Hindawi
Security and Communication Networks
Volume 2021, Article ID 5558873, 12 pages
https://doi.org/10.1155/2021/5558873

Research Article

Reversible Privacy Protection with the Capability of Antiforensics

Liyun Dou,[1] **Zichi Wang**,[1] **Zhenxing Qian**,[2] **and Guorui Feng**[1]

[1]*School of Communication and Information Engineering, Shanghai University, Shanghai, China*
[2]*Shanghai Institute of Intelligent Electronics and Systems, School of Computer Science, Fudan University, Shanghai, China*

Correspondence should be addressed to Zhenxing Qian; zxqian@fudan.edu.cn

Received 1 February 2021; Revised 8 March 2021; Accepted 18 March 2021; Published 12 April 2021

Academic Editor: Beijing Chen

In this paper, we propose a privacy protection scheme using image dual-inpainting and data hiding. In the proposed scheme, the privacy contents in the original image are concealed, which are reversible that the privacy content can be perfectly recovered. We use an interactive approach to select the areas to be protected, that is, the protection data. To address the disadvantage that single image inpainting is susceptible to forensic localization, we propose a dual-inpainting algorithm to implement the object removal task. The protection data is embedded into the image with object removed using a popular data hiding method. We further use the pattern noise forensic detection and the objective metrics to assess the proposed method. The results on different scenarios show that the proposed scheme can achieve better visual quality and antiforensic capability than the state-of-the-art works.

1. Introduction

Photo sharing has become a widespread user activity with the advent of intelligent mobile devices and online social networks (OSN). Image distributions cause privacy concerns and the requirement to modify permissions since the shared content contains sensitive data of users. By providing unique rights to selected communicating parties in OSN, users' security and privacy can be strengthened. A well-established form of privacy protection is to blur a part of an image, which can be achieved by various image processing techniques, for example, blurring, mosaic, masking, and object removal, as shown in Figure 1. In these methods, the first three must introduce a significant amount of distortion to hide the underlying content. Object removal provides more natural viewing conditions and is able to protect the content. This process is reversible such that the original data can be accessed with permissions [1].

After object removal in an image, the broken parts can be inpainted using the surrounding contents. Generally, image inpainting algorithms can be divided into three groups, including the statistical-based, the diffusion-based, the patch-based, and the deep generative models-based methods [2, 3]. Statistical methods use parametric models to describe textures but fail when additional intensity gradients are applied [4]. Diffusion-based methods propagate pixels from the known areas of the image [5–7] using smoothness priors; however, blurring occurs when large and high-frequency regions need to be inpainted. Patch-based and deep generative models are the most widely used, where the former fills the holes in the image using the patch from local or global search regions [8–12] and the latter exploits semantics learned from large-scale datasets [13–15]. None of the inpainting algorithms have considered the secrecy of the inpainted areas from the security perspective. The inpainted images are easy to be detected and located by forensic algorithms.

In this paper, we propose a new privacy protection scheme using image inpainting and data hiding, which realizes the antiforensics capability. When considering the undetectability of edge inpainting, we use the algorithm of the DFNet network [16]. The regions around the broken edge are inpainted twice, and the inpainting results are fused to achieve the capability of antiforensics. By combining image dual-inpainting and data hiding, a privacy protection scheme with antiforensics capability is realized. We combine local variation within and between channels and use the popular data hiding algorithm HILL [17] to embed the

Figure 1: Common privacy protection methods: (a) original image; (b) blurring; (c) mosaic; (d) masking; and (e) object removal.

protection data. The rest of this paper is organized as follows: we introduce the related works in Section 2. The proposed method is depicted in Section 3. Experimental results and analysis are provided in Section 4. Section 5 concludes the whole paper.

2. Related Works

In this section, we introduce the works that are related to the proposed method, including the image inpainting, the data hiding, and the image forensics.

2.1. Image Inpainting. Image inpainting is a method to fill the missing information in an image and is quite important in the field of image processing. Nowadays, the deep generative models-based methods are widely used in the field of image inpainting [14, 18–23]. Numerous methods can be divided into two categories [24]. One approach is to use an effective loss function or construct an attention model to fill in the missing regions to try to make the content more realistic. They use the content in the background to fill, and a better way is to fix the unknown region by partial convolution [18]. The other approach focuses on structural consistency. To ensure the continuity of the image structure, these approaches usually adopt edge-based contextual priors. For example, [19] designed an edge linking strategy that can well solve the image semantic structure inconsistency problem.

Regardless of the inpainting method, there is a discontinuous transition zone at the edge of the inpainting. This area will become a forensic object and thus easy to locate the inpainting area by someone who is interested, which is quite unsafe. In order to not only achieve a good visual effect but also secure safety, a smooth transition needs to be achieved in advance. An iterative method to optimize the pixel gradients in the edge transition regions is proposed in [25],. The quality of fusion depends on whether the incorporated content is consistent with the original content in terms of gradient changes. Thus, Hong et al. [16] design a learnable fusion block to implement pixel-level fusion in the transition region, which is named deep fusion network for image completion (DFNet). The results show that DFNet has superior performances, especially in the aspects of harmonious texture transition, texture detail, and semantic structural consistency.

2.2. Data Hiding. To further optimize the data embedding problem in information hiding, adaptive embedding algorithms are widely proposed. Among them, STC (Syndrome Trellis Coding) [26] based adaptive architectures are most preferred by researchers. This method uses a predefined distortion function to minimize the additive distortion between stego and cover. For the multiscale characteristics of the image space, the design of the distortion function has attracted more and more attention. For instance, Li et al. [17] proposed a new distortion function for image information hiding. The cost function is composed of a high-pass filter and two low-pass filters. The high-pass filter is used to locate the difficult-to-predict parts of an image and then employ the low-pass filters to make the low-cost values more clustered. Furthermore, the methods of MiPOD (Minimizing the Power of Optimal Detector) [27] and ASO (Adaptive Steganography by Oracle) [28] were proposed one after another. In addition, a number of distortion functions have been proposed for JPEG steganography as well, such as IUERD (Improved UERD) [29], UED (Uniform Embedding Distortion) [30], and RBV (Residual Blocks Value) [31].

In addition, some work uses machine learning algorithms to design steganalysis tools to detect steganography. Most of these approaches learn a general steganography model through a supervised strategy and then use it to distinguish suspicious images [32–35]. With the rapid development of deep learning, the performance of steganalysis has been greatly improved [36–38]. However, depth features still have limitations in steganalysis [39]. For example, the truncation and quantization operations in the feature extraction process are difficult to be learned by existing networks. Therefore, feature extraction is still a challenge in steganalysis, and many rich feature sets have been used for JPEGY steganalysis. The main available feature sets include JPEG rich-model [40], DCTR GFR (Gabor filter residuals) [41], and DCTR (Discrete Cosine Transform Residual) [42]. In the classification process, the ensemble classifier is considered to be effective in measuring the feature set [43, 44].

