Computer Systems Science & Engineering
DOI: 10.32604/csse.2023.023747
Article

Tech Science Press

Copy-Move Geometric Tampering Estimation Through Enhanced SIFT Detector Method

J. S. Sujin[1,*] and S. Sophia[2]

[1]Department of Electronics and Communication Engineering, Sri Krishna College of Technology, Coimbatore, 641042, India
[2]Department of Electronics and Communication Engineering, Sri Krishna College of Engineering and Technology, Coimbatore, 641008, India
*Corresponding Author: J. S. Sujin. Email: sujinjsresearch21@outlook.com
Received: 20 September 2021; Accepted: 21 October 2021

Abstract: Digital picture forgery detection has recently become a popular and significant topic in image processing. Due to advancements in image processing and the availability of sophisticated software, picture fabrication may hide evidence and hinder the detection of such criminal cases. The practice of modifying original photographic images to generate a forged image is known as digital image forging. A section of an image is copied and pasted into another part of the same image to hide an item or duplicate particular image elements in copy-move forgery. In order to make the forgeries real and inconspicuous, geometric or post-processing techniques are frequently performed on tampered regions during the tampering process. In Copy-Move forgery detection, the high similarity between the tampered regions and the source regions has become crucial evidence. The most frequent way for detecting copy-move forgeries is to partition the images into overlapping square blocks and utilize Discrete cosine transform (DCT) components as block representations. Due to the high dimensionality of the feature space, Gaussian Radial basis function (RBF) kernel based Principal component analysis (PCA) is used to minimize the dimensionality of the feature vector representation, which improves feature matching efficiency. In this paper, we propose to use a novel enhanced Scale-invariant feature transform (SIFT) detector method called as RootSIFT, combined with the similarity measures to mark the tampered areas in the image. The proposed method outperforms existing state-of-the-art methods in terms of matching time complexity, detection reliability, and forgery location accuracy, according to the experimental results. The F1 score of the proposed method is 92.3% while the literature methods are around 90% on an average.

Keywords: Multi sensor; data fusion; discriminator; orientation; pose; position; mean average precision; recall

1 Introduction

Image forging is the alteration of a digital image in order to conceal some of the image's significant or useful information. It can be difficult to distinguish the modified region from the original image in some circumstances. As image manipulation software is widely available, counterfeiting has become a simple and low-cost method of distorting or concealing facts. This also means that counterfeiting has made this multimedia more vulnerable to alteration. Because of the sophisticated image editing tools available, photographs are susceptible to a variety of modifications; as a result, their authenticity is being called into doubt, particularly where images have persuasive force, such as in a court of law, news stories, or insurance claims.

The demand for authenticity and the integrity of the image drive the detection of a fabricated image. The goal of visual content forgeries is to make modifications in such a way that they are difficult to detect with the naked eye, and then utilize these creations for harmful purposes. Digital image forensics, a fascinating discipline that looks for evidence of forgeries in digital photos, has arisen in recent years [1]. The basic goal of digital image forensics is to look for forgeries in photographs using either active or passive blind techniques. Watermarking and digital signatures, for example, are active approaches that rely on information stored in the images a priori. The lack of information, on the other hand, may limit the use of active techniques in practice. To authenticate the photos, passive procedures are used that do not require any prior knowledge about them.

Image splicing and region duplication by copy-move forgeries are two common methods of image manipulation [2]. To generate a forged image, image splicing uses parts from numerous photographs. It's difficult to tell the difference between tempered and authentic sections because duplicated regions appear to be identical with compatible components. A counterfeiter may also use post processing techniques like blurring, edge smoothing, and noise to hide the visual traces of image forgeries.

