A Deep Learning Model to Inspect Image Forgery on SURF Keypoints of SLIC Segmented Regions

Diaa Mohammed Uliyan
Department of Information Security, College of Computer Science and Engineering, University of Ha'il, Saudi Arabia
d.uliyan@uoh.edu.sa (corresponding author)

Received: 13 November 2023 | Revised: 25 November 2023 | Accepted: 27 November 2023

 DOI: https://doi.org/10.48084/etasr.6622

ABSTRACT

Copy-Move Forgery (CMF) is a common form of image manipulation attack that involves copying and pasting a part of an image to another position within the same image. This study proposes a Deep Learning (DL) model for detecting CMF, particularly in the presence of various malicious attacks. The proposed approach involves several steps, including converting the input image to grayscale, preprocessing the image using the Simple Linear Iterative Clustering (SLIC) algorithm to generate superpixel partitions, and then extracting keypoint features using the Speeded Up Robust Features (SURF) detector. Finally, a Generative Adversarial Network (GAN) is employed for feature description and matching. To assess the effectiveness of the approach, the types of features used for copy-move forgery were addressed. The proposed approach was examined under rotation, blurring, jpg compression, and scaling attacks. Furthermore, experimental results showed that the proposed approach can detect multiple CMFs with high accuracy. Finally, the proposed method was compared with recent state-of-the-art methods.

Keywords-image forgery; SURF keypoints; deep learning; SLIC segmentation

I. INTRODUCTION

Every user can share and exchange multimedia through social media networks. Multimedia content is exposed to several violations, therefore, the concept of multimedia security includes several ways of protection [1]. The increasing spread of ways to create and modify images, due to the presence of many available relative tools and software, has led to an increase in manipulated images, which can spread deceptive information [2]. Fake news uses multimedia content to mislead people based on fake visual content. Fake news often relies on sensitive or even incorrect images to elicit outrage or other emotional responses from consumers to encourage their spread. For this reason, it is necessary to verify the authenticity of these images, since the images express a lot of major information in many areas such as the field of criminal investigation, insurance dealing, financial reports, intelligent processing, journalism, and medical imaging. It is also critical to detect them because the process of unauthorized image manipulation affects the reading of images and changes the results associated with them. In light of this, many researchers are trying to develop algorithms to detect image forgery [3]. CMF is one of the most prominent image manipulations, as shown in Figure 1.

Digital image forgery is classified into active [4-5] and passive [6] approaches. The active approach means that when an image is created, some information is included within it to achieve the principle of attributing the image to its owner. When processing the image and trying to extract this information, it becomes clear whether the image has been modified or not. The lack of this information when trying to extract it means that the image has been modified. The passive approach is a blind approach to detect if the image has been tampered with or not, based on the characteristics and some features in the image itself without a referenced image. Regarding the passive approach, the image forgeri methods are [7]:

- Copy-move forgery: falsifying images by copying a portion of the image and moving it to another place [8].
- Image splicing: inserting a piece of one or more imported images into the original to deceive the viewer [9]. This is a very common image forgery manipulation.
- Image retouching: applying some enhancement or filtering to an image [10].
- Image resampling: change of geometrical properties in some content of the image, such as rotation, flipping, reflection, expansion, and skewness [11].
- Image morphing: forming a new image that is completely different from two previous images. The necessary effects are made to harmonize the two images with each other [12].

CMF is one of the most important algorithms for detecting image forgery, concerned with finding copies of a portion in

the image and moving it to another place within the same image. The task of coping part of the image and then pasting it to another place is trivial and can be found with the naked eye, but many transformations can be also applied, such as scaling, blurring, etc., to delude the spectator of the image that it is a real image without any manipulation. Figure 2 shows the general approach to the CMF process for the copied region P1 and the moved region P2 in the image itself with translation, rotation, or scaling attacks, respectively.

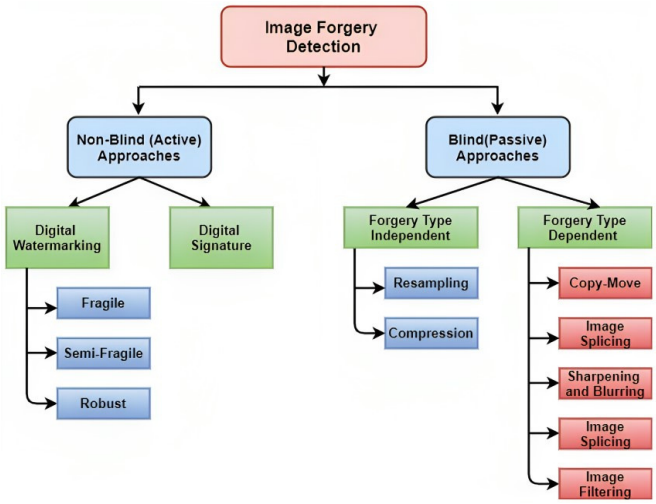


Fig. 1. Classification of security attacks in image forgery detection methods.

