Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 7715753, 10 pages
https://doi.org/10.1155/2022/7715753

Research Article

Image Anomaly Detection Based on Adaptive Iteration and Feature Extraction in Edge-Cloud IoT

Weiwei Zhang,[1] **Xinhua Tang**,[2] **and Jiwei Zhang**[3]

[1]*School of Science, Shandong Jianzhu University, Jinan 250101, China*
[2]*School of Cyberspace Security, Shandong University of Political Science and Law, Jinan 250014, China*
[3]*School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications, Beijing 100876, China*

Correspondence should be addressed to Xinhua Tang; 000522@sdupsl.edu.cn

Received 4 November 2021; Revised 13 December 2021; Accepted 30 December 2021; Published 27 January 2022

Academic Editor: Yingjie Wang

Copyright © 2022 Weiwei Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The Internet of Things (IoT) has penetrated into various application fields. If the multimedia information obtained by the IoT device is tampered with, the subsequent information processing will be affected, resulting in an incorrect service and even security threat. Therefore, it is very necessary to study multimedia forensics technology for IoT security. In the edge-cloud IoT environment, an image anomaly detection technology for security service is proposed in this paper. First, preprocessing is performed before image anomaly detection. Then, we extracted sparse features from the image to roughly localize the region of anomaly detection. Feature extraction based on the polar cosine transform (PCT) is then performed only on the candidate region of anomaly detection. To further improve the detection accuracy, we use iterative updating. This method makes use of the feature that the edge node is closer to the multimedia source in physical location and migrates the complex computing task of image anomaly detection from the cloud computing center to the edge node. Provide a security service for abnormal data and deploy it to the edge-cloud server to reduce the pressure on the cloud. Overall, preprocessing improves the ability of feature extraction in smooth or small region of anomaly detections, and the iterative strategy enhances the security service. Experimental results demonstrate that the proposed method outperforms state-of-the-art methods.

1. Introduction

In recent years, with the continuous integration of emerging technologies such as artificial intelligence, blockchain [1], big data [2], and the Internet of Things (IoT) [2–7] and the increasing number of intelligent devices [8], the image data to be processed by the IoT has increased exponentially. IoT technology has penetrated into many fields, and its development has attracted extensive attention. A large number of multimedia data are generated in IoT. If these multimedia data are tampered with, it will threaten the information security and the Internet [9]. Therefore, the research of multimedia forensics is of great significance. Image forensics is an important branch of multimedia forensics. Aiming at the problems of high delay and low processing efficiency of edge cloud, an image anomaly detection method based on edge computing is proposed. Deploy the image security service task to the edge device closest to the image data to be processed to share the computing pressure of the cloud server.

The methods of image anomaly detection [10] can be divided into active methods and passive methods. Active methods are aimed at embedding useful information in an image and then verifying the authenticity and integrity of the image by evaluating the embedded information. However, conventional digital cameras lack digital watermarking functions for security. Consequently, active methods cannot be used when embedded information is unavailable. Alternatively, passive methods, also known as blind forensics, do not require preprocessing of digital images. Thus, it is used to identify the authenticity of images without embedded information, being more applicable than active methods. To conceal tampering and make the image visually more realistic, postprocessing can be applied to the cloned area with methods such as rotation, loss JPEG compression, scaling, and other distortions.

Two main types of passive forensic algorithms are used. One is based on block matching, also known as dense-field algorithm, and the other is based on key points, also known as sparse-field algorithm. Dense-field algorithms usually divide an image into circular or square overlapping blocks to extract a feature vector from each block. After lexicographic sorting, the similarity between the successive vectors is evaluated, and the region of anomaly detection is determined by thresholding. Generally, dense-field algorithms have high computational complexity and may lead to false matching of similar smooth areas in natural images. On the other hand, sparse-field algorithms extract selected points, called key points, to generate feature descriptors. Key points have distinctive characteristics and can reflect essential characteristics of an image to identify target objects. However, sparse-field algorithms cannot extract enough key points from smooth or small areas in images, limiting their performance. In addition, the sparsity of key points impedes the accurate localization of duplicated areas.

To handle the abovementioned problems and leverage both dense-field and sparse-field algorithms, we propose an algorithm integrating these algorithms. First, the region of anomaly detection is roughly localized using a sparse-field algorithm, and then, a dense-field algorithm is applied to accurately determine the region of anomaly detection. Furthermore, we propose an adaptive iterative strategy to improve the localization accuracy. The main contributions of this study are summarized as follows:

(1) In the edge-cloud IoT, an anomaly detection technology for security service is proposed to further construct the trust mechanism of network data. This method makes use of the feature that the edge node is closer to the multimedia source in physical location and migrates the complex computing task of image anomaly detection from the cloud computing center to the edge node.

(2) The advantages of dense-field and sparse-field algorithms are combined in the proposed method. The proposed algorithm first obtains the approximate location of anomaly detection by sparse-field algorithm and then obtains the accurate location of anomaly detection by dense-field algorithm.

(3) An adaptive iterative strategy is introduced to improve the accuracy of tampering localization. Even if few matching points are available, the region of anomaly detection can be accurately determined

The remainder of this paper is organized as follows. Section 2 presents related work. In Section 3, we detail the proposed algorithm. Section 4 reports experimental results. Finally, we draw conclusions in Section 5.

