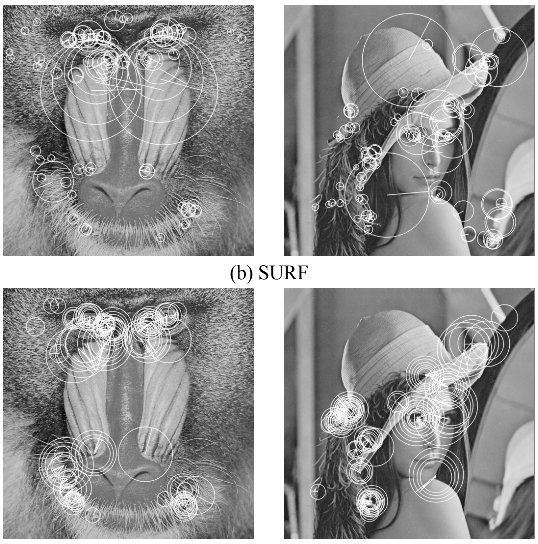
### A. Choosinging Feature Point

It is particularly important to choose the feature point that have certain invariant properties for common attacks to reduce the failure of watermark extraction due to locating deviation.

In 2004, Lowe proposed the scale-invariant feature transform algorithm (SIFT) [15], which uses the convolution of the original image and the Gaussian kernel to establish the scale space, and extracts the scale-invariant feature points on the Gaussian difference space pyramid. The algorithm has certain affine, viewing angle, rotation and illumination invariance, making it has been quickly applied in object recognition, wide baseline image matching, three-dimensional reconstruction and image retrieval. To resist scaling attacks, the size of the embedding region is usually set proportional to the scale of the feature points in a watermarking algorithm. Fig.1 (a) shows 100 SIFT feature points extracted from different images. The radius of the circle represents the feature point's scale. Due to the large difference in scale of SIFT feature points extracted from different images, it is difficult to set a moderate scale range to ensure that each image has SIFT feature points of appropriate size.



(a) SIFT



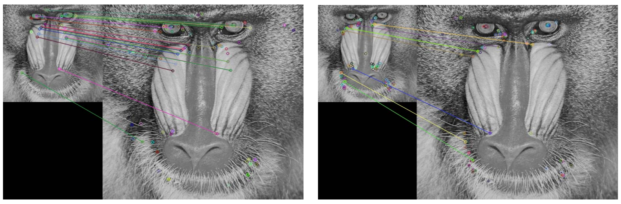
(b) SURF

(b) ORB

Fig.1. Extracted SIFT, SURF, ORB in different images

Bay proposed speeded up robust features (SURF) [16] in 2006. Aimed at the large calculation of the SIFT, the approximate Harr wavelet method and Hessian-determinant-based feature detection method are used to extract features point in SURF. The approximate Harr wavelet value can be calculated effectively in the integrated images at different scales, which simplifies the construction of second-order differential templates and improves the efficiency of feature detection in scale space. Its speed is 5-10 times that of SIFT, and their performance is comparable in most cases. Therefore, it has been widely used in the field of computer vision, especially for the occasions that require high running time. According to Fig.1 (b), SURF feature points are more abundant than SIFT for different images, and the scale range is basically the same. Thus, within a moderate scale range, each image can extract the corresponding SURF feature points.

ORB (Oriented FAST and Rotated BRIEF) [17] is a new feature proposed in ICCV2011. It combines FAST feature detector with BRIEF feature descriptor, and makes improvements and optimizations based on them. Since the corner points extracted by FAST do not have scale information and direction information, ORB uses an image pyramid to detect the corner points on each layer of pyramid to maintain the invariance of the scale, and use the gray centroid method (Intensity Centroid) to maintain rotation invariance. The advantage of ORB is the fast calculation, which is several times faster than SURF. Fig.1 (c) shows that the ORB feature points extracted from the images are relatively concentrated, and the positions of most feature points are the same. Fig.2 (a) and (b) give the matching situations between an image and its image after scaling 0.5 times of SURF and ORB respectively. The SURF has a larger number of matches than the ORB, indicating that SURF feature points are more robust under scaling attacks.



(a) SURF (b) ORB

Fig.2. The matching situation of SURF feature points and ORB feature points after image scaling 0.5 times

In summary, considering the efficiency, size, and robustness, we choose to use SURF feature points as the reference for locating the embedding region.