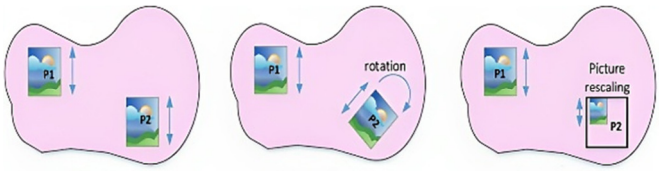


Fig. 2. CMF mechanism.

The main problem is image violation through copy-move forgery by trying to copy an object or hide something in the image. To solve this issue, a DL model was used to detect copy-move forgeries by investigating keypoints in segmented regions. The main objectives of this study were: (1) To accurately determine copy-move forgery regarding malicious attacks, (2) To detect objects or regions regarding primitive features using SURF, (3) To develop a robust detection algorithm against various transformations, such as rotation, scaling, and various jpeg compression factors. The proposed model combines two algorithms, SLIC segmentation and SURF feature extraction, to achieve both accurate localization of the forged regions and effective feature extraction for accurate detection of copy-move forgeries. Some advantages of using SLIC and SURF together are:

- Computational efficiency: The SLIC algorithm can efficiently reduce the number of pixels in a single image, making it easier to compute feature descriptors faster using the SURF algorithm. This helps to minimize the computational complexity and processing time of digital image analysis and processing tasks.
- Robustness: The SURF algorithm [13] was designed to be robust to various image transformations, such as scaling, rotation, and affine distortion.

II. RELATED WORKS

CMFD methods are classified into three categories [14-15]:

- Keypoint-based [16] identifies and chooses regions with high entropy in the image. The process of selecting keypoints starts from the preprocessing step that converts the image to grayscale to facilitate the extraction of primitive features of regions of interest in the image using the local maxima algorithm. Due to the power of the keypoint algorithm in detecting primitive features for any region manipulated with many geometric transformations, such as rotation, resizing, occlusion, and noise, it is widely applied to find the origin of images and recognize objects.
- Block-based [17]: Image regions are divided as overlapping square or circular blocks, and then the feature vector is calculated to characterize a strong and low-dimensional representation of the characteristics of the local image. Then, blocks of similar characteristics are matched according to the extracted descriptors. There is often a Euclidean distance between the descriptors to evaluate the similarity between two blocks.
- Segmentation-based [18-19]: Such methods suggest preprocessing the suspected image by dividing it into semantically independent regions through a segmentation algorithm, where the comparison between them is performed to increase the accuracy of matching similar regions.

Some studies used DL-based methods in the field of CMFD [20]. In [21], the mask region-based convolution neural network was proposed, which is a DL approach to locate and segment modified portions of suspicious images. This method achieved an average precision of 0.769, however, it was insufficient to deal with postprocessing attacks because of its low representation of features' capacity. In [22], a customized CNN based on the VGG-16 model was proposed that used transfer learning to increase CMFD accuracy, but it was computationally expensive and took a long time to derive results. The training time was 2.8 hr while the inference time of an image was 0.0532 s. In [23], deep features, such as deep convolution neural networks, were used to perform boundary-to-pixel direction segmentation using the SD-Net method, although it is vulnerable to noise attacks.

III. PROPOSED METHOD

Figure 3 shows the framework of the proposed CMFD. It consists of several processes, beginning with grayscale conversion of the suspicious image and preprocessing it into a collection of superpixels using SLIC segmentation. Then, it extracts the SURF features from each segment and matches them. Finally, GAN is used to filter out the real image. The proposed method aims to accurately detect copy-move forgery.

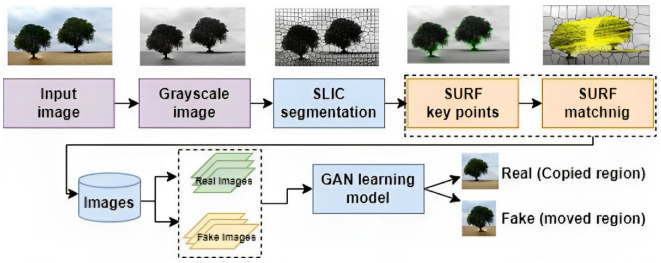


Fig. 3. Framework of the proposed CMFD.