2. Related Work

Edge-cloud calculation in IoT means processing data at the edge of the network. Edge computing may solve the problems of response time requirements, battery life constraints, and bandwidth cost savings and provide data security services [11]. Ferrari et al. used full-cloud and edge-cloud architectures for industrial IoT anomaly detection [12]. The results show that edge domain can reduce data transmission and communication delay. Feature extraction and feature matching are the bases in image anomaly detection [13]. In a dense-field algorithm, detection involves block feature extraction and feature matching across blocks [14]. The discrete cosine transform (DCT) was first proposed by Fridrich et al. [15]. However, the corresponding algorithm has high computational complexity and low robustness. Subsequent improvements to feature extraction measures have been proposed, such as principal component analysis (PCA) [16], singular value decomposition (SVD) [17], discrete wavelet transform (DWT) [18], blur-invariant moment features [19], and local binary patterns (LBP) [20]. Bayram et al. [21] extracted scale-invariant features from each block using the Fourier-Mellin transform (FMT). However, this algorithm is only robust for small region rotations. On the other hand, the Zernike moments (ZM) proposed by Ryu et al. [22, 23] and the polar cosine transform (PCT) proposed by Li [24] allow to extract robust rotation-invariant features from small overlapping blocks. For matching, lexicographic sorting is widely used [25]. To accelerate matching, *k*-dimensional trees [19] and locality-sensitive hashing [24] have been adopted to detect similar patches. However, these algorithms have high computational complexity because all image blocks should be matched. Recently, a fast approximate nearest neighbor search algorithm called Patch Match (PM), which is based on nearest neighbor search, was introduced [26, 27]. Regarding performance, sparse-field algorithms are faster than dense-field algorithms because the former should process fewer points. The scale-invariant feature transform (SIFT) was proposed by Lowe [28] in 1999. Luo et al. [29] extracted rotation and scale invariant descriptors. Subsequently, an accelerated version called speeded up robust features (SURF) was proposed [30]. Other fast feature detection and description algorithms include oriented features from accelerated segment test (FAST) and rotated binary robust independent elementary features (BRIEF) [31], multisupport region order-based gradient histogram [32], and histogram of oriented gradients.

In recent years, blockchain [33] and deep learning have been used for information protection [34, 35]. Fusion strategies based on SIFT have achieved suitable detection results [36–39]. In particular, the histogram of oriented gradients has been applied to feature extraction and tampering detection using a support vector machine (SVM) [36]. Nonoverlapping superpixel segmentation has been used as a preprocessing step before applying feature extraction [37]. Features have been extracted and matched in two different color spaces for rough detection [38], and DCT features have been extracted for accurate localization. Furthermore, key points have been detected using a uniqueness metric and described using PCT [39], with iterative improvement enabling accurate localization. Despite its advantages, SIFT has various drawbacks. Specifically, it cannot detect tampering of smooth or small areas in an image. In addition, the sparsity of feature points provided by SIFT impedes to accurately locate the region of anomaly detection. We propose three strategies to overcome the limitations of this method. First, the target image is represented in the Lab color space in smooth areas. Second, rescaling is applied in small areas. Third, the localization accuracy is improved by combining dense-field and sparse-field algorithms.

3. Proposed Algorithm

IoT technology [40] has penetrated into many fields [41], and its development has attracted extensive attention [42]. Edge cloud is a cloud computing platform built on edge infrastructure based on the core and edge computing capabilities of cloud computing technology to form an elastic cloud platform with comprehensive capabilities in computing, network, storage, and security at the edge. The edge-cloud IoT architecture is shown in Figure 1. We can see that the edge cloud, central cloud, and IoT terminal in Figure 1 form an end-to-end "cloud three-body collaboration" technical framework. By placing tasks such as computing and intelligent data analysis at the edge, cloud pressure can be reduced. The image data generated by massive terminal devices are transmitted to the cloud computing layer [43, 44] for centralized processing through the network, which has the problems of large amount of calculation and large image processing delay. An image anomaly detection method for security service, which is based on edge calculation, is proposed in this paper. Taking advantage of the fact that the edge nodes are closer to the multimedia source in physical location, the complex image analysis and processing computing tasks are migrated from the cloud computing center to the edge computing layer.

We propose an iterative algorithm based on dense-field and sparse-field algorithms in edge-cloud IoT. First, SIFT is applied to roughly locate the region of anomaly detection. Then, PCT feature extraction is performed only on the candidate region of anomaly detection, and PM is used for matching. As SIFT may partially identify a region of anomaly detection, an adaptive iterative strategy is introduced to further improve the localization accuracy. Finally, after morphological operations, the region of anomaly detection is accurately localized.

The flowchart of the proposed algorithm is shown in Figure 2. The algorithm comprises a rough localization stage (including preprocessing) and an accurate localization stage. The following subsections detail each process in the proposed algorithm.

3.1. Image Preprocessing. Firstly, the image is preprocessed. The image analysis process does not need to transmit the image to the cloud through the network for processing but directly analyzes and processes the image in the edge server close to the data source. SIFT is a feature extraction and matching algorithm that provides higher accuracy and robustness to scaling attacks than similar algorithms such as SURF, BRIEF, oriented FAST, and rotated BRIEF. SIFT can extract key points on a spatial scale without being affected by illumination, affine transformations, noise, and other image factors such as corner points, edge points, bright spots in dark areas, and dark spots in bright areas. Based on these key points, feature descriptors of each key point are generated. Owing to its superior performance, we use SIFT for feature extraction in the rough localization stage.

A common preprocessing step before applying SIFT is representing the target RGB (red–green–blue) image in grayscale. However, detection often fails when using grayscale images, especially in smooth areas. To prevent this problem, channels a and b of the Lab color space, the grayscale image, and contrast limited adaptive histogram equalization have been used for preprocessing before feature extraction [38]. Reducing the contrast threshold and rescaling the image have also been used as preprocessing methods [45]. Although such preprocessing methods can increase the number of matching points, they apply various techniques simultaneously, resulting in a large computational overhead. Figure 3 gives an example using SIFT for two preprocessing methods. Figures 3(a)–3(c) gives the tampered image, the tampered image and the ground truth, separately. Figures 3(d) and 3(e) show key points extracted from the grayscale and Lab space (channel a), separately. We can see that the key points in Lab space are denser than those in grayscale. In contrast, after representing the RGB image in Lab color space, tampering could be detected using channel a, as shown in Figure 3(f). The Lab color space allows to extract more key points than the grayscale representation for smooth areas. Nevertheless, the grayscale representation is more robust than the Lab color space against various postprocessing attacks.

In the proposed algorithm, three preprocessing methods are used: (1) RGB-to-grayscale transformation, (2) RGB-to-Lab transformation, and (3) image resizing. However, if these methods are used simultaneously, the computational overhead would notably increase. Therefore, only when one preprocessing method fails, the next method is used, effectively reducing the calculation burden. On the other hand, the proposed algorithm does not require many matching points for rough localization. Thus, if three or more matching points are identified, accurate localization can proceed iteratively. The main preprocessing steps are described as follows.

Step 1. The RGB image is converted into a grayscale image, and feature matching is performed.

Step 2. If the security detection fails, the image is represented in the Lab color space for detection.