### *B. Determining the embedding regions*

To ensure that the regions located by the feature points have better imperceptibility, stronger robustness, and suitable capacity after embedding the watermarks, we first extract SURF feature points with greater response strength and moderate scale from the image. The initial embedding regions are centered on the extracted feature points, and the size of the embedding regions is proportional to the scale of the feature points. The sliding window of the same size as each embedding region is used to traverse the areas around each feature point. Comprehensively analyze the texture, grayscale, and position of the region in the window to give them corresponding scores, and select the window region with the highest score as the candidate embedding region. If two candidate regions overlap, the watermark information embedded in them will interfere with each other and affect the correctness of the watermark extraction. Therefore, the overlapping candidate regions also need to be excluded to obtain the final embedding regions. Consequently, this process needs to solve three problems: the selection of feature points, the determination of candidate regions, and the elimination of overlapping regions.

#### *1. The selection of feature points*

The SURF feature points with greater response strength are more stable, that is, the more likely these feature points will be extracted again after the image is attacked. Therefore, we need firstly sort the extracted SURF feature points according to the response strength from large to small, and retain the first 100 feature points, and other feature points are excluded.

Due to the scale invariance of the SURF feature points, given the size of the embedding region is proportional to the scale of the feature points, the embedding region can remain consistent before and after the image has been scaled and cropped. If the size of the embedding region is too large, the region may easily contain some smooth parts, thereby reducing the visual quality of the watermarked image. On the contrary, the capacity of the watermark will decrease, which is difficult to meet the actual requirements. Therefore, it is necessary to select a moderate scale and a certain number of feature points to locate the embedding region of the watermark. The medium scale is set as follows.

Randomly select 100 images with a size of 854*480, and extract the SURF feature points whose response strength is in the top 100. Fig. 3 shows the scale of the feature points extracted from all the images. We can see that the number the feature points with the scale in the range of 16 to 25 is large and their scale is moderate, so we use this range as a benchmark, set the medium scale range of other images as follows:

$$MinScale = \sqrt{\frac{M*N}{854*480}} \times 16 \tag{1}$$

$$MaxScale = \sqrt{\frac{M*N}{854*480}} \times 25 \tag{2}$$

where $M*N$ is the size of the image.

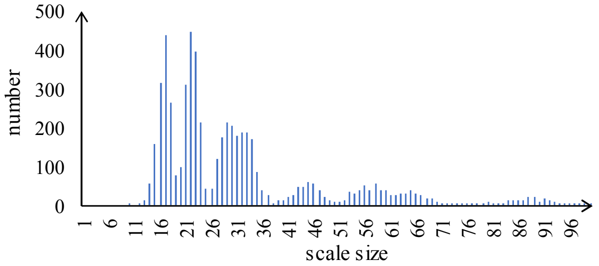


Fig.3. the scale of the feature points extracted from all the images

*2. The determination of candidate regions*

Compared to the smooth region, the watermark is embedded in the texture region with better imperceptibility at the same embedding strength. Relatively, the texture region can tolerate greater embedding strength under the same imperceptibility condition. Therefore, the richer texture region should be given a higher score to improve the robustness and imperceptibility of the watermark. Moreover, if the watermark need to be embedded in the region with the gray value being close to 0 or 255, the gray value will be beyond the boundary and be cut off, so that the watermark cannot be embedded. Thus, the more central the gray value of the region, the less likely it is to beyond the boundary, and a higher score should be given. In addition, the region closer to the edge is lower in attention and more likely to be cropped. So, the region near the center should also be given a higher score. In summary, we define the regions with the highest score around each feature point as the candidate embedding region. The score is calculated as follows:

$$T = \sum_{i=0}^{m-1}\sum_{j=0}^{n-1} \frac{\left|AC_{i,j}\right|}{AC_{\max}} \Big/ \left(m \times n - 1\right) \tag{3}$$

where $T$ represents the texture score of a region, $i$ and $j$ are not equal to 0 at the same time, the size of the region is $m \times n$, $AC_{i,j}$ represents the AC coefficient at the $i^{th}$ row and $j^{th}$ column in DCT matrix, and $AC_{\max}$ represents the maximum of all AC coefficients.

$$G = \sum_{i=0}^{m-1}\sum_{j=0}^{n-1} \left(1 - \frac{\left|P_{i,j} - 128\right|}{128}\right) \Big/ m \times n \tag{4}$$

where G represents the grayscale score of a region, and $P_{i,j}$ represents the pixel value at the $i^{th}$ row and $j^{th}$ column in a region.

$$C = \sqrt{\frac{\left(x_{rc} - x_{ic}\right)^2 + \left(y_{rc} - y_{ic}\right)^2}{2 \times m \times n}} \tag{5}$$

where $C$ represents the central score of a region, ($x_{rc}$, $y_{rc}$) is the coordinates of the center point of a region, and ($x_{ic}$, $y_{ic}$) is the coordinates of the center point of an image.

$$S = \alpha \cdot T + \beta \cdot G + \lambda \cdot C \tag{6}$$

where $S$ represents the total score of a region, $\alpha$, $\beta$, $\lambda$ are the weight of texture score, grayscale score and central score, respectively. We set $\alpha = 0.7$, $\beta = 0.2$, $\lambda = 0.1$.