A copy-move forgery is the most common type of picture falsification among the numerous types. One or more areas are reproduced at different positions inside the same image in the digital image copy-move forgery. Image forgeries are classified as copy-move forgeries, image splicing, image retouching, lighting condition based forgeries, and so on [3]. A section of an image is copied and pasted into another part of the same image to hide an item or duplicate particular image elements in case of copy-move forgery method. Post-processing on tampered images or photographs, on the other hand, can make detecting instances of fabrication much more difficult. There could be different types of image forgeries as listed in Fig. 1 below:

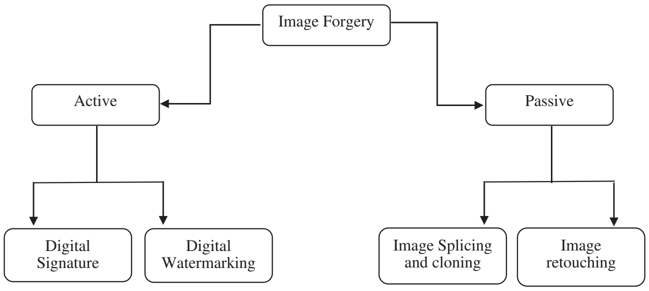

Figure 1: Image forgeries types

Digital watermarking and digital signature are two different sorts of active approaches. To identify copyright, a digital watermark is put to the shot. It is the process of concealing specific data (a series of bits) within a digital image. The author's serial number, company emblem, relevant wording, and so forth are examples of distinctive information [4]. Watermarks can be seen or unseen. Image retouching, image splicing, and copy-move assault are examples of passive approaches. Image retouching is regarded a fabrication of a digital image that is only marginally destructive. Although the original image does not change significantly, certain features of it have been minimized. The splicing method is a form of falsification process that involves combining two or more photos to generate a single image. This is also known as picture composition, and it is where various image processing procedures are carried out [5]. To make counterfeit more plausible, duplicated sections are frequently enlarged, shrunk, or rotated, making it more difficult to spot forgery photos. Passive methods don't require any additional information and can be termed blind in comparison to the original image. These methods rely on extracting some elements from the image under test and making a decision based on pre-defined rules or statistical thresholds.

So far, several forensic detection strategies have been proposed. The available approaches, however, are effective in detecting specific image modifications due to the problem's complexity. It is still necessary to develop a system for identifying any image change. At the moment, the ideal answer is to use a variety of technologies in order to integrate diverse ways to detecting forgeries. SIFT (Scale Invariant Feature Transform) is one such method. The purpose of SIFT is to compare two photographs based on key points, or places of interest [6]. The SIFT descriptors are built to withstand translations, rotations, and zooms. SIFT, on the other hand, shows to be even more resistant to a variety of additional transformations, including changes in viewpoint, noise, blur, contrast shifts, and scene distortion in practice. Fig. 2 shows an example of copy-move forgery.

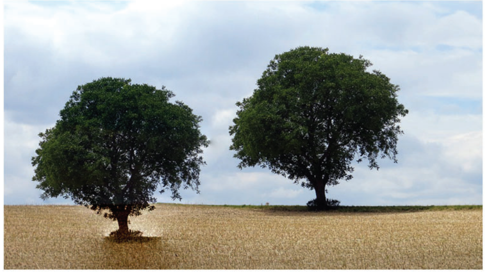

Figure 2: A sample copy-move forgery

In order to make the forgeries real and inconspicuous, geometric or post-processing techniques are frequently performed on tampered regions during the tampering process. The remarkable resemblance between the modified and source regions has become crucial evidence in the detection of forgeries. Though there are a variety of traditional methods for detecting forgery in different ways, there is no single method that can handle all of these situations. Furthermore, present approaches have a significant temporal complexity, particularly during the feature matching stage, and the location of tampered regions is not precise enough to suit practical needs. In actual forensics applications, determining the tampered regions is more significant and vital than forgery detections.

In this work, we propose to use a RootSIFT (root scale-invariant feature transform) method which is an enhanced SIFT descriptor. RootSIFT descriptors are densely extracted as local features, whereas color histograms in HSV space are extracted and quantized as global features. To generate a lower-dimensional descriptor and discriminate the underlying variances of data, the retrieved features are merged and reduced. The visual locally aggregated features (VLAD) technique is used to encode image descriptors. In the coming sections, we detail the proposed method along with the similarity matching techniques to have improved image forensic detection.