Step 3. Otherwise, the image is expanded to repeat detection. If the security detection fails after applying the three preprocessing methods, the image is considered as authentic and safe.

3.2. Rough Localization Stage. Rough localization stage mainly includes three processes: (1) feature extraction, (2) generalized two nearest neighbor matching, and (3) mismatch elimination by using random samples with invariant

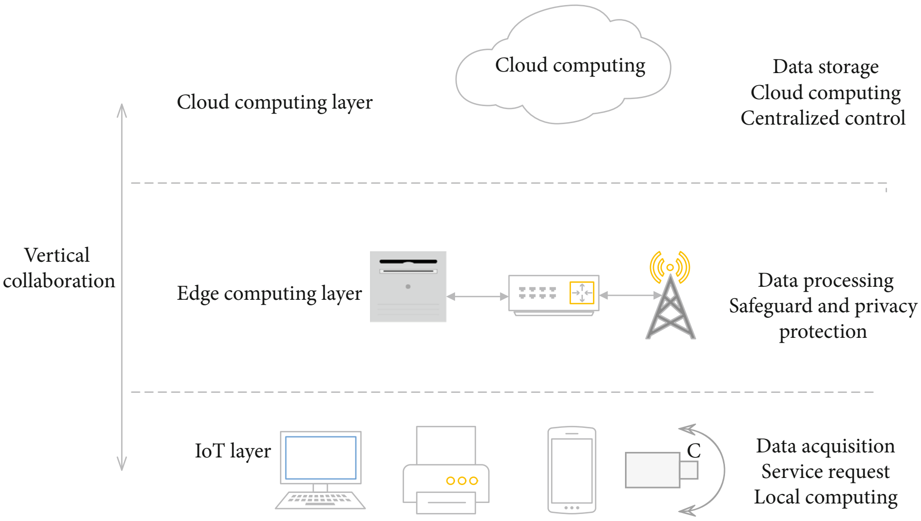


Figure 1: Edge-cloud IoT architecture.

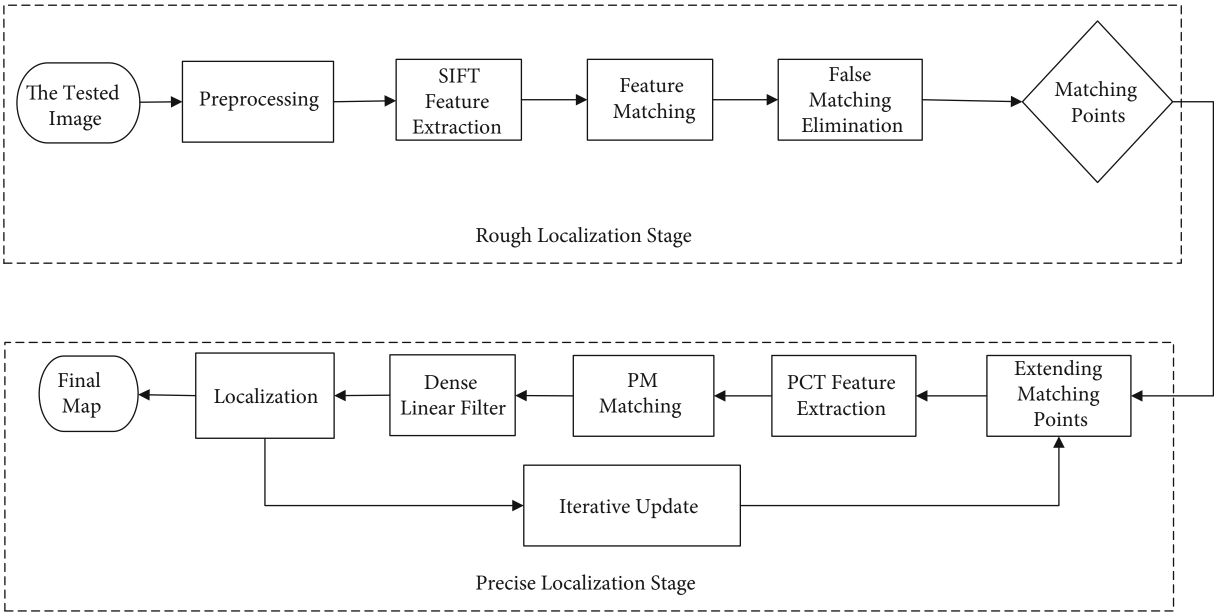
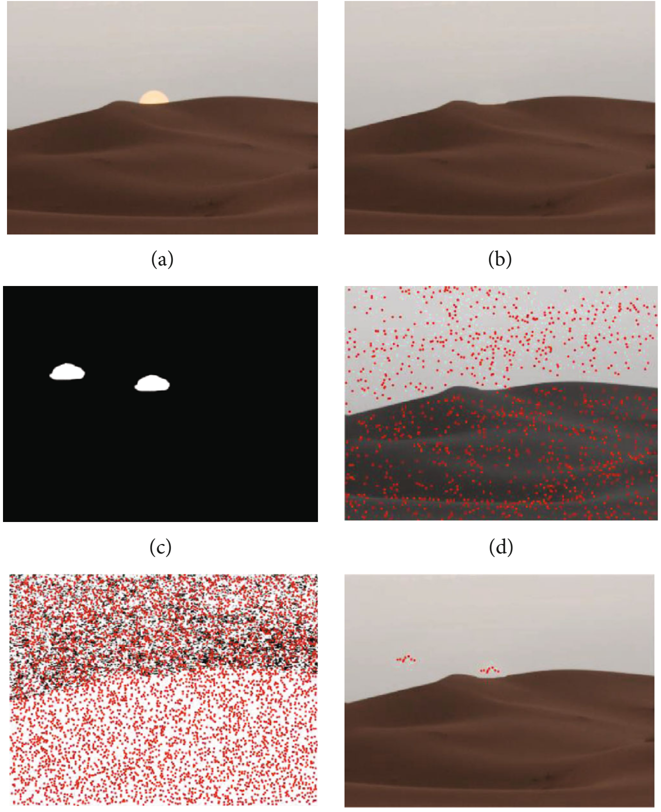


Figure 2: Flowchart of the proposed algorithm.

compatibility. We use the VLFeat open-source library [46] for feature extraction and description. After preprocessing, 128-dimensional SIFT features are extracted. We denote the key points as $x_i(i = 1,\cdots,n)$ and the feature descriptors as $f_i(i = 1,\cdots,n)$ for n feature points. Then, generalized two nearest neighbor matching is applied [47]. The Euclidean distance between a feature descriptor and the other descriptors is calculated. For example, we calculate the distance between f_1 and $f_2, f_3, \cdots, f_n$ and obtain distance vector $D = \{d_1, d_2,\cdots,d_{n-1}\}$ after sorting. If $d_k/d_{k+1} < T_{\text{thresh}}$ and $d_{k+1}/d_{k+2} \geq T_{\text{thresh}}$ for $k(1 \leq k \leq n-2)$, then feature point x_1 and the key points with distances of $\{d_1, d_2,\cdots,d_k\}$ from x_1 are considered to be matching. In this study, we set the threshold T_{thresh} to 0.05.