*3. The elimination of overlapping regions*

The way to delete overlapping regions is that: Sort all candidate regions by its total score from large to small, traverse the candidate regions in order, and calculate the coordinate difference between the center point of the visiting candidate region and the center point of other candidate regions. If the difference in the horizontal coordinate is smaller than half of the sum of the length of the two regions and the difference in the ordinate is smaller than half of the sum of the width of the two regions, it can be judged that these two texture regions overlap. The region with a small total value in overlap regions are deleted to obtain the final embedding regions.

### C. Watermark embedding and extracting process

After locating the embedding regions in an image, the same watermark can be embedded in these regions to ensure the visual quality of the watermarked image, and the error can be corrected by analyzing multiple extracting watermarks from these regions to further improve the robustness. All existing watermarking methods can be employed to embed and extract watermark in embedding regions. We use the same watermark embedding and extracting methods as the state-of-the-art watermarking method (TBAQT) proposed in [1].

Finally, we need to determine the final watermark in multiple extracted watermarks from the embedding regions. Since the same watermark is embedded in each embedding region, the watermark extracted from the embedding region will have a strong correlation with the original watermark even if the individual bits are extracted incorrectly. While the false watermark is extracted from the region where the watermark is not embedded, and they do not satisfy the rules of embedding and extraction, which is equivalent to being randomly generated, the correlation between the false watermark and the original watermark is weak. According to the correlation principle, the false watermarks can be excluded even if the original watermark is not provided. Thus, pairwise correlation is applied in any two extracted watermarks. If the correlation between a watermark and more than half of the watermarks is higher than 0.5, this watermark is taken as the preliminary watermark. Finally, all the preliminary watermarks are aligned in bits, and the mode of each bit is taken to form the final watermark.

# IV. Experimental Evaluation

This section evaluates the performance of FTOL based on numerous simulations. Experimental setup, test datasets, evaluation criteria are described in Subsection A. Subsection B and C compare our proposed algorithm with the state-of-the-art TBAQT algorithm on imperceptibility and robustness.

## *A. Experimental setup*

TBAQT is a state-of-the-art local watermarking algorithm based on texture and uses the same watermark embedding and extracting methods in the embedding regions as our proposed algorithm, which allows for a fair comparison to ours. Therefore, the overall performance is evaluated by comparing TBAQT to FTOL in Subsection B and C. The test set [18] includes 100 randomly selected images of different sizes. These images are commonly used for evaluating image processing methods and techniques. The identical original watermark, 128 bits 0/1 sequence, is embedded into these images. Imperceptibility is judged by comparing the SSIM of the original image and with that of the watermarked image. Robustness is measured by the bit error rate (BER) between the original watermark and the extracted watermark. The final SSIM and BER values are calculated as the means of the results obtained from the 100 images in all experiments. Better results shall possess higher SSIM and lower BER. All experiments were performed on a PC with 3.4 GHz Intel Core i7 CPU and 16GB RAM, running in 64-bit Windows 7. The software for simulations was Visual Studio 2015 with OpenCV 3.4.10.

## *B. Comparison of imperceptibility*

This experiment evaluates the imperceptibility of the watermark by comparing the SSIM between the original image and the watermarked images. the two algorithms are set to the same parameter to guarantee the same embedding strength. The corresponding SSIM of each image as shown in Fig. 4.

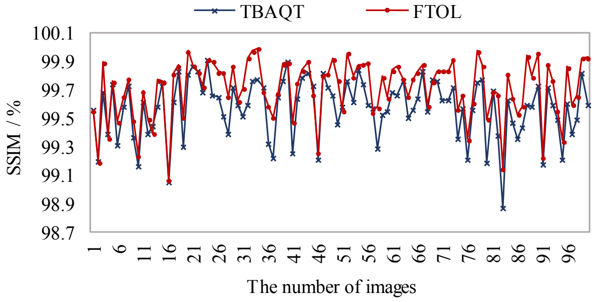


Fig. 4 SSIM of 100 images with the same embedding strength

We can see that the most SSIM of images obtained by FTOL and TBAQT are greater than 99%, which reflects very good visual quality. Because FTOL and TBAQT only embed the watermark in the regions with rich texture, which occupies a small proportion in the image and has good concealment to the watermark. Consequently, the entire image can accommodate a higher SSIM. Compare FTOL and TBAQT, The SSIM obtained by FTOL is slightly greater than TBAQT. It illustrates that the texture of the embedding region located by FTOL is richer than TBAQT, so that the imperceptibility of the watermark embedded by FTOL is better.