2.3. Image Forensics. Currently, there are two forensic methods of detecting image inpainting [45, 46]. In [45], the authors find that the Laplacian operations along the isophote direction in the inpainted regions are different from the other regions. Accordingly, the inpainted regions can be identified by exploring the changes of local variances between intra- and interchannels. In [46], noise pattern analysis is used to locate the inpainted regions. For the images captured by one camera, the noise patterns in each image are approximately the same and vice versa. Therefore, the noise pattern can be used as the fingerprint for a camera, which is widely adopted in image forensics.

The noise pattern analysis algorithm in [46] is popular. In this model, the pixel values can be constructed by ideal

pixel values, multiplicative noises, and various additive noises, which can be expressed by

$$I = f((I + K) \cdot O) + a, \tag{1}$$

where I and O are the actual pixel and ideal pixel value of the natural scene, a is the sum of various additive noises, $f(\bullet)$ is the camera processing like CFA interpolation, and K is the coefficient for noise pattern. In equation (1), the multiplicative noise $K{\cdot}O$ is the theoretical expression of the noise pattern, which is a multiplicative noise in the high frequencies related to the image contents. Generally, we can use a low-pass filter to remove the additive noises. The residual noise is then used to estimate the noise pattern [47], as shown in the following equation:

$$p = I - F(I), \tag{2}$$

where $F(\bullet)$ is the low-pass filter and p is the estimated noise pattern. The noise pattern can be used to distinguish the content from different images. Therefore, the inpainted region can be detected after extracting the noise pattern from each part of the image.

During inpainting, since there are limited pixels around the damaged regions, each diffusion is smoothed based on the surrounding pixels to accomplish the diffusion. Therefore, the pixels located in the inpainted region satisfy $I_t^n(i, j) = 0$, which means that the results of Laplacian operation on this position remain unchanged along the isophote direction after the diffusion-based inpainting. The Laplacian variation along the isophote direction can be calculated by

$$\delta_{\Delta I(i,j)} = \Delta I(i, j) - \Delta I(i_v, j_v), \quad \forall (i, j) \varepsilon I \tag{3}$$

where $\Delta I(i, j)$ is the (i, j)-th Laplacian value and $\Delta I(i_v, j_v)$ is the result of Laplacian operation on a virtual pixel on (i_v, j_v). The virtual pixel is located at the direction of $\nabla I^{\perp}(i, j)$, and its distance to the pixel $I(i, j)$ is identical to 1.

3. Proposed Method

In this section, we present an antiforensic framework to perform object removal in images using dual-inpainting and data hiding. As shown in Figure 2, the proposed framework contains four parts. We first select the protected area interactively and calculate the percentage of the area in the whole image. Then, the background with the missing protected area was inpainted. In order to achieve a satisfactory visual effect and be as forensic-free as possible, an image dual-inpainting algorithm is proposed, as shown in Figure 3 and described in Section 3.1–3.3. For the inpainted image, region segmentation is performed based on the changes of local variances between the intra- and interchannels. Meanwhile, the protected region is embedded into the background after converting it into a bitstream by combining the HILL embedding algorithm and considering the segmentation. On the recipient side, we can extract the embedded data, fuse it with the background image, and recover the original image.

3.1. Protection Region Selection. We interactively specify the area in an image to be protected, which also means that the hidden area is determined. After that, we calculate the number of the pixels to be hidden, including the values and coordinates of these RGB pixels. The pixels are converted into bit stream for embedding. We define the bits of each pixel as 5×9, in which "5" stands for pixel values in three channels, horizontal and vertical coordinate values, and "9" means that we convert each decimal to 9 bits. In a color image, information can be embedded in all three channels at each position. Thus, the maximum amount of embeddable information is three times the image size. The maximum embedding ratio T is calculated to be 6.66% per image. Let t be the proportion of the selected protection region. The proportion should be smaller than a predefined threshold T. An example of the interactive region selection is shown in Figure 4.

3.2. Background Processing. After specifying the protection area, we remove the contents in this area and inpaint the image. When inpainting large areas, it is often not possible to perfectly blend the inpainted area with the existing content, especially in the edge areas [16]. To fill this gap, the DFNet network [23] introduces a fusion block, which combines the structural and texture data and smoothly blends them during the inpainting process. As shown in Figure 5, I is the input image, F_k is the feature maps from k-th layer, and I_k is resize of I. The learnable function M is designed to extract the raw completion C_k from feature maps F_k, which is as follows:

$$C_k = M(F_k), \tag{4}$$

where M denotes the channel conversion operation, which converts n channel feature maps into 3-channel images under the condition of constant resolution.

In addition, another learning function A is used to generate the alpha composition map a_k:

$$a_k = A(F_k, I_k). \tag{5}$$

Map a_k usually is obtained by synthesis from a single channel or 3 channels for imagewise alpha composition. Previous experience has demonstrated that channelwise alpha composition performs better. A is a convolutional module which consists of 3 convolutional layers with kernel sizes of 1, 3, and 1, respectively. The final result I_k' is achieved by

$$I_k' = B(a_k, C_k, I_k) = a_k * C_k + (1 - a_k) * I_k. \tag{6}$$

The fusion block makes the image inpainted by the DFNet network almost visually free of edge discontinuity. Although the DFNet network achieves good visual results, it is not suitable for privacy protection since it can be easily localized for forensics. For example, pattern noise of the image detection reveals clear artifacts in the restoration edge area. To conceal these traces and achieve the privacy-preserving, further manipulation of the inpainting image is required.

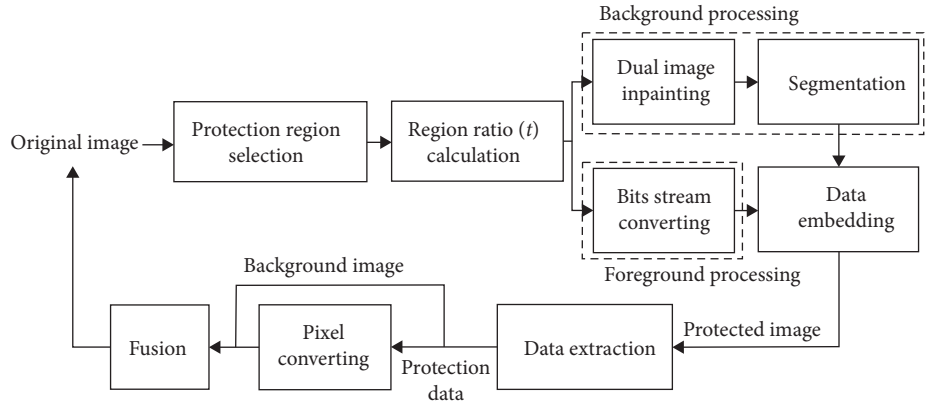


Figure 2: Architecture of the proposed scheme.