As many similar areas can appear in natural images, false matching should be prevented. To this end, we use agglomerative hierarchical clustering [47] to filter out classes with less than three points. Furthermore, we use robust random sample consensus to estimate homograph that allows to filter out the effects of unwanted outliers. When at least two classes are detected and at least three matched pairs between classes are available, we consider that the image is tampered.

The sparse-field algorithm can only provide an approximate location of the anomaly detection through the above-mentioned steps. For smooth or small region of anomaly detections, few matched points may be extracted, undermining the accuracy. As shown in Figure 3(f), after rough localization, only eight matching points are obtained, being



(a) (b)

(c) (d)

(e) (f)

Figure 3: Key point extraction and matching for image represented in (a) the original image, (b) the tampered image, (c) the ground truth, (d) SIFT key point detection in grayscale, (e) SIFT key point detection in Lab color space (channel a), and (f) matched key points.

difficult to accurately determine the region of anomaly detection. Therefore, we use a dense-field algorithm and an iterative strategy for accurate localization in the following stage.

3.3. Accurate Localization Stage. To improve the localization accuracy, we use an iterative update strategy as described below.

Step 1. By centering at the matching points, the candidate tampering area (R) is expanded as follows:

$$R(x,y)=\begin{cases}1 & \left\|(x,y)-x_j\right\|\le \dfrac{B}{2}, \\ 0 & \text{otherwise},\end{cases} \quad \forall (x,y)\in I, \tag{1}$$

where $x_j (j=1,\cdots,m)$ represents the matching points obtained during rough localization, I represents the target image, $B=30+[0.1\sqrt{M\times N}]$ is the expansion radius, and $M\times N$ is the size of the target image.

Step 2. Using R, block matching is used for accurate localization. Considering the powerful distinguishing performance of PCT, we use it to extract block features [24]. Specifically, 9-dimensional PCT block features are extracted from expanded matching area R. Let $f(r,\theta)$ denote the polar coordinates of the image. The PCT with order n and repetition l can be expressed as

$$M_{n,l}=\Omega_n\int_0^{2\pi}\int_0^1\left[H_{n,l}(r,\theta)\right]^* f(r,\theta)\,r\,dr\,d\theta, \tag{2}$$

where $H_{n,l}(r,\theta)=\cos\left(\pi n r^2\right)e^{il\theta}$ is the kernel equation of PCT and

$$\Omega_n=\begin{cases}\dfrac{1}{\pi} & n=0, \\ \dfrac{2}{\pi} & n\neq 0.\end{cases} \tag{3}$$

Then, the PCT feature vector can be calculated as

$$f=\left\{\left|M_{n,l}\right| \mid n+l\le 3, 0\le n, l<3\right\}. \tag{4}$$

After PCT block feature extraction, PM [26] and dense linear filtering are applied for matching and filtering out mismatches, respectively. The PM algorithm proposed by Barnes et al. [26] is an approximate nearest neighbor search algorithm. The algorithm searches for similar image blocks globally in a single image through neighborhood search and random sampling. It mainly includes three steps:

```
Input:
$I$: the tested image;
$T_{\text{iter}}$: the maximum number of iteration;
$T_{\text{term}}$: algorithm termination threshold;
Obtaining the candidate tampering area $R(x, y)$ using formula (1);
while $i \leq T_{\text{iter}}$ do
    $\text{map}^{(i)} \longleftarrow R(x, y)$ PCT feature extraction, PM matching and dense linear filtering;
    $\text{Cor\_map}^{(i)} \longleftarrow \text{map}^{(i)}$ corrosion;
    $\text{Exp\_map}^{(i)} \longleftarrow \text{Cor\_map}^{(i)}$ expansion;
    $\text{Dif\_map}^{(i)} \longleftarrow \text{Exp\_map}^{(i)} - \text{Cor\_map}^{(i)} > 0$;
    $\text{map\_new}^{(i)} \longleftarrow \text{Dif\_map}^{(i)}$ PCT feature extraction, PM matching and dense linear filtering;
    $\text{map}^{(i+1)} = \text{map}^{(i)} \cup \text{map\_new}^{(i)}$;
    $\text{map}^{(i+1)}$ morphology open operation;
    if $i \geq 2$ then
        $\nabla\text{map\_}F^{(i)} \longleftarrow \text{diff}(\text{map}^{(1)}, \cdots, \text{map}^{(i)})$//calculate the first derivative;
        if $\nabla\text{map\_}F^{(i)} \leq T_{\text{term}} | i \geq T_{\text{iter}}$ then
            break;
        end if
    end if
    $i = i + 1$;
end while
Output: the tamper localization $\text{map}^{(i)}$.
```

Algorithm 1: The proposed adaptive iterative algorithm.

random initialization, propagation, and random search. Filtering is mainly aimed at finding a dense approximate neighbor matching between image blocks through initialization, propagation, and random search. After this step, we obtain candidate region of anomaly detection $\text{map}^{(i)}$, where i is the number of iterations.

Step 3. To remove isolated small erroneous detections, corrosion is applied to $\text{map}^{(i)}$ with radius B, obtaining area $\text{Cor\_map}^{(i)}$ after corrosion.

Step 4. $\text{Cor\_map}^{(i)}$ is expanded with radius $(B + 10)$, obtaining area $\text{Exp\_map}^{(i)}$.