A. *Preprocessing Stage using SLIC Segmentation*

Simple Linear Iterative Clustering (SLIC) is a method used to generate superpixels in an image. Superpixels are defined as groups of intensities that share common features, such as color palettes. The goal of SLIC segmentation is to divide an image into multiple regions with similar superpixels, making it more significant and easier to investigate. SLIC has various practical applications, including object detection and recognition tasks [24]. The SLIC algorithm works by merging pixels based on similar features such as colors. It operates in a five-dimensional color and image plane space, where the combined color and spatial information is used to effectively produce condensed and almost homogeneous superpixels. The algorithm is simple to use, with a lone parameter specifying the intended number of superpixels. Its performance in producing superpixels at a reduced computational cost has been demonstrated to be equivalent to or higher in quality than currently available state-of-the-art methods. Here is how the SLIC segmentation algorithm works:

1. Initialization: The algorithm starts by evenly distributing a set of cluster centers throughout the image plane. The number of cluster centers is determined by the desired number of superpixels.
2. Assignment: Each pixel is then assigned to the nearest cluster center based on its color palettes and spatial closure. The color similarity is measured in a 5-dimensional space, which combines the pixel's color information and its spatial location in the image.
3. Update: After the initial assignment, the cluster centers are updated by taking the mean color and position of all the pixels assigned to them.
4. Convergence: Steps 2 and 3 are repeated until the cluster centers no longer move significantly or a maximum number of iterations is reached. This ensures that the algorithm converges to a stable solution.
5. Postprocessing: Once the algorithm has converged, the final superpixel segmentation is obtained by labeling each pixel with the index of its corresponding cluster center.

The advantages of using the SLIC algorithm for image segmentation are:

- Speed: SLIC is a fast algorithm that can efficiently process large colored images, making it suitable for real-time applications and preprocessing tasks.
- Boundary adherence: SLIC has good boundary adherence, which means that the generated superpixels are closely aligned with the edges and contours of objects in the image.

This can be beneficial for the copy-move forgery detection task that requires accurate object localization and boundary detection. SLIC algorithm, as k-means [25], takes an intended number of regions K as input. The approximate size of each region in an image with N pixels is N/K pixels. Every grid interval $S = \sqrt{N/K}$ would have a region center C for approximately equal-sized regions. The region center C_k is represented as a five-dimensional vector $[R_k, G_k, B_k, x_k, y_k]$ for $k = 1, 2, \ldots, K$.

The first phase involves routinely sampling K region centers C_k on the image plane (x, y) at RGB channels R_k, G_k, B_k and moving them to the places corresponding to the lowest gradient position in a 3×3 neighborhood computed by:

$$G(x,y) = ||I(x+1,y) - I(x-1,y)||^2 + ||I(x,y+1) - I(x,y-1)||^2 \quad (1)$$

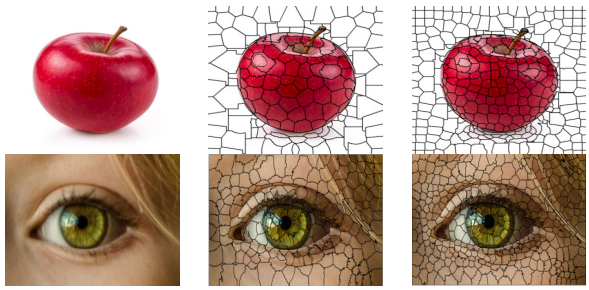
where $I(x, y)$ is the image corresponding to pixel x, y and $\|.\|$ is the L_2 norm. This is done to avoid setting the center on an edge and to decrease the possibility of selecting a noisy pixel. Following that, each picture pixel is given to one of the segments based on distance, i.e. the distance between a pixel i and all the region centers is computed, and the pixel is assigned to the segment whose center has the shortest distance to i. The SLIC algorithm assumes that pixels associated with a region are located on the x, y plane inside a $2S$×$2S$ radius around the region's center. This step narrows the search area. The distance measure D is defined as the sum of the color space d_{RGB} and spatial space $d_{x,y}$ distances normalized by the grid interval S.

$$d_{RGB} = \sqrt{(R_k - R_i)^2 + (G_k - G_i)^2 + (B_k - B_i)^2} \quad (2)$$

$$d_{x,y} = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2} \quad (3)$$

$$D_s = d_{RGB} + \frac{m}{S} * d_{xy} \quad (4)$$

Therefore, the color and spatial distances in (4) are balanced using the variable $m = 10$. A higher number of superpixels will result in smaller superpixels and a more detailed segmentation, while a lower number of superpixels will result in larger superpixels and a more coarse segmentation, as shown in Figure 4.

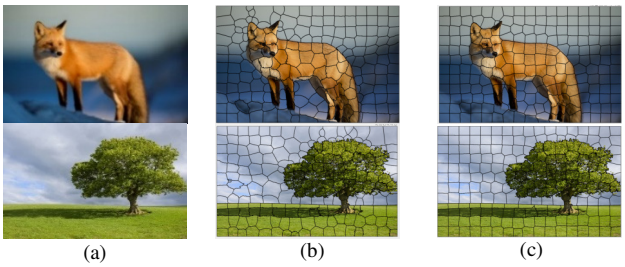


(a) (b) (c)

Fig. 4. Segmentation in low and high superpixel factors: (a) Original image, (b) low superpixel factor, and (c) high superpixel factor.