## *C. Comparison of robustness*

This subsection compares the capability of FTOL and TBAQT to recover the hidden watermark in various scenarios, in the context of image processing and geometric attacks. The experiment sets the appropriate embedding strength for each algorithm, so that their average SSIM of the 100 watermarked images obtained by TBAQT and FTOL are basically the same (their average SSIM are 99.335% and 99.341% respectively), to compare the robustness of the two algorithms under the same imperceptibility conditions. The types and parameters of the attacks and simulation results are listed in Table I.

TABLE I. BERs under various attacks

| **Attack Types** | | **Parameters** | **TBAQT (%)** | **FTLO (%)** |
|---|---|---|---|---|
| Image Processing Attacks | JPEG Compression | 30% | 0.65 | **0.62** |
| | | 50% | 0.26 | **0.26** |
| | | 70% | **0.00** | **0.00** |
| | Gauss Noise | 0.05 | 0.54 | **0.32** |
| | | 0.03 | 0.11 | **0.05** |
| | | 0.01 | **0.00** | **0.00** |
| | Salt & Pepper Noise | 0.05 | 0.02 | **0.00** |
| | | 0.04 | **0.00** | **0.00** |
| | | 0.03 | **0.00** | **0.00** |
| | Average Filtering | 3×3 | 6.32 | **5.85** |
| | Median Filtering | 3×3 | 4.23 | **3.78** |
| | Histogram Equalization | —— | 0.00 | **0.00** |
| | Luminance Change | 0.6 | **0.00** | **0.00** |
| | | 2 | 0.02 | **0.00** |
| Geometric Attacks | Flipping | horizontal | **0.00** | **0.00** |
| | | vertical | **0.00** | **0.00** |
| | Rotation | 90° | **0.00** | **0.00** |
| | | 180° | **0.00** | **0.00** |
| | | 270° | **0.00** | **0.00** |
| | cropping | 25% | **0.00** | **0.00** |
| | | 32% | **0.00** | **0.00** |
| | | 50% | 0.01 | **0.00** |
| | Shielding | 0.5 | **0.00** | **0.00** |
| | | 0.6 | **0.00** | **0.00** |
| | | 0.7 | 0.05 | **0.05** |
| | Resizing | 0.6 | **0.00** | **0.00** |
| | | 1.6 | **0.00** | **0.00** |
| | | 2 | **0.00** | **0.00** |
| | Scaling | 0.6 | 3.85 | **3.21** |
| | | 1.6 | **0.00** | **0.00** |
| | | 2 | **0.00** | **0.00** |

Both TBAQT and FTOL perform well against most of attacks and FTOL extracts watermark with slightly higher accuracy than TBAQT. This is attributed that the richer the texture of the embedded region, the greater the embedding strength of the watermark, so it can effectively resist various attacks. For low-pass filtering (including mean filtering and median filtering), their performance is mediocre, Because when the filtering template processes the edge of the embedding region of TBAQT, pixels near the edge and outside the embedding region are introduced, which affects the pixels at the edge of the embedded region, resulting in the watermark embedded in these locations being extracted incorrectly. The shielding attack cuts the center of the image compared to the upper left corner of the cropped image, which has a more severe effect on the image. Because they embed the same watermark in multiple non-overlapping texture regions, which occupies a small part in the image and has no overlap. Therefore, even if some of the embedding regions are cut, there may still be embedding regions that are not affected. For scaling attack, they only extract watermarks incorrectly with a scaling ratio less than 0.6. This is because the embedding region itself is much smaller than the image. When the image is scaled to a small size, the size of the embedding region is proportionally smaller, resulting in some watermark bits being lost and not correctly extracted.

## V. Conclusion

This paper proposes an optimized local image watermarking combining feature point and texture. In order to effectively located the embedding region with rich texture, first use the SURF feature points with moderate scales to locate the initial embedding regions. A scoring scheme by comprehensively analyzing texture, scale, and position is proposed to evaluate the region around the feature points, and select the regions with the highest score near each feature point as the candidate embedding regions. the watermark is embedded in the non-overlapping embedding regions of the image to ensure the imperceptibility and robustness of the watermark. The experimental results not only prove the correctness of the theoretical analysis, but also show that the proposed algorithm can not only guarantee the visual quality of the watermarked image, but also effectively resist various common image processing attacks and geometric attacks.