Step 5. Map $\text{dif\_map}^{(i)}$ is obtained as $(\text{Exp\_map}^{(i)} - \text{Cor\_map}^{(i)} > 0)$. The algorithm returns to Step 2 to obtain $\text{map\_new}^{(i)}$. Except for the first iteration, PCT feature matching is applied only to new area $\text{dif\_map}^{(i)}$ during any other iteration.

Step 6. The candidate region of anomaly detection is updated as $\text{map}^{(i+1)} = \text{map}^{(i)} \cup \text{map\_new}^{(i)}$ $(i \geq 1)$.

Step 7. The morphology open operation is applied to delete objects with area below T in $\text{map}^{(i+1)}$. In this study, we used eight neighborhoods and a minimum clone size T of 1200.

Step 8. The candidate region of anomaly detections obtained over iterations is denoted as $\text{map\_}F^i = \{\text{map}^{(1)}, \cdots, \text{map}^{(i)}\}$ $(i \geq 2)$. Their first derivative is denoted as $\nabla\text{map\_}F^i = \text{diff}(\text{map\_}F^i)$. If $\nabla\text{map\_}F^i(\text{end}) \leq T_{term}$ (set to 500 in this study) or the number of iterations exceeds maximum limit T_{iter} (set to 5 in this study), the algorithm terminates. Otherwise, the algorithm returns to Step 3 to start a new iteration. The pseudocode is shown in Algorithm 1.

Table 1: F_1 score of various anomaly detection methods.

Study	Image level (%)	Pixel level (%)
Amerini et al. [48]	67	44
Li et al. [49]	86	85
Bravo-Solorio and Nandi [25]	94	85
Christlein et al. [50]	67	52
Tahaoglu et al. [38]	94	97
This study	96	97

4. Experimental Results

We evaluated the performance of the proposed algorithm on the GRIP dataset [14]. This dataset contains 80 original images of 768×1024 pixels along with the corresponding copy-move forged images and ground truths. Most of the copies in this dataset are obtained from smooth areas. For the experiments, we used a computer equipped with a 2.60 GHz Intel(R) Core i7-9850H CPU running a MATLAB R2019a implementation.

4.1. Evaluation Criteria. We calculated the precision and recall at the image level and pixel level to evaluate the performance of the proposed anomaly detection algorithm:

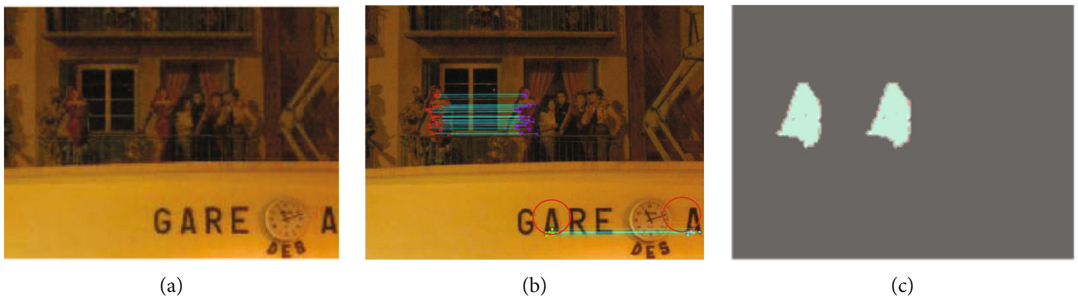


Figure 4: Example of false matching elimination: (a) tampered image, (b) detection results from SIFT, and (c) false matching elimination.

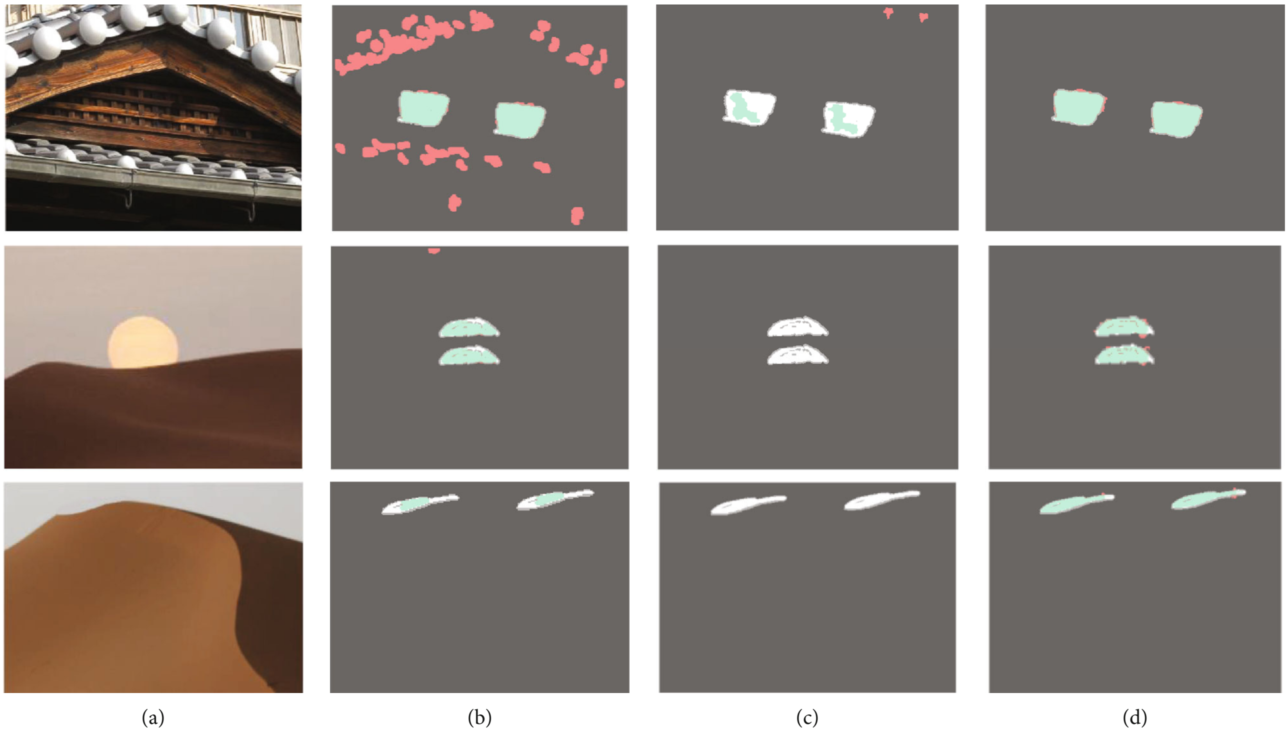


Figure 5: Anomaly detection results of different methods: (a) target image and detection results of (b) PM [27], (c) SIFT [39], and (d) the proposed algorithm.

$$\text{Precision} = \frac{T_p}{T_p + F_p}, \tag{5}$$

$$\text{Recall} = \frac{T_p}{T_p + F_N}, \tag{6}$$

where T_P is the number of tampered images in image level (or tampered pixels in pixel level) correctly detected, F_P is the number of original images in image level (or original pixels in pixel level) erroneously detected as tampered, and F_N is the number of tampered images in image level (or tampered pixels in pixel level) incorrectly detected as authentic.