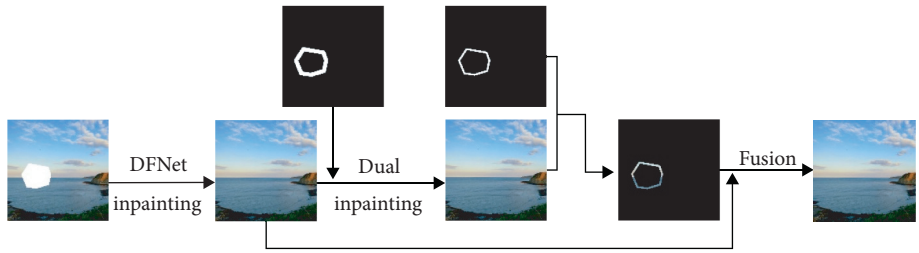


Figure 3: Dual-inpainting process architecture.

The detection area is mostly found in the edge area of the restoration, so we consider secondary processing of the edge area to eliminate the traces left during the restoration process. In this process, we used the mathematical morphology of the dilation operation and the erosion operation. In the dilation operation, the structural element B is used as an external window to increase the overall boundary of the target image. In the erosion operation, the structural elements serve as the internal windows to eliminate the boundary of the image. The dilation operation is expressed by equation (7) and erosion operation can be expressed by equation (8):

$$I \oplus B = \left\{(i, j) | B_{(i,j)} \cap I \neq \varnothing\right\}, \tag{7}$$

$$I \Theta B = \left\{(i, j) | B_{(i,j)} \subseteq I\right\}, \tag{8}$$

$$B_{(i,j)} = \{(x, y) | x = m + i, y = n + j, (m, n) \in B\}. \tag{9}$$

The specific dual-inpainting process is shown in Figure 3. Firstly, the background image should be inpainted using the DFNet network. Then, we apply a mathematical morphological dilation operation on the edges of the broken region mask map. Based on this mask map, secondary inpainting of the primary inpainted image is performed in the region. In addition, mathematical morphology erosion operation is then applied to the secondary inpainted region, leaving only a portion of the region close to the edge. Note that the dilation operation uses a larger size of structural elements than that of the erosion operation to ensure the results of the secondary inpainting of the lower edge are preserved. The results of the secondary inpainting of the edge region are fused with the primary repair map to obtain a graph of the experimental results of antiedging detection.

3.3. Area Segmentation and Data Hiding. To hide the secret data of the protection region, we employ the popular data hiding framework which can be achieved by STC [17]. We improve the popular cost function HILL for STC to fit the requirements in our method.

In the STC framework, the theoretical minimum steganography distortion D for the marked image with an embedding amount of γ (bits) can be defined as

$$D = \sum_{i=1}^{M} \sum_{j=1}^{N} \left(p_{i,j}^{+} \rho_{i,j}^{+} + p_{i,j}^{-} \rho_{i,j}^{-}\right),$$

$$p_{i,j}^{+} = \frac{e^{-\lambda \rho_{i,j}^{+}}}{\left(1 + e^{-\lambda \rho_{i,j}^{+}} + e^{-\lambda \rho_{i,j}^{-}}\right)}, \tag{10}$$

$$p_{i,j}^{-} = \frac{e^{-\lambda \rho_{i,j}^{-}}}{\left(1 + e^{-\lambda \rho_{i,j}^{+}} + e^{-\lambda \rho_{i,j}^{-}}\right)},$$

where $p_{i,j}^{+}$ and $p_{i,j}^{-}$ are the probabilities of adding 1 or subtracting 1 on $c_{i,j}$, $0 < p_{i,j}^{+} + p_{i,j}^{-} < 1$, and $\rho_{i,j}$ stands for the distortion values used to measure the effects of modification. The parameter λ ($\lambda > 0$) is used to make the ternary data entropy of the modification probability identical to the capacity γ, as shown in the following equation:

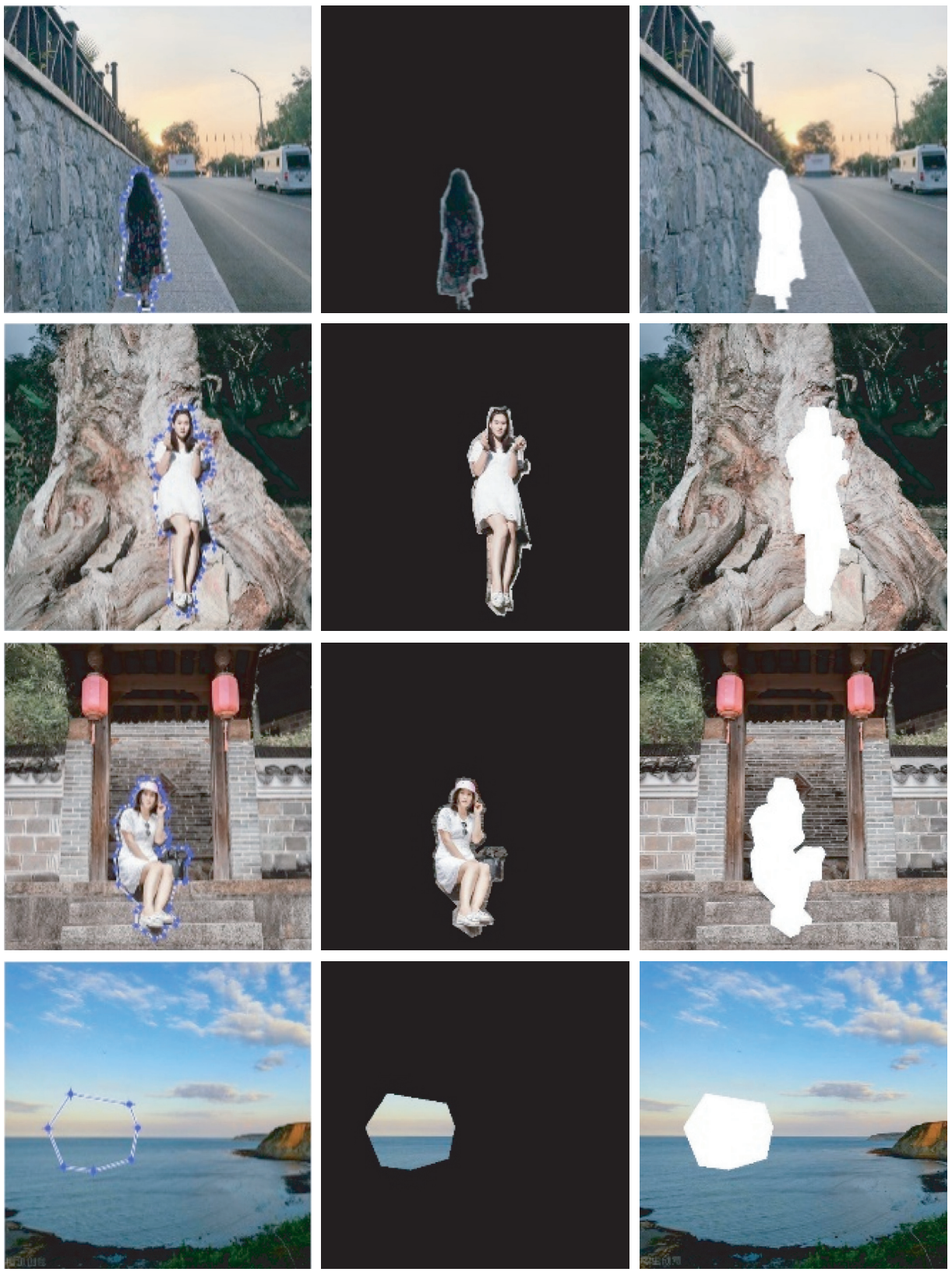


Figure 4: Protection region selection. (a) Interactive selection. (b) Protected areas. (c) Background image.

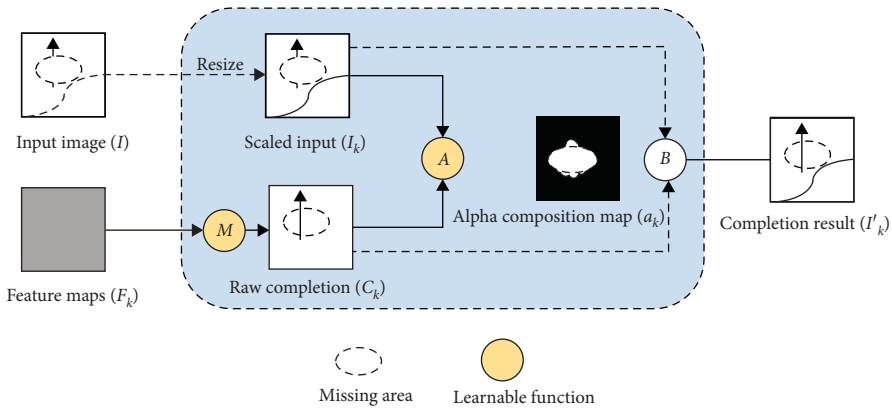


Figure 5: Illustration of fusion block.

$$-\sum_{i=1}^{M}\sum_{j=1}^{N}\left\{p_{i,j}^{+}\log_2 p_{i,j}^{+}+p_{i,j}^{-}\log_2 p_{i,j}^{-}+\left\{\left(1-p_{i,j}^{+}-p_{i,j}^{-}\right)\log_2\left(1-p_{i,j}^{+}-p_{i,j}^{-}\right)=\gamma. \right.\right. \tag{11}$$

To achieve the minimum distortion D, STC encoding is used. Let the secret bits $m = [m_1, m_2, \ldots, m_\gamma]^T \in \{0, 1\}^\gamma$, cover pixels $c = [c_1, c_2, \ldots, c_{MN}]^T$, and stego pixels $y = [y_1, y_2, \ldots, y_{\mathrm{MN}}]^{\mathrm{T}}$. Then, m can be embedded into c using

$$\begin{aligned}\mathrm{Emb}(c,m) &= \arg\min_{\mathbf{y}_l\in C(\mathbf{m})} D(c,y),\\ D(c,y) &= \sum_{c_i\neq y_i}\rho_i^{(y_i-c_i)},\end{aligned} \tag{12}$$

where $y_l \in \{0, 1\}^{MN}$ is the least significant bits of the stego image, $C(m) = \{z \in \{0, 1\}^{MN} | Hz = m\}$ is the companion set of m, and $H \in \{0, 1\}^{\gamma\times MN}$ is a predefined low-density parity test matrix related to embedding speed and embedding efficiency. The embedded bits m can be extracted simply by a matrix multiplication operation:

$$m = Hyl. \tag{13}$$

To fit the requirements in our method, we improve the popular cost function HILL for STC by combining variations within and between adjacent pixel channels. Specifically, we divide the cover image into four regions (marked with green, blue, black, and red in Figure 6) using the cost values of HILL and edge connectivity. The pixel complexity of the four regions decreases in the order of green, blue, black, and red. In other words, the green region has the most complex pixels and is the best embedding region for the whole image. Therefore, secret bits are embedded into the green region preferentially.