The output of the SLIC segmentation stage is a set of labeled subregions S_b in image I and the center c_i of each S_b in the i-th region are saved in feature $C = [c_1, c_2, \ldots, c_i]$, where i is the total number of segmented regions and depends on the pixel factor parameter.

The compactness factor controls the trade-off between the spatial distance and the color distance when assigning pixels to superpixels. As shown in Figure 5, from left to right, a lower compactness factor results in superpixels with more irregular subregions, while a larger compactness factor will result in superpixels with regular subregions, making it easier to locate the center of each subregion.



(a) (b) (c)

Fig. 5. SLIC segmentation in variant compactness factor: (a) Image, (b) compactness factor = 20, and (c) compactness factor = 60.

B. Keypoints Detection using SURF

Speeded-Up Robust Features (SURF) [13] detects and describes features in an image plane and is widely used in the field of computer vision applications, such as object recognition, image registration, and classification. It is an upgraded version of the SIFT algorithm [26] with improved speed. The algorithm produces 64 or 128 features by working in 4 steps: (1) Scale-space extrema detection, (2) keypoint localization, (3) orientation assignment, and (4) keypoint descriptor. The SURF algorithm has two parts, the detector and the descriptor. In the detector part, a Hessian matrix is used because of its detection performance strength, since it does not consume a lot of power for computation and produces accurate results. Given a point $p = (x, y)$ in an image I, the Hessian matrix $H(p, \sigma)$ at the point p and scale σ is defined as:

$$H(p,\sigma) = \begin{bmatrix} Lxx(p,\sigma) & Lxy(p,\sigma) \\ Lyx(p,\sigma) & Lyy(p,\sigma) \end{bmatrix} \quad (5)$$

where $Lxx(p, \sigma)$, $Lxy(p, \sigma)$, $Lyx(p, \sigma)$, and $Lyy(p, \sigma)$ are the second-order Gaussian derivatives of image I at point p and scale σ. SURF additionally selects the scale using the Hessian determinant. The prevailing orientation is estimated by adding all responses within a 60° sliding orientation frame. The Laplacian is replaced by the Difference of Gaussian (DoG) to reduce the cost of computation. The image is then converted to an integral image to improve the speed of executing the box filter. For the descriptor part, SURF uses wavelet response, since it is invariant to lighting variables. The feature descriptor is based on the sum of the Haar wavelet response around the point of interest. A $20s \times 20s$ neighborhood is drawn around the keypoint, where s is the size, and it is divided into four quadrants. Horizontal and vertical wavelet responses are obtained for each subregion, and a vector is constructed as:

$$v = (\sum dx, \sum dy, \sum |dx|, \sum |dx|) \quad (6)$$

The wavelet response can be easily determined using an integral image at any scale σ, and is also invariant to the scale variable by converting the descriptor into a unit vector. Figures 6 and 7 show how features are extracted using the SURF algorithm and identify similar regions in the image itself. The result of this stage is to determine the keypoints of the segmented regions S_b that start from the center c_i of each S_b.

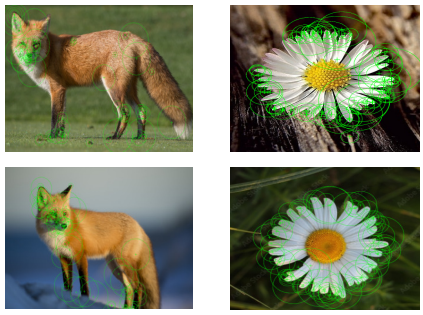


Fig. 6. Detection of SURF keypoint features.

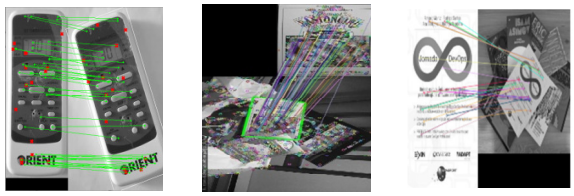


Fig. 7. Locate similar keypoints and matching between image regions.

C. Deep Learning Model using GAN

The Generative Adversarial Network (GAN) [27] is one of the most significant recent advancements in the field of unsupervised deep generative models. Figure 8 depicts the architecture of a typical GAN, which consists of two main neural networks motivated by the two-player min-max game: a generator and a discriminator. The generator attempts to create genuine images to trick the discriminator, while the discriminator attempts to discern between authentic and fake images. The matched regions created by SURF were used as input to the GAN to determine which region was copied (Real) and which was moved (Fake). The copied regions obtained from the discriminator are considered authentic images, while the forged regions obtained from the generator are considered forged. The generator can learn the pattern distribution of the training images, and the discriminator can extract the feature to characterize the image as real or fake. The training process of a GAN can be described as a two-player min-max game, where the generator tries to maximize the probability that the discriminator makes a mistake, while the discriminator tries to minimize its error rate. This adversarial relationship between the generator and the discriminator is what gives GAN its name. During training, the generator and discriminator are updated iteratively using back propagation. The generator's weights are updated based on the feedback from the discriminator, while the discriminator's weights are updated based on its performance in classifying real and fake images.