The precision represents the accuracy of the predicted results, and the recall represents the accuracy of the total positive samples. Thus, higher precision and recall indicate a better algorithm. However, a low recall implies a high precision and vice versa. Thus, we used another comprehensive measure, the F_1 score, obtained as the harmonic mean of the precision and recall:

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{7}$$

4.2. GRIP Dataset. Given an image, we need to determine the presence of tampering, in which case it becomes necessary to accurately localize the region of anomaly detection. We evaluated the proposed algorithm at the image level and pixel level separately. We combined 160 images, including 80 original images and 80 tampered images from the GRIP dataset. At the image level, we obtained precision of 93%, recall of 1, and F_1 score of 96%. At the pixel level, we obtained precision of 95%, recall of 99%, and F_1 score of 97%.

The F_1 score obtained from different methods are listed in Table 1. At the image level, the proposed algorithm provides the highest F_1 score. At the pixel level, the proposed

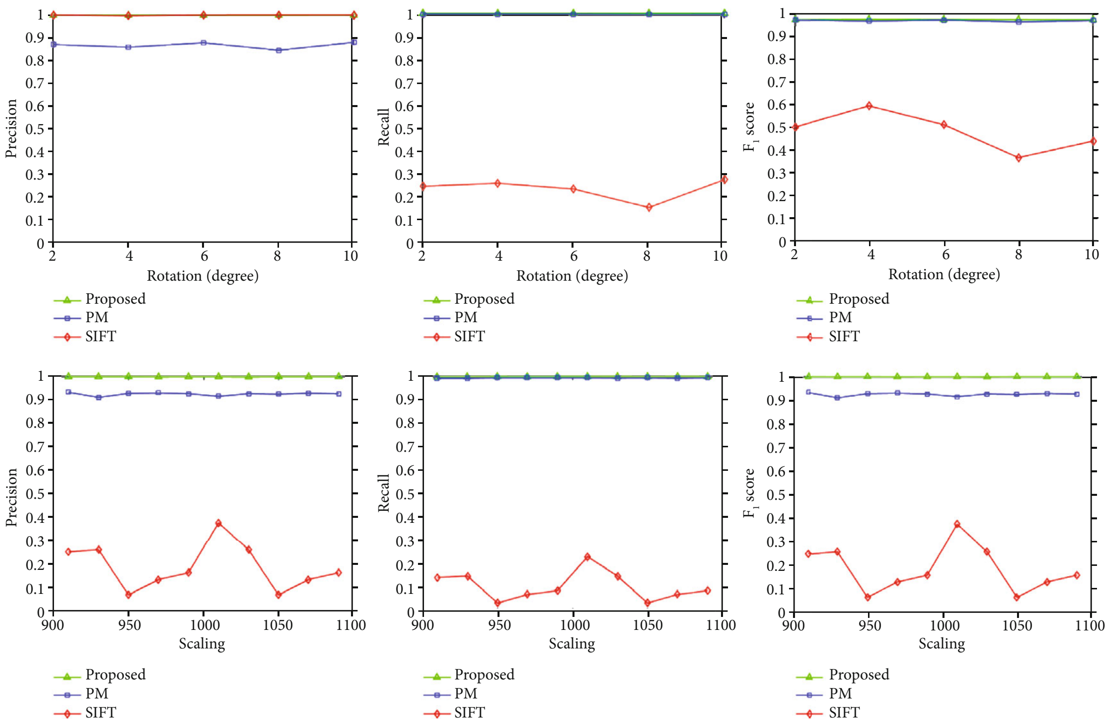


Figure 6: Detection result under rotation and scaling attacks.

algorithm has the same F_1 score as the method in Ref. [38] and higher F_1 score than the other methods. The proposed algorithm provides better detection because mismatched points obtained from rough localization are likely eliminated after accurate localization. Figure 4 shows an example of this situation. We tested the tampered image in Figure 4(a) at the image level. The detection results for SIFT matching are shown in Figure 4(b). The points enclosed by the red circle indicate SIFT mismatching, which is eliminated after PM matching, as shown in Figure 4(c).

Figure 5 shows examples of textured, mixed, and smooth region of anomaly detections. Figure 5(a) shows the forged images, and Figures 5(b)–5 (d) show the corresponding results for PM [27], SIFT [39], and the proposed algorithm, respectively. The red area in the detection result indicates false detection, while the white area indicates that tampering could not be detected, and the green area indicates correct detection. The remaining black areas represent areas that neither have been tampered with nor have been misdetected. The PM algorithm suitably detects tampering in smooth areas (second and third rows), but it provides false detection for the textured area (first row). SIFT fails to accurately localize the region of anomaly detection and is completely unable to detect tampering in the smooth area. In contrast, the proposed algorithm combining SIFT and PCT provides the best detection results.

4.3. FAU Dataset. We also used the public image dataset in Ref. [50] to test the performance of the proposed algorithm under rotation and scaling attacks in smoothed areas. In Figure 6, we tested 15 rotation or scaling images of smoothed area tampering. The first row shows rotation attacks from 2° to 10°, with step of 2°. The second row shows scaling attacks from 91% to 109%, with the step as 2%. We compare the proposed method with the state-of the art method: the SIFT-based method [39], indicated in red, and the PM-based method [27], indicated in blue. The results indicated in green are the detection result of the proposed method. We can see that the proposed scheme performed better than the other two methods in smoothed area tampering.

Table 2: Mean computation time of various anomaly detection methods.