4. Experimental Results

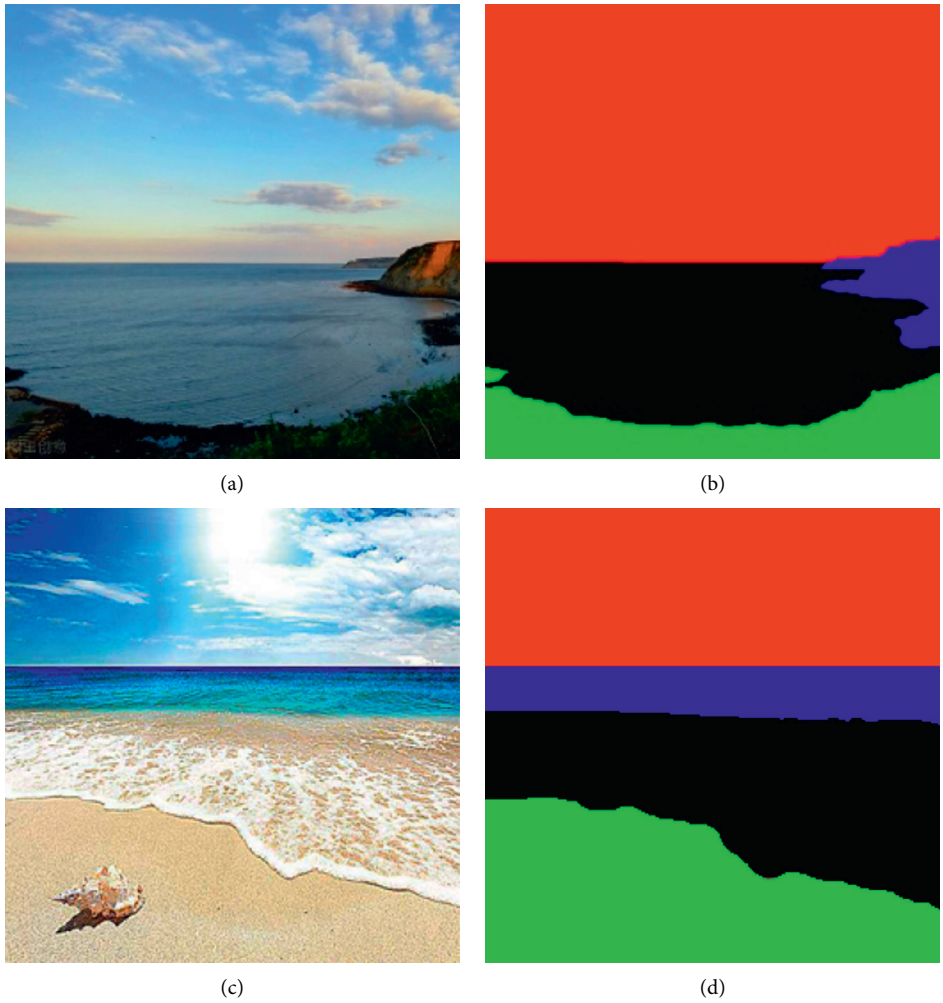
This section presents the experimental evaluation results. Firstly, we introduce the database employed and the corresponding parameters. Then, experiments for each part are presented in turn and their validity is demonstrated.

4.1. Performance for Antiforensics. To evaluate the performance of antiforensics, we randomly select images from the database for validation and interactively select the areas to be protected, as mentioned in Section 2.

In each image, the selection of the protected area is irregular shape generally. For later embedding of data, we strictly controlled the ratio of protected areas to the image to less than 6.66%. We use two separate forensic approaches for the forensic analysis of our results: one is pattern forensics by pattern noise, and the other one is based on changes between and within adjacent pixel channels.

Firstly, we select 50 landscape images sized 512×512 from Today's Headlines. As shown in Figure 7, we selected four of them, $I1$, $I2$, $I3$, and $I4$ in turn. Table 1 lists the space proportion t and the number of pixels to be embedded in the whole image of the corresponding protection area of the four images in Figure 7. Figure 7(c) shows the images after being inpainted based on DFNet, Figure 7(e) shows the images after being inpainted by our method, and Figures 7(d) and 7(f) show the pattern noise maps of Figures 7(c)~7(e), respectively. Comparing with the ground truth Figure 7(b), we find that Figure 7(d) has obvious traces at the repair edges, which makes the repair region easy to be forensically located. While our method overcomes this drawback well, it is difficult to forensically locate our tampered region from the pattern noise forensic aspect only. It shows that our aspect has a good antipattern noise forensic effect.

In Figure 8, we show the experimental results for five images ($M1$, $M1$, $M3$, $M4$, and $M5$) in the UCID database, sized 384×512. Table 2 lists the space proportion t and the number of pixels to be embedded in the whole image of the corresponding protection area of the four images in Figure 8. Two traditional methods and a deep learning method are used for comparison, where the traditional methods are edge-oriented and Delaunay-oriented provided by G'MIC [48], a full-featured open-source framework for image processing. The deep learning-based one is the DFNet method mentioned in [16].



(a) (b)

(c) (d)

Figure 6: Examples for area segmentation. (a) Original image. (b) The result of area segmentation.

Comparing from the subjective vision, both our experimental results and the deep learning method outperform the traditional method and achieve good visual connectivity at the edges. In particular, in row 7 of Figure 8, the effect at the red petal achieves a good visual effect after blending with the primary restored image by our secondary processing of the restored edges.

In addition, we localized the inpainted image for forensics by the forensic algorithm proposed in [46], as shown in the even rows of Figure 7. The traditional restoration-based algorithm is easy to be detected and located, and the DFNet-based restoration also achieved good antiforensic results. However, the images obtained by our method are more suitable to hide the area to be protected. In particular, the results are better when the area to be protected accounts for less than 4% of the whole image.

In Table 3, we show the F1 values of the five images in Figure 8, where a smaller F1 value indicates a worse ability to correctly locate the image and indicates that we have a better antiforensic effect. We can see from Table 3 that our method is superior in terms of objective indicators.

4.2. Experiment Setup. In our experiments, we use the free user-shared image dataset provided by Today's Headlines, which contains a large number of people landscapes, and various life images. We also use the UCID database. Based on the maximum amount of data that can be embedded in an image, it can be calculated that the size of the protected area must not exceed 6.66% of the whole image ($T = 6.66\%$) no matter how large the image size is. For the structural elements for the mathematical morphology of the background process, the circular structure is employed since it has a