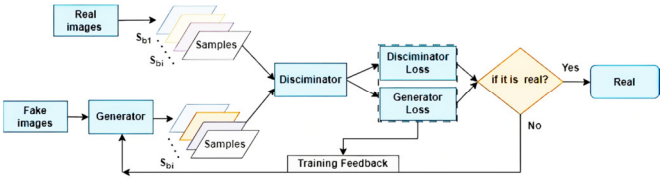


Fig. 8. Steps of GAN to classify real or fake images.

GAN is a learning method that maps an image with a random noise denoted z to an authentic image denoted y by $G:\{x, z\}\rightarrow$. It trains the generator to produce outputs that cannot be distinguished from "real" images by an adversarial discriminator D which is trained to recognize fakes as well as possible. GAN can be expressed by:

$$L_{GAN}(G,D) = E_{x,y}[\log D(x,y)] + E_{x,z}[\log(1 - D(x, G(x,z))] \quad (7)$$

where G attempts to reduce the expectation value and D attempts to achieve it as follows:

$$G^* = Arg_{\min_G \max_D}(\mathcal{L}_{GAN}(G,D)) \quad (8)$$

Combining the GAN expectation value with a more typical loss, such as the $L2$ distance, the discriminator's function persists, but the generator has been assigned to be close to the real output in an $L2$ manner. In addition, the possibility of using the $L1$ distance denoted by $\| \|_1$ rather than the $L2$ was investigated, because $L1$ encourages less distortion.

$$\mathcal{L}_{L1}(G) = E_{x,y,z}[\| y - G(x,z) \|_1] \quad (9)$$

The final G was formulated as:

$$G^* = Arg_{\min G \max D} \; \mathcal{L}_{GAN}(G,D) + \lambda \mathcal{L}_{L1}(G) \quad (10)$$

The network may still learn how to map from x to y without z, but it would give predictable outputs and consequently refuse to identify any pattern other than a λ value.

IV. RESULTS AND DISCUSSION

The functionality of the proposed method was investigated using two benchmarking databases: CoMoFoD [28] and MICC F220 [29]. The CoMoFoD database was used to detect copy-move forgery by analyzing 260 forged JPG or PNG image sets categorized by manipulation and postprocessing methods, including rotation, scaling, JPEG compression, and blurring. Furthermore, the MICC F220 dataset consists of 220 images, 110 of which are forged and 110 are originals.

A. SLIC Segmentation with SURF Keypoints

Superpixels in an image have perceptual meaning when pixels in the same region share similar visual features. These are joined together to provide a compacted depiction of the objects, which is highly beneficial for detecting copy-move forgeries. This SLIC segmentation technique generates superpixels by grouping pixels based on color and spatial closeness at the image level. It was assumed that the copied and moved regions in the forged image share the same color properties, texture attributes, and shape primitives. To address this issue, SLIC was used to address the color and texture aspects of the suspected regions. The SURF method was used to locate primitive features. Figure 9 shows the extracted SLIC with SURF features to observe a copy-move forgery. Figure 9(d) shows how duplicated portions with the same keypoints could be helpful in the detection of copy-move forgeries.

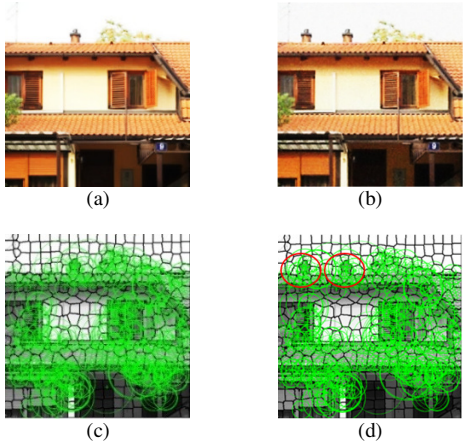


Fig. 9. Detection results of CMF using SLIC+SURF: (a) Original image, (b) forged image, and (c), (d) SLIC+SURF.

B. Matching Keypoints Between Copy-Move Regions using SURF

The green rectangle in Figure 10(a) depicts the object in the original image, whereas the red rectangles in Figure 10(b) represent the copy-move forgeries. SURF features in yellow are appropriately connected between duplicated regions.

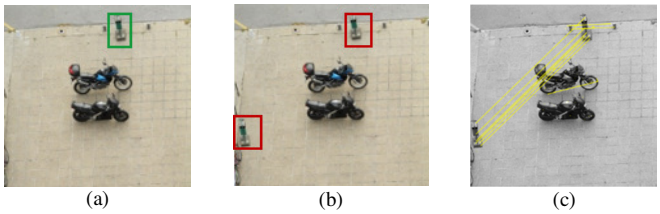


Fig. 10. SURF features from image: (a) Original image, (b) forged image, and (c) SURF keypoints.