## Acknowledgment

This work was supported by the National Key R&D Program of China (2019YFF0302800) and the Key R&D Program of Shanxi (201903D421007). It was also the research achievement of the Key Laboratory of Digital Rights Services, which is one of the National Science and Standardization Key Labs for Press and Publication Industry.

## References

[1] Y. Huang, B. Niu, H. Guan, and S. Zhang, “Image texture based adaptive watermarking algorithm,” Journal of Beijing University of Aeronautics and Astronautics, vol. 45, no. 12, pp. 2403-2414, 2019.

[2] M. Cedillohernandez, F. Garciaugalde, M. Nakanomiyatake, and H. Perezmeana, “Robust Hybrid Color Image Watermarking Method based on DFT Domain and 2D Histogram Modification,” Signal, Image and Video Processing, vol. 8, no. 1, pp. 49-63, 2014.

[3] C. Das, S. Panigrahi, V. K. Sharma, and K. Mahapatra, “A Novel Blind Robust Image Watermarking in DCT Domain Using Inter-Block Coefficient Correlation,” Aeu-International Journal of Electronics and Communications, vol. 68, no. 3, pp. 244-253, 2014.

[4] T. Zong, Y. Xiang, I. Natgunanathan, S. Guo, W. Zhou, and G. Beliakov, “Robust Histogram Shape-based Method for Image Watermarking,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 25, no. 5, pp. 717-729, 2015.

[5] T. Huynhthe, O. Banos, S. Lee, Y. Yoon, and T. Letien, “Improving Digital Image Watermarking by Means of Optimal Channel Selection,” Expert Systems with Applications, vol. 62, no. 1, pp. 177-189, 2016.

[6] S. A. Parah, J. A. Sheikh, N. A. Loan, and G. M. Bhat, “Robust and Blind Watermarking Technique in DCT Domain Using Inter-Block Coefficient Differencing,” Digital Signal Processing, vol. 53, pp. 11-24, 2016.

[7] N. M. Makbol, B. E. Khoo, T. H. Rassem, and K. Loukhaoukha, “A New Reliable Optimized Image Watermarking Scheme based on the Integer Wavelet Transform and Singular Value Decomposition for Copyright Protection,” Information Sciences, vol. 417, pp. 381-400, 2017.

[8] Y. Huang, H. Guan, B. Niu, and S. Zhang, “A Spread-Spectrum Watermarking Scheme with Adaptive Embedding Strength and PSNR Guarantee,” proceeding in International Conference On Anti-Counterfeiting, Security, And Identification, 2018, pp. 82-87.

[9] T. Huynhthe *et al.*, “Selective Bit Embedding Scheme for Robust Blind Color Image Watermarking,” Information Sciences, vol. 426, pp. 1-18, 2018.

[10] Q. Su and B. Chen, “Robust Color Image Watermarking Technique in the Spatial Domain,” Soft Computing, vol. 22, no. 1, pp. 91-106, 2018.

[11] Y. Huang, B. Niu, H. Guan, and S. Zhang, “Enhancing Image Watermarking with Adaptive Embedding Parameter and PSNR Guarantee,” IEEE Transactions on Multimedia, vol. 21, no. 10, pp. 2447-2460, 2019.

[12] P. Bas, J.-M. Chassery, and B. M. Macq, “Geometrically Invariant Watermarking Using Feature Points,” IEEE Transactions on Image Processing, vol. 11, no. 9, pp. 1014-1028, 2002.

[13] C. W. Tang and H. M. Hang, “A Feature-based Robust Digital Image Watermarking Scheme,” IEEE Transactions on Signal Processing, vol. 51, no. 4, pp. 950-959, 2003.

[14] X. Ye, X. Chen, D. Meng, and Y. Wang, “A SIFT-based DWT-SVD Blind Watermark Method against Geometrical Attacks,” proceeding in International Congress on Image and Signal Processing, 2015, pp. 323-329.

[15] D. G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints,” International Journal of Computer Vision, vol. 60, no. 2, pp. 91-110, 2004.

[16] H. Bay, T. Tuytelaars, and L. V. Gool, “SURF: Speeded up robust features,” proceeding in the 9th European conference on Computer Vision, vol. 1, 2006.

[17] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski, “ORB: an efficient alternative to SIFT or SURF,” proceeding in IEEE International Conference on Computer Vision (ICCV 2011), Barcelona, Spain, 2011.

[18] http://www.cs.washington.edu/research/imagedatabase/.