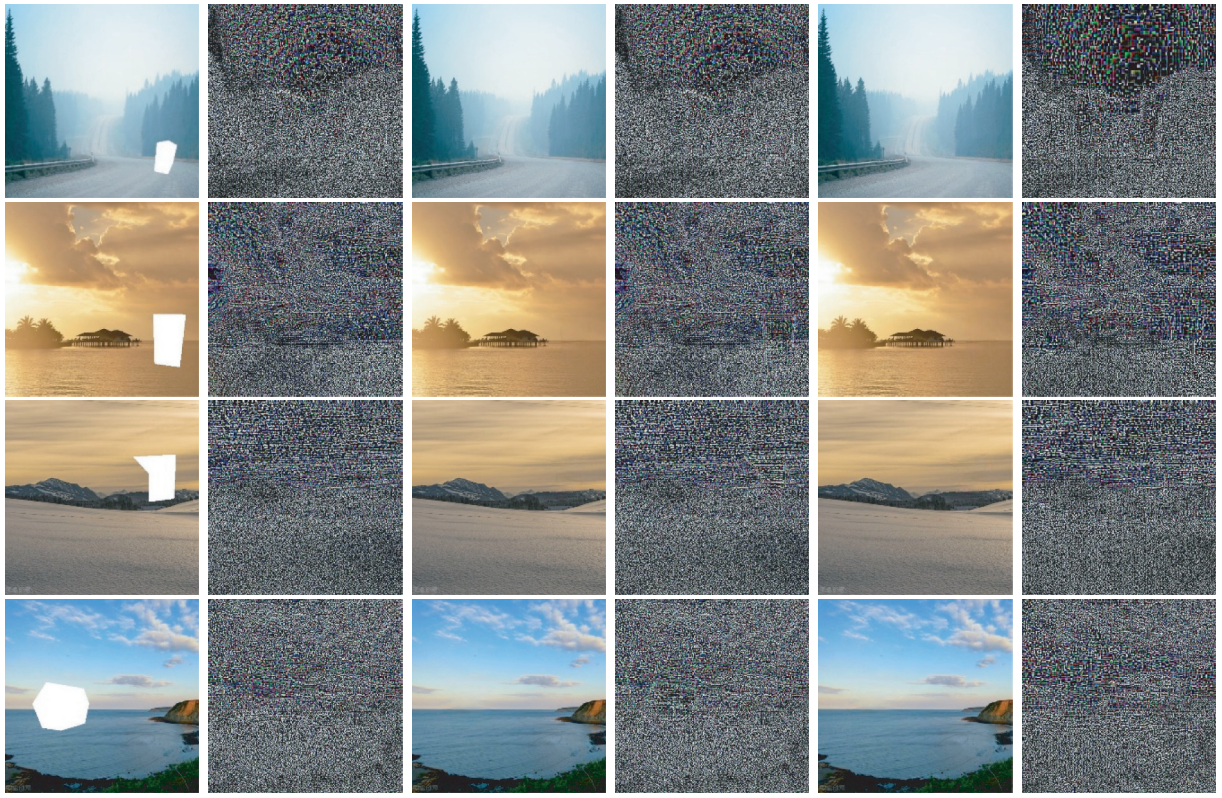


Figure 7: Examples from Today's Headlines. (a) Original image; (b) ground truth; (c) images inpainted via DFNet; (d) the pattern noise of (c); (e) images restored via our method; and (f) the pattern noise of (e).

Table 1: The percentage of protected areas in the whole image(t) and the total number of pixels in the protected area(p), $I1$, $I2$, $I3$, and $I4$ represent the four pictures in Figure 7, respectively.

Image	$I1$	$I2$	$I3$	$I4$
T	1.52%	3.99%	3.51%	5.53%
P	3985	10459	9201	14497

smoother edge where the structure size is 10 for the dilation operation and 5 for the erosion operation.

To evaluate the performance of image dual-inpainting against detection and localization, we adopt F1-score, peak signal-to-noise ratio (PSNR), and mean square error (MSE) objective indicators to evaluate the inpainting results:

$$\mathrm{F1} = \frac{2\mathrm{TP}}{(2\mathrm{TP} + \mathrm{FN} + \mathrm{FP})}, \tag{14}$$

where TP (true positive), FN (false negative), and FP (false positive) stand for the number of detected inpainted pixels, undetected inpainted pixels, and wrongly detected untouched pixels, respectively:

$$\begin{aligned} \mathrm{PSNR} &= 10 \times \log_{10} \frac{255^2 * \mathrm{MN}}{\sum_{i=1}^{M} \sum_{j=1}^{N} [B(i,j) - A(i,j)]^2}, \\ \mathrm{MSE} &= \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} [B(i,j) - A(i,j)]^2, \end{aligned} \tag{15}$$

where $A(i, j)$ and $B(i, j)$ are the original image and the inpainted image, respectively.

4.3. Reversibility Analysis. In this section, we show that our privacy protection method is effective during communication or sharing. Meanwhile, our method is fully reversible, which enables data to be extracted when it reaches the recipient side.

In Figure 9, we show five sets of comparisons between the recovered images and the original images. The first two of which are from the Today's Headlines database and the last three from the UCID database. In the prerecovery and embedding image operations, there is no damage or tampering to the regions other than the region to be protected. Therefore, under the condition of having the pixel values and coordinates of the region to be protected, the original images can be recovered.