C. Detection of CMF under JPEG Compression

JPEG compression with a Quality Factor (*QF*) of 20 and 80 for a copy-move region deforms the forged image. A region was copied and pasted into a non-overlapping region at random spatial locations. Figure 11 shows how the proposed method can detect duplicated regions with the maximum True Positive Rate (TPR) and the lowest False Negative Rate (FNR) for all *QF*s. For a JPEG *QF* of 20, TPR was 96% and FNR was 3% for the worst case. For a JPEG *QF* of 80, TPR was 98% and FNR was 2%.

D. Detection of Multiple CMF under Scaling, Blurring, and Rotation

The proposed method was evaluated in terms of postprocessing attacks such as scaling, blur, and rotation. As shown in Figure 12, GAN reduces false matches and significantly improves the visual detection result in both types of postprocessing-based attacks.

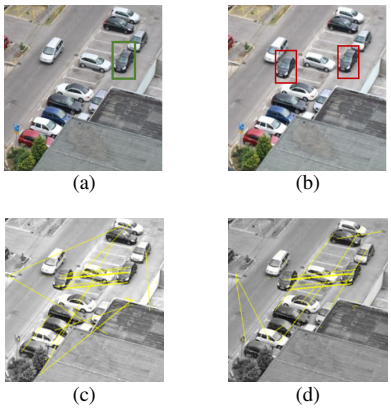


Fig. 11. CMFD results based on high and low QF in JPEG compression: (a) Original image, (b) forged image (QF = 80), (c) detection results with QF = 80, and (d) detection results with QF = 30.

E. Performance Evaluation and Discussion

Table I shows the evaluation results of the proposed method and Table II shows a comparison with three recent state-of-the-art methods: (1) segmentation-based [30-31], (2) deep learning-based [32], and (3) keypoint-based [16]. All implementations were carried out in MATLAB using the image processing toolbox. Its performance was measured in terms of Precision (P), Recall (R), and F1-score (F1), as specified below:

$$\mathrm{P} = \frac{T_p}{T_P + F_P} \times 100\% \quad (11)$$

$$\mathrm{R} = \frac{T_p}{T_P + F_N} \times 100\% \quad (12)$$

$$F_1\text{-score} = 2 \times \frac{\mathrm{P} \times \mathrm{R}}{\mathrm{P} + \mathrm{R}} \times 100\% \quad (13)$$

V. CONCLUSION

This study presented a segmentation concept that exhibits higher discriminative powers compared to a single key point, using SLIC segmentation as a preprocessing technique in image analysis. Additionally, a SURF key point-based method was implemented for each segmented region in the image. This approach is known for its resilience to translation and post-processing attacks. In addition, a GAN framework was used to reduce the number of false positives. The experimental results confirmed that the proposed method exhibited resilience to rotation, blurring, scaling, and jpg compression, and demonstrated its effectiveness compared to current state-of-the-art approaches.

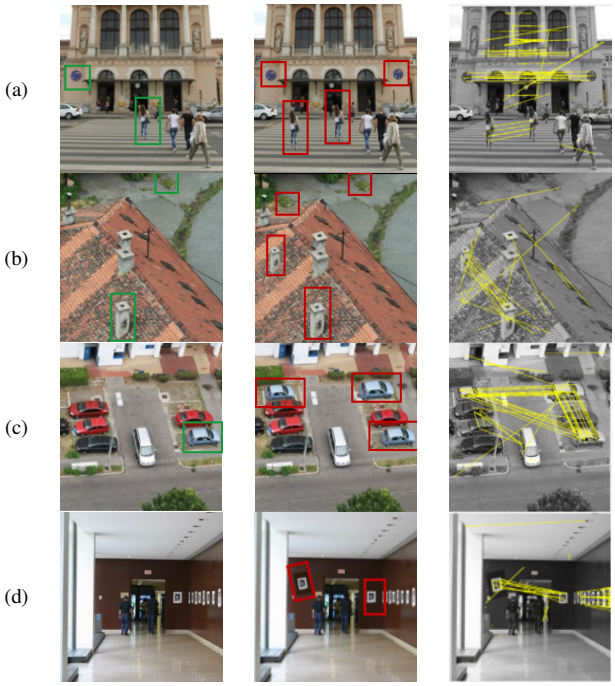


Fig. 12. Visual detection of multiple CMF under scaling, blurring, and rotation attacks: (a) multiple CMF and 0.6 scaling up factor, (b) multiple CMF and 0.3 scaling down factor, (c) multiple CMF and bluring σ = 0.2, and (d) multiple CMF and rotation $\theta = 45°$.

TABLE I. THE PROPOSED METHOD'S PERFORMANCE IN DETECTING CMF IN FORGED IMAGES.