Study	Mean computation time (s)
Tahaoglu et al. [38]	418
Zandi et al. [39]	437
This study	653

The computation time of the proposed algorithm and similar methods is listed in Table 2. By calculating 160 images in the GRIP dataset, the mean computation time of the proposed algorithm is slightly higher than that of the methods in Refs. [38, 39], but it remains within an acceptable range.

5. Conclusions

At present, in the image anomaly detection task of IoT, a large number of terminal devices transmit images to the cloud computing center through the network, resulting in large computing load and high image processing delay. In the edge-cloud IoT, a security service-oriented image anomaly detection technology is proposed in this paper. The RGB image is represented in grayscale and channel *a* of the Lab color space, and it is resized for preprocessing. Then, SIFT feature extraction is applied. The preprocessing methods are not performed simultaneously, but each method is applied only if the preceding one cannot detect tampering, effectively reducing the computational overhead. SIFT feature matching then provides a rough localization of the anomaly detection, while PCT block feature extraction and PM feature matching provide accurate localization of the anomaly detection. An adaptive iterative update strategy is introduced to gradually improve the localization accuracy. The performance of the proposed algorithm was evaluated at the image and pixel levels. The experimental results show that migrating the image security service task to the edge computing device can reduce the pressure of the computing center, deal with the image data anomaly detection in time, and improve the image privacy and security. In the future, deep learning algorithms will be combined to improve the scope of application of image anomaly detection.

Data Availability

The datasets in the experiments include GRIP and FAU, which can be accessed in reference [14, 50], separately. [14]. D. Cozzolino, G. Poggi, and L. Verdoliva, “Copy-move forgery detection based on PatchMatch,” in Proceedings of IEEE international conference of Image Process, pp. 5312-5316, Izmir, Turkey, 2014. [50]. V. Christlein, C. Riess, J. Jordan, C. Riess, and E. Angelopoulou, “An evaluation of popular copy-move forgery detection approaches,” IEEE Transactions on Information Forensics and Security, vol. 7, no. 6, pp. 1841-1854, 2012.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the Doctoral Research Fund of Shandong Jianzhu University (No. X20022Z) and the Shandong Province Soft Science Research Project (Grant no. 2020RKB01671).

References

[1] Y. Xu, J. Ren, Y. Zhang, C. Zhang, B. Shen, and Y. Zhang, “Blockchain empowered arbitrable data auditing scheme for network storage as a service,” *IEEE Transactions on Services Computing*, vol. 13, no. 2, pp. 289–300, 2020.

[2] X. Zhou, W. Liang, K. I. K. Wang et al., “Academic influence aware and multidimensional network analysis for research collaboration navigation based on scholarly big data,” *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 1, pp. 246–257, 2021.

[3] W. Liang, Y. Hu, X. Zhou, Y. Pan, and K. Wang, “Variational few-shot learning for microservice-oriented intrusion detection in distributed industrial IoT,” *IEEE Transactions on Industrial Informatics*, 2021.

[4] X. Zhou, X. Yang, J. Ma, and K. Wang, “Energy efficient smart routing based on link correlation mining for wireless edge computing in IoT,” *IEEE Internet of Things Journal*, vol. 99, 2021.

[5] Z. Cai and Z. He, “Trading private range counting over big IoT data,” in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 144–153, Dallas, TX, USA, 2019.

[6] X. Zhou, X. Xu, W. Liang, Z. Zeng, and Z. Yan, “Deep-learning-enhanced multitarget detection for end-edge-cloud surveillance in smart IoT,” *IEEE Internet of Things Journal*, vol. 8, no. 16, pp. 12588–12596, 2021.

[7] X. Zheng and Z. Cai, “Privacy-preserved data sharing towards multiple parties in industrial IoTs,” *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 968–979, 2020.

[8] Z. Cai, Z. Xiong, H. Xu, P. Wang, W. Li, and Y. Pan, “Generative adversarial networks,” *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–38, 2021.

[9] Z. Cai, Z. He, X. Guan, and Y. Li, “Collective data-sanitization for preventing sensitive information inference attacks in social networks,” *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2018.

[10] X. Zhou, W. Liang, S. Shimizu, J. Ma, and Q. Jin, “Siamese neural network based few-shot learning for anomaly detection in industrial cyber-physical systems,” *IEEE Transactions on Industrial Informatics*, vol. 17, no. 8, pp. 5790–5798, 2021.

[11] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, “Edge computing: vision and challenges,” *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[12] P. Ferrari, S. Rinaldi, E. Sisinni et al., “Performance evaluation of full-cloud and edge-cloud architectures for Industrial IoT anomaly detection based on deep learning,” in *2019 II Workshop on Metrology for Industry 4.0 and IoT (MetroInd4.0&IoT)*, pp. 420–425, Naples, Italy, 2019.

[13] Z. Cai and X. Zheng, “A private and efficient mechanism for data uploading in smart cyber-physical systems,” *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 766–775, 2020.

[14] D. Cozzolino, G. Poggi, and L. Verdoliva, “Copy-move forgery detection based on PatchMatch,” in *2014 IEEE International Conference on Image Processing (ICIP)*, pp. 5312–5316, Izmir, Turkey, 2014.

[15] A. J. Fridrich, B. D. Soukal, and A. J. Lukáš, “Detection of copy-move forgery in digital images,” *Proceedings of Digital Forensic Research Workshop*, , pp. 289–302, Springer-Verlag Press, Berlin, 2003.

[16] A. C. Popescu and H. Farid, *Exposing Digital Forgeries by Detecting Duplicated Image Regions*, Dartmouth Computer Science Technical Report TR2004-515, USA, 2004.

[17] X. Kang and S. Wei, “Identifying tampered regions using singular value decomposition in digital image forensics,” in *International Conference on Computer Science and Software Engineering*, pp. 926–930, Wuhan, China, 2008.

[18] M. Bashar, K. Noda, N. Ohnishi, and K. Mori, “Exploring duplicated regions in natural images,” *IEEE Transactions on Image Processing*, vol. 99, pp. 1–40, 2010.

[19] B. Mahdian and S. Saic, “Detection of copy-move forgery using a method based on blur moment invariants,” *Forensic Science International*, vol. 171, no. 2-3, pp. 180–189, 2007.