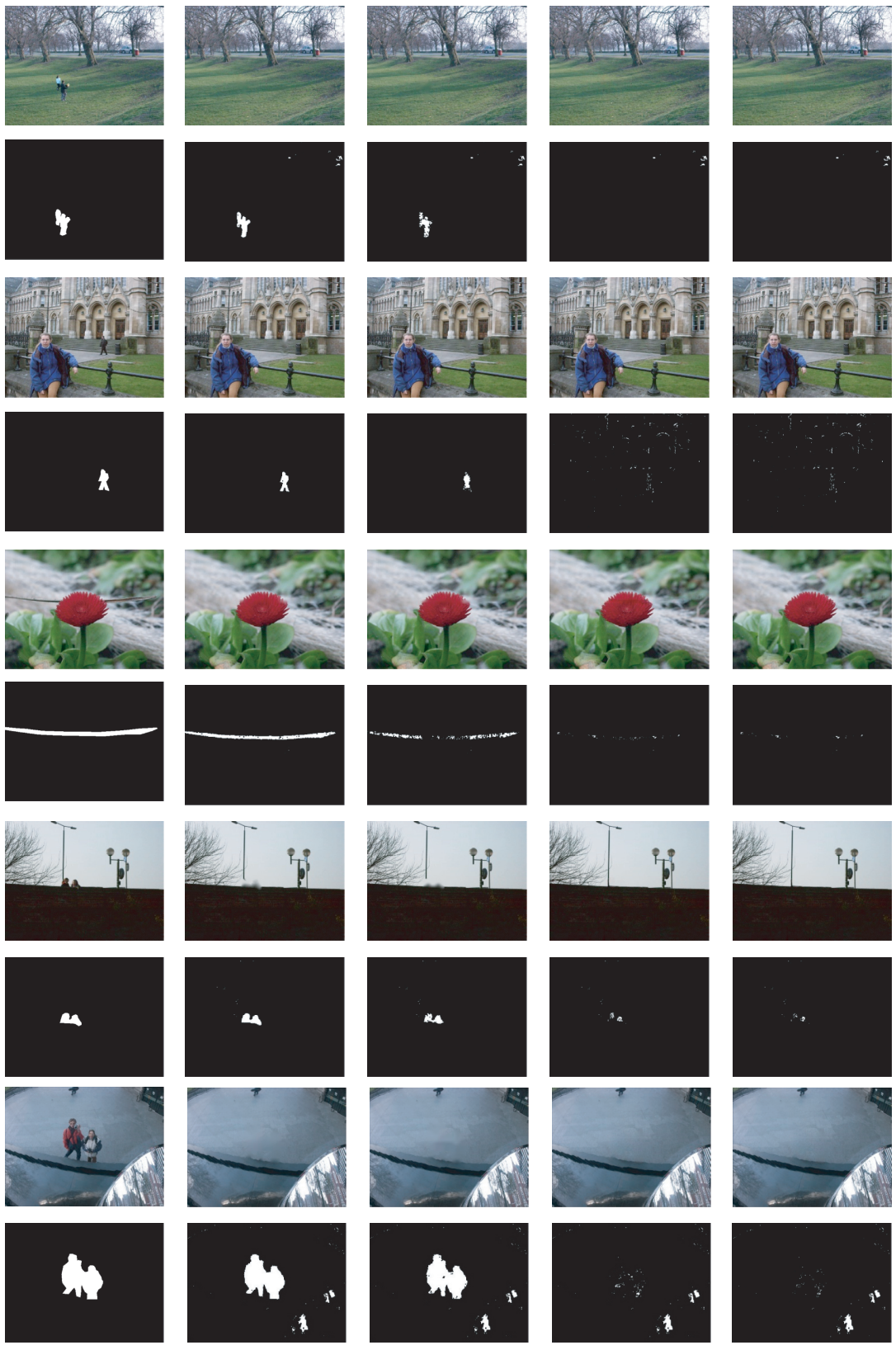


FIGURE 8: Examples from the UCID database. Rows 1, 3, 5, 7, and 9: from left to right, the first image is the original image, and the second to the fifth images represent the inpainted image by references [16, 48] and our method, respectively. Rows 2, 4, 6, 8, 10: from left to right, the first image is ground truth, and the second to the fifth images represent the localization result calculated by forensic algorithm 2.

Table 2: The percentage of protected areas in the whole image (t) and the total number of pixels in the protected area (p), $M1$, $M2$, $M3$, $M4$, and $M5$ represent the four pictures in Figure 8, respectively.

Image	$M1$	$M2$	$M3$	$M4$	$M5$
T	1.13%	0.77%	3.78%	0.87%	6.03%
P	2219	1513	7423	1707	11853

Table 3: F1-scores obtained on the UCID database for different inpainting algorithms and images.

Algorithm	Edge-oriented	Delaunay-oriented	DFNet	Dual-inpainting
$M1$	0.6948	0.8311	0.0045	0.0001
$M2$	0.6521	0.8512	0.0861	0.0040
$M3$	0.5242	0.8486	0.0981	0.0550
$M4$	0.7662	0.8992	0.2808	0.1762
$M5$	0.8945	0.9025	0.1297	0.0572

Figure 9: Examples for reversibility analysis. (a) Original images and (b) recovered images.

5. Conclusion

Currently, most of the privacy protection methods only focus on visual quality, while the real protection needs to be considered from the perspective of image security analysis. We propose a reversible privacy protection scheme using image dual-inpainting and data hiding, in which the original image can be perfectly recovered. Experimental results show that after the inpainting of the image with the removal of the area to be protected by the dual-inpainting algorithm, antiforensics for the two current methods for target removal forensics can be achieved. The later embedding and extraction of the protected region also achieve an effective combination of the two research directions of antiforensics and steganography. In addition, reversible privacy protection not only effectively stops snooping but also guarantees that the original image can be recovered when needed.

Data Availability

In our experiments, we use the free user-shared image dataset provided by Today's Headlines, which contains a large number of people landscapes and various life images. We also use the UCID database.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the Natural Science Foundation of China (U20B2051).

References

[1] L. Yuan and T. Ebrahimi, "Image transmorphing with JPEG," in *Proceedings of the IEEE International Conference On Image Processing (ICIP)*, pp. 3956–3960, Quebec, Canada, September 2015.

[2] J. Yu, Z. Lin, J. Yang, and X. Shen, "Generative image inpainting with contextual attention," in *Proceedings of the CVF Conference on Computer Vision and Pattern Recognition CVPR*, Piscataway, NJ, USA, June 2018.

[3] P. Akyazi and P. Frossard, "Graph-based inpainting of disocclusion holes for zooming in 3d scenes," in *Proceedings of European Signal Processing Conference (EUSIPCO)*, Dublin, Ireland, September 2018.

[4] A. Levin, A. Zomet, and Y. Weiss, "Learning how to inpaint from global image statistics," in *Proceedings of the 9th IEEE*

International Conference on Computer Vision (ICCV), pp. 305–312, Nice, France, October 2003.

[5] M. Bertalmio, A. Bertozzi, and G. Sapiro, "Navier-Stokes, fluid dynamics, and image and video inpainting," in *Proceedings of the Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. 355–362, Kauai, HI, USA, December 2001.

[6] D. Tschumperle and R. Deriche, "Vector-valued image regularization with pdes: a common framework for different applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 4, pp. 506–517, 2005.

[7] M. Ghoniem, Y. Chahir, and A. Elmoataz, "Geometric and Texture Inpainting Based on Discrete Regularization on Graphs," in *Proceedings of the 16th IEEE International Conference on Image Processing (ICIP)*, pp. 1349–1352, IEEE, Caen, France, 2009.

[8] A. Criminisi, P. Pérez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Transactions on Image Processing*, vol. 13, no. 9, pp. 1200–1212, 2002.

[9] S. Korman and S. Avidan, "Coherency sensitive hashing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 6, pp. 1099–1112, 2016.

[10] O. Meur, J. Gautier, and C. Guillemot, "Examplar-based inpainting based on local geometry," in *Proceedings of the 18th IEEE International Conference on Image Processing (ICIP)*, pp. 3401–3404, IEEE, Brussels, Belgium, October 2011.

[11] K. He and J. Sun, "Statistics of patch offsets for image completion," in *Computer Vision–ECCV*, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds., Springer, Berlin, Heidelberg, pp. 16–29, 2012.

[12] P. Buyssens, M. Daisy, D. Tschumperlé, and O Lezoray, "Exemplar-based inpainting: technical review and new heuristics for better geometric reconstructions," *IEEE Transactions on Image Processing: A Publication of the IEEE Signal Processing Society*, vol. 24, no. 6, pp. 1809–1824, 2015.

[13] J. Xie, L. Xu, and E. Chen, "Image denoising and inpainting with deep neural networks," *Advances in Neural Data Processing Systems*, pp. 341–349, 2012.

[14] S. Iizuka, E. Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, p. 107, 2017.

[15] R. Yeh, C. Chen, T. Lim, and A. G. Schwing, M. Hasegawa-Johnson, M. N. Do, Semantic image inpainting with deep generative models," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, pp. 6882–6890, Honolulu, HI, USA, 2017.

[16] X. Hong, P. Xiong, and R. Ji, "Deep fusion network for image completion," 2019, http://arxiv.org/abs/1904.08060.