Database	SLIC and SURF without GAN			SLIC+SURF with GAN		
	P (%)	**R (%)**	**F1 (%)**	**P (%)**	**R (%)**	**F1 (%)**
CoMoFoD [28]	91.41	92.88	91.52	95.51	93.21	94.34
MICC F220 [29]	90.21	90.11	90.42	94.12	93.12	93.62

TABLE II. PERFORMANCE COMPARISON WITH OTHER METHODS

Method	**Database**	**Preprocessing step**	**Descriptor model**	**Accuracy**	**Time (sec)**
[30]	CoMoFoD	SLIC segmentation	GLCM	TPR= 93.75 FPR= 7.25 F1=93.75	32.76
[31]	CoMoFoD	DenseNet-41 Segmentation	Mask-RCNN	P=98.12 R= 95.85 F1=96.97	45
[16]	CoMoFoD	SIFT keypoints	Agglomerative Hierarchical Clustering	TPR = 97.65 FPR = 9.02	6120
[32]	CoMoFoD	CNN features	VGG16 & Buster Net DNN	P= 78.22 R= 73.89 F1=75.98	NA
Proposed Method	CoMoFoD	SLIC +SURF	GAN	P=95.51 R=93.21 F1=94.34	40

ACKNOWLEDGMENT

This study was carried out in the College of Computer Science and Engineering at the University of Hail, Kingdom of Saudi Arabia.

REFERENCES

[1] A. Mahfuth, S. Yussof, A. A. Baker, and N. Ali, "A systematic literature review: Information security culture," in *2017 International Conference on Research and Innovation in Information Systems (ICRIIS)*, Langkawi, Malaysia, Jul. 2017, pp. 1–6, https://doi.org/10.1109/ICRIIS.2017.8002442.

[2] H. Wu, J. Zhou, J. Tian, and J. Liu, "Robust Image Forgery Detection over Online Social Network Shared Images," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA, Jun. 2022, pp. 13430–13439, https://doi.org/10.1109/CVPR52688.2022.01308.

[3] M. Ali Qureshi and M. Deriche, "A review on copy move image forgery detection techniques," in *2014 IEEE 11th International Multi-Conference on Systems, Signals & Devices (SSD14)*, Barcelona, Spain, Oct. 2014, pp. 1–5, https://doi.org/10.1109/SSD.2014.6808907.

[4] H. G. Zaini, "Image Segmentation to Secure LSB2 Data Steganography," *Engineering, Technology & Applied Science Research*, vol. 11, no. 1, pp. 6632–6636, Feb. 2021, https://doi.org/10.48084/etasr.3859.

[5] A. Munshi, "Randomly-based Stepwise Multi-Level Distributed Medical Image Steganography," *Engineering, Technology & Applied Science Research*, vol. 13, no. 3, pp. 10922–10930, Jun. 2023, https://doi.org/10.48084/etasr.5935.

[6] S. Gupta and N. Mohan, "Color Channel Characteristics (CCC) for Efficient Digital Image Forensics," *Engineering, Technology & Applied Science Research*, vol. 8, no. 1, pp. 2555–2561, Feb. 2018, https://doi.org/10.48084/etasr.1744.

[7] J. Fridrich, D. Soukal, and J. Lukas, "Detection of copy-move forgery in digital images," in *Proceedings of Digital Forensic Research Workshop*, 2003, vol. 3, no. 2, pp. 652–63.

[8] A. D. Warbhe, R. V. Dharaskar, and V. M. Thakare, "A Survey on Keypoint Based Copy-paste Forgery Detection Techniques," *Procedia Computer Science*, vol. 78, pp. 61–67, Jan. 2016, https://doi.org/10.1016/j.procs.2016.02.011.

[9] B. Xiao, Y. Wei, X. Bi, W. Li, and J. Ma, "Image splicing forgery detection combining coarse to refined convolutional neural network and adaptive clustering," *Information Sciences*, vol. 511, pp. 172–191, Feb. 2020, https://doi.org/10.1016/j.ins.2019.09.038.

[10] S. Sadeghi, S. Dadkhah, H. A. Jalab, G. Mazzola, and D. Uliyan, "State of the art in passive digital image forgery detection: copy-move image forgery," *Pattern Analysis and Applications*, vol. 21, no. 2, pp. 291–306, May 2018, https://doi.org/10.1007/s10044-017-0678-8.

[11] Y. Guo, X. Cao, W. Zhang, and R. Wang, "Fake Colorized Image Detection," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 8, pp. 1932–1944, Dec. 2018, https://doi.org/10.1109/TIFS.2018.2806926.

[12] T. Thakur, K. Singh, and A. Yadav, "Blind Approach for Digital Image Forgery Detection," *International Journal of Computer Applications*, vol. 179, no. 10, pp. 34–42, Jan. 2018, https://doi.org/10.5120/ijca2018916108.

[13] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded Up Robust Features," in *Computer Vision – ECCV 2006*, Graz, Austria, 2006, pp. 404–417, https://doi.org/10.1007/11744023_32.