[20] Y. Zhu, X. Shen, and H. Chen, "Covert copy-move forgery detection based on color LBP," *Zidonghua Xuebao/Acta Automatica Sinica*, vol. 43, no. 3, pp. 390–397, 2017.

[21] S. Bayram, H. T. Sencar, and N. Memon, "An efficient and robust method for detecting copy-move forgery," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1053–1056, Taipei, Taiwan, 2009.

[22] S. J. Ryu, M. J. Lee, and H. K. Lee, "Detection of copy-rotate-move forgery using Zernike moments," in *International Workshop on Information Hiding*, pp. 51–65, Springer, 2010.

[23] S. J. Ryu, M. Kirchner, M. J. Lee, and H. K. Lee, "Rotation invariant localization of duplicated image regions based on Zernike moments," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 8, pp. 1355–1370, 2013.

[24] Y. Li, "Image copy-move forgery detection based on polar cosine transform and approximate nearest neighbor searching," *Forensic Science International*, vol. 224, no. 1-3, pp. 59–67, 2013.

[25] S. Bravo-Solorio and A. K. Nandi, "Exposing duplicated regions affected by reflection, rotation and scaling," *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 1880–1883, Prague, Czech Republic, 2011.

[26] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, "PatchMatch," *ACM Transactions on Graphics*, vol. 28, no. 3, pp. 1–11, 2009.

[27] D. Cozzolino, G. Poggi, and L. Verdoliva, "Efficient dense-field copy-move forgery detection," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 11, pp. 2284–2297, 2015.

[28] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, pp. 1150–1157, Kerkyra, Greece, 1999.

[29] W. Luo, J. Huang, and G. Qiu, "Robust detection of region-duplication forgery in digital image," in *18th International Conference on Pattern Recognition (ICPR'06)*, pp. 746–749, Hong Kong, 2006.

[30] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: speeded up robust features," in *Computer Vision-ECCV*, pp. 404–417, Springer, Graz, Austria, 2006.

[31] Y. Zhu, X. Shen, and H. Chen, "Copy-move forgery detection based on scaled ORB," *Multimedia Tools and Applications*, vol. 75, no. 6, pp. 3221–3233, 2016.

[32] O. Miksik and K. Mikolajczyk, "Evaluation of local detectors and descriptors for fast feature matching," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pp. 2681–2684, Tsukuba, Japan, 2012.

[33] Y. Xu, C. Zhang, G. Wang, Z. Qin, and Q. Zeng, "A blockchain-enabled deduplicatable data auditing mechanism for network storage services," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 3, pp. 1421–1432, 2021.

[34] X. Yan, B. Cui, Y. Xu, P. Shi, and Z. Wang, "A method of information protection for collaborative deep learning under GAN model attack," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 18, no. 3, pp. 871–881, 2021.

[35] X. Zhou, Y. Li, and W. Liang, "CNN-RNN based intelligent recommendation for online medical pre-diagnosis support," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 18, no. 3, pp. 912–921, 2021.

[36] A. Parashar, A. K. Upadhyay, and K. Gupta, "An effectual classification approach to detect copy-move forgery using support vector machines," *Multimedia Tools and Applications*, vol. 78, no. 20, pp. 29413–29429, 2019.

[37] C. Pun, X. Yuan, and X. Bi, "Image forgery detection using adaptive over-segmentation and feature points matching," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 8, pp. 1705–1716, 2015.

[38] G. Tahaoglu, G. Ulutas, B. Ustubioglu, and V. V. Nabiyev, "Improved copy move forgery detection method via L∗a∗b∗ color space and enhanced localization technique," *Multimedia Tools and Applications*, vol. 80, no. 15, pp. 23419–23456, 2021.

[39] M. Zandi, A. Mahmoudi-Aznaveh, and A. Talebpour, "Iterative copy-move forgery detection based on a new interest point detector," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 11, pp. 2499–2512, 2016.

[40] Y. Xu, J. Ren, G. Wang, C. Zhang, J. Yang, and Y. Zhang, "A blockchain-based nonrepudiation network computing service scheme for industrial IoT," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3632–3641, 2019.

[41] Y. Xu, Z. Liu, C. Zhang, J. Ren, Y. Zhang, and X. Shen, "Blockchain-based trustworthy energy dispatching approach for high renewable energy penetrated power systems," *IEEE Internet of Things Journal*, 2021.

[42] X. Yan, Y. Xu, X. Xing, B. Cui, and T. Guo, "Trustworthy network anomaly detection based on an adaptive learning rate and momentum in IIoT," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 9, pp. 6182–6192, 2020.

[43] C. Zhang, Y. Xu, Y. Hu, J. Wu, J. Ren, and Y. Zhang, "A blockchain-based multi-cloud storage data auditing scheme to locate faults," *IEEE Transactions on Cloud Computing*, 2021.

[44] Y. Xu, Q. Zeng, G. Wang, C. Zhang, J. Ren, and Y. Zhang, "An efficient privacy-enhanced attribute-based access control mechanism," *Concurrency and Computation Practice and Experience*, vol. 32, no. 5, 2020.

[45] Y. Li and J. Zhou, "Fast and effective image copy-move forgery detection via hierarchical feature point matching," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 5, pp. 1307–1322, 2019.

[46] A. Vedaldi and B. Fulkerson, "Vlfeat: an open and portable library of computer vision algorithms," in *International Conference on Multimedia*, pp. 1469–1472, Firenze, Italy, 2010.

[47] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, and G. Serra, "A SIFT-based forensic method for copy-move attack detection and transformation recovery," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 1099–1110, 2011.

[48] I. Amerini, L. Ballan, R. Caldelli, A. del Bimbo, L. del Tongo, and G. Serra, "Copy-move forgery detection and localization by means of robust clustering with J-linkage," *Signal Processing: Image Communication*, vol. 28, no. 6, pp. 659–1669, 2013.

[49] Jian Li, Xiaolong Li, Bin Yang, and Xingming Sun, "Segmentation-based image copy-move forgery detection scheme," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 3, pp. 507–518, 2015.

[50] V. Christlein, C. Riess, J. Jordan, C. Riess, and E. Angelopoulou, "An evaluation of popular copy-move forgery detection approaches," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 6, pp. 1841–1854, 2012.