[17] B. Li, M. Wang, J. Huang, and X. Li, "A new cost function for spatial image steganography," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pp. 4206–4210, Paris, France, October 2014.

[18] G. Liu, F. Reda, K. Shih et al., "Image inpainting for irregular holes using partial convolutions," 2018, http://arxiv.org/abs/1804.07723.

[19] K. Nazeri, E. Ng, T. Joseph, F. Z. Qureshi, and M. Ebrahimi, "Edge connect: generative image inpainting with adversarial edge learning," 2019, http://arxiv.org/abs/1901.00212.

[20] D. Pathak, P. Krähenbühl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: feature learning by inpainting," *Computer Vision and Pattern Recognition (CVPR)*, http://arxiv.org/abs/1604.07379, 2016.

[21] Z. Lin, X. Liu, Q. Huang et al., "Contextual-based image inpainting: Infer, match, and translate," 2018, http://arxiv.org/abs/1711.08590.

[22] C. Luo, W. Zuo, M. Wang, Z. Hu, and H. Zhang, "Semantic image inpainting with progressive generative networks," *ACM Multimedia*, pp. 1937–1947, 2018.

[23] T. Yian, L. Alexander, G. Schwing et al., "Semantic image inpainting with deep generative models," 2018, http://arxiv.org/abs/1607.07539.

[24] M. Bertalmio, G. Sapiro, V. Caselles et al., "Image inpainting," in *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 417–424, ACM Press/AddisonWesley Publishing Co, Minneapolis, MN, USA, July 2000.

[25] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," *ACM Transactions on Graphics*, vol. 22, no. 3, pp. 313–318, 2003.

[26] T. Filler, J. Judas, and J. Fridrich, "Minimizing additive distortion in steganography using syndrome-trellis codes," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 920–935, 2011.

[27] V. Sedighi, R. Cogranne, and J. Fridrich, "Content-adaptive steganography by minimizing statistical detectability," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 2, pp. 221–234, 2016.

[28] S. Kouider, M. Chaumont, and W. Puech, "Adaptive steganography by oracle (ASO)," in *Proceedings of the IEEE International Conference On Multimedia and Expo*, pp. 1–6, San Jose, CA, USA, July 2013.

[29] Y. Pan, J. Ni, and W. Su, "Improved uniform embedding for efficient JPEG steganography," in *Proceedings of the 2016 International Conference on Cloud Computing and Security*, pp. 125–133, Nanjing, China, June, July 2016.

[30] L. J. Guo, J. Q. Ni, and Y. Q. Shi, "Uniform embedding for efficient JPEG steganography," *IEEE Trans. Data Forensics and Security*, vol. 9, no. 5, pp. 814–825, 2014.

[31] Q. Wei, Z. Yin, Z. Wang et al., "Distortion function based on residual blocks for JPEG steganography," *Multimedia Tools and Applications*, vol. 77, no. 14, pp. 17875–17888, 2018.

[32] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868–882, 2012.

[33] V. Holub and J. Fridrich, "Random projections of residuals for digital image steganalysis," *IEEE Transactions on Data Forensics and Security*, vol. 8, no. 12, pp. 1996–2006, 2013.

[34] T. Denemark, V. Sedighi, V. Holub et al., "Selection-channel-aware rich Mmodel for steganalysis of digital images," in *Proceedings of the IEEE International Workshop on data Forensics and Security*, pp. 48–53, Atlanta, GA, USA, December 2014.

[35] V. Holub and J. Fridrich, "Phase-aware projection model for steganalysis of JPEG images," *SPIE, media watermarking, security, and Forensics*, vol. 9409, pp. 94090T–940911, 2015.

[36] M. Chen, V. Sedighi, M. Boroumand et al., "JPEG-phase-aware convolutional neural network for steganalysis of JPEG images," in *in Proceedings of the 5th ACM Workshop On Data Hiding and Multimedia Security*, pp. 75–84, Philadelphia, PA, USA, June 2017.

[37] G. Xu, "Deep convolutional neural network to detect J-UNIWARD," in *Proceedings of the 5th ACM Workshop On Data Hiding and Multimedia Security*, pp. 67–73, Philadelphia, PA, USA, June 2017.

[38] J. Zeng, S. Tan, B. Li et al., "Large-scale JPEG Image steganalysis using hybrid deep-learning framework," *IEEE Trans. data Forensics and Security*, vol. 13, no. 5, pp. 1200–1214, 2018.

[39] B. Li, Z. Li, S. Zhou et al., "New steganalytic features for spatial image steganography based on derivative filters and threshold lbp operator," *IEEE Trans. data Forensics and Security*, vol. 13, no. 5, pp. 1242–1257, 2018.

[40] J. Kodovsky and J. Fridrich, "Steganalysis of JPEG images using rich models," *International Society for Optics and Photonics*, vol. 8303, 2012.

[41] X. F. Song, F. L. Liu, C. F. Yang et al., "Steganalysis of adaptive JPEG steganography using 2D gabor filters," in *Proceedings of the 3rd ACM Workshop on Data Hiding and Multimedia Security*, pp. 15–23, New York, NY, USA, June 2015.

[42] C. Xia, Q. Guan, X. Zhao et al., "Improving GFR steganalysis features by using gabor symmetry and weighted histograms," in *Proceedings of the 5th ACM Workshop on Data Hiding and Multimedia Security*, pp. 55–66, Philadelphia, PA, USA, June 2017.

[43] J. Kodovsky, J. Fridrich, and V. Holub, "Ensemble classifiers for steganalysis of digital media," *IEEE Transactions on Data Forensics and Security*, vol. 7, no. 2, pp. 432–444, 2014.

[44] F. Li, X. Zhang, B. Chen, and G. Feng, "JPEG steganalysis with high-dimensional features and bayesian ensemble classifier," *IEEE Signal Processing Letters*, vol. 20, no. 3, pp. 233–236, 2013.

[45] J. Lukas, J. Fridrich, and M. Goljan, "Detecting digital image forgeries using sensor pattern noise," *Proceedings of SPIE Electronic Imaging*, vol. 6072, pp. 362–372, 2006.

[46] H. Li, W. Luo, and J. Huang, "Localization of diffusion-based inpainting in digital images," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 12, pp. 3050–3064, 2017.

[47] M. Chen, J. Fridrich, and M. Goljan, "Digital imaging sensor identification (further study)," *Proceedings of SPIE Electronic Imaging*, vol. 6505, Article ID 65050P, 2007.

[48] G'MIC, "GREYC's magic for image computing," http://gmic.eu.