[14] M. Verma and D. Singh, "Survey on image copy-move forgery detection," *Multimedia Tools and Applications*, Aug. 2023, https://doi.org/10.1007/s11042-023-16455-x.

[15] X. Wang, X. Wang, P. Niu, and H. Yang, "Accurate and robust image copy-move forgery detection using adaptive keypoints and FQGPCET-GLCM feature," *Multimedia Tools and Applications*, May 2023, https://doi.org/10.1007/s11042-023-15499-3.

[16] B. Yang, X. Sun, H. Guo, Z. Xia, and X. Chen, "A copy-move forgery detection method based on CMFD-SIFT," *Multimedia Tools and Applications*, vol. 77, no. 1, pp. 837–855, Jan. 2018, https://doi.org/10.1007/s11042-016-4289-y.

[17] J. Zhong, Y. Gan, J. Young, L. Huang, and P. Lin, "A new block-based method for copy move forgery detection under image geometric transforms," *Multimedia Tools and Applications*, vol. 76, no. 13, pp. 14887–14903, Jul. 2017, https://doi.org/10.1007/s11042-016-4201-9.

[18] S. Tinnathi and G. Sudhavani, "An efficient copy move forgery detection using adaptive watershed segmentation with AGSO and hybrid feature extraction," *Journal of Visual Communication and Image Representation*, vol. 74, Jan. 2021, Art. no. 102966, https://doi.org/10.1016/j.jvcir.2020.102966.

[19] J. Li, X. Li, B. Yang, and X. Sun, "Segmentation-Based Image Copy-Move Forgery Detection Scheme," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 3, pp. 507–518, Mar. 2015, https://doi.org/10.1109/TIFS.2014.2381872.

[20] N. Goel, S. Kaur, and R. Bala, "Dual branch convolutional neural network for copy move forgery detection," *IET Image Processing*, vol. 15, no. 3, pp. 656–665, 2021, https://doi.org/10.1049/ipr2.12051.

[21] X. Wang, H. Wang, S. Niu, and J. Zhang, "Detection and localization of image forgeries using improved mask regional convolutional neural network," *Mathematical Biosciences and Engineering*, vol. 16, no. 5, pp. 4581–4593, 2019, https://doi.org/10.3934/mbe.2019229.

[22] Y. Rodriguez-Ortega, D. M. Ballesteros, and D. Renza, "Copy-Move Forgery Detection (CMFD) Using Deep Learning for Image and Video Forensics," *Journal of Imaging*, vol. 7, no. 3, Mar. 2021, Art. no. 59, https://doi.org/10.3390/jimaging7030059.

[23] Q. Li, C. Wang, X. Zhou, and Z. Qin, "Image copy-move forgery detection and localization based on super-BPD segmentation and DCNN," *Scientific Reports*, vol. 12, no. 1, Sep. 2022, Art. no. 14987, https://doi.org/10.1038/s41598-022-19325-y.

[24] C. Y. Ren and I. Reid, "gSLIC: a real-time implementation of SLIC superpixel segmentation," University of Oxford, UK, Jun. 2011.

[25] A. Likas, N. Vlassis, and J. J. Verbeek, "The global k-means clustering algorithm," *Pattern Recognition*, vol. 36, no. 2, pp. 451–461, Feb. 2003, https://doi.org/10.1016/S0031-3203(02)00060-2.

[26] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004, https://doi.org/10.1023/B:VISI.0000029664.99615.94.

[27] I. Goodfellow *et al.*, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, Oct. 2020, https://doi.org/10.1145/3422622.

[28] D. Tralic, I. Zupancic, S. Grgic, and M. Grgic, "CoMoFoD — New database for copy-move forgery detection," in *Proceedings ELMAR-2013*, Zadar, Croatia, Sep. 2013, pp. 49–54, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6658316.

[29] 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, Sep. 2011, https://doi.org/10.1109/TIFS.2011.2129512.

[30] M. M. A. Alhaidery, A. H. Taherinia, and H. I. Shahadi, "A robust detection and localization technique for copy-move forgery in digital images," *Journal of King Saud University - Computer and Information Sciences*, vol. 35, no. 1, pp. 449–461, Jan. 2023, https://doi.org/10.1016/j.jksuci.2022.12.014.

[31] T. Nazir, M. Nawaz, M. Masood, and A. Javed, "Copy move forgery detection and segmentation using improved mask region-based convolution network (RCNN)," *Applied Soft Computing*, vol. 131, Dec. 2022, Art. no. 109778, https://doi.org/10.1016/j.asoc.2022.109778.

[32] Y. Wu, W. Abd-Almageed, and P. Natarajan, "BusterNet: Detecting Copy-Move Image Forgery with Source/Target Localization," in *Computer Vision – ECCV 2018*, Munich, Germany, 2018, pp. 170–186, https://doi.org/10.1007/978-3-030-01231-1_11.