Rest of the article consists of 5 chapters: chapter 2 describes survey related to proposed work. chapter 3 discusses the proposed methodology and chapter 4 has result evaluation. Chapter 5 concludes the work with its advantages.

2 Related Work

This section's goal is to provide a complete overview of the state of the art in picture forensics. These techniques were created to determine the origins of a digital image or if the content is legitimate or modified without having any prior knowledge of the image under investigation (and thus are defined as passive). All of these tools work by detecting the existence, absence, or inconsistency of certain traces intrinsically linked to the digital image by the acquisition equipment and any subsequent activity. Multimedia forensics is a branch of forensic science that explores the application of scientific methods to extract probative facts from physical or digital evidence [7–10]. The goal of multimedia forensic tools is to disclose the traces left in multimedia information at each stage of its life cycle by leveraging existing digital imaging and multimedia security research knowledge.

Many research has been carried out in order to comprehend and identify the added artifacts, as well as to assess their impact on the original signal and the end user. The most noticeable ones are probably blocking, ringing, and blurriness [11]. Blockiness distortion is a type of visual distortion in which the underlying block encoding structure appears. The coarse quantization of the spatial frequency components during the encoding process is a common source of blockiness. Artifacts known as ringing distortions arise as false signals (rings) around sharp visual transitions. They look as rings towards the margins when viewed visually. Blurriness is characterized as a loss of spatial features and a drop in edge sharpness in moderate to high frequency portions of an image, such as around scene objects or in rough textured areas.

Each artifacts quantitative appraisal is a difficult process. Several metrics have been created to achieve this goal. Because there is typically more than one artifacts in the image, the intensity is not uniform over the image region, and the mechanism for taking the measurements is still an open topic, an overall solution is still a long way off. All objective picture quality assessment metrics can be categorised based on how much original data is required during the quality assessment. FR metrics are useless in the realm of forensics forgery detection since they need the presence of the original image [12]. In this scenario, a bitwise comparison is enough to generate the map of image changes. In this industry, even RR metrics are challenging to apply. In fact, those metrics necessitate a pre-agreement with the content provider for measuring and adding the required features to the content; additionally, those features must be resilient to changes in data format, transcoding, and so on. The adoption of NR metrics is the most practical solution to our goal.

Any processing applied to digital media is referred to as picture editing. Modifying an image can be done for a variety of purposes, including improving its quality or changing its semantic content [13,14]. In the first scenario, the processed image will include the same information as the original, but in a more usable/pleasant manner. As a result, we call this type of alteration "innocent". In the latter situation, however, the image's semantic content is altered, usually by adding or hiding something. This type of alteration is referred to as "malicious".

The copy-move and cut-and-paste attacks are the most common malicious alterations. Copy-move is the process of copying a section of a picture (of any size or shape) and placing it in a different spot inside the same image. This method is clearly useful when the forger wishes to conceal or duplicate something that is already present in the original image. The other picture forging approach is cut-and-paste, or splicing: starting with two photos, the attacker selects a section of the first and pastes it on the second, usually to change the content and meaning [15]. Splicing is likely more popular than copy-move method, since it is substantially more flexible and allows the production of images with significantly different content than the original.

Various digital picture forensics approaches have been presented to identify digital image counterfeiting. Passive forensics methods are more prevalent than active forensics approaches, which require embedded extra information, due to their broader application situations. Sheng Qu suggested a technique based on the Faster R-CNN network, which has good object detection performance [16]. Their strategy is divided into two parts. Their noise feature extraction channel has been introduced to detect the noise discrepancy between the real and tampering areas. The other original Faster R-CNN network is used to extract features from the original image in order to detect tampering evidence such as large contrast differences and strange tampering boundaries. The authors then employ the bilinear collecting layer to combine the different characteristics from these two channels in order to perceive the picture tampering traces even more efficiently.

The research of Yong Yew Yeap and colleagues focuses on passive forgery detection on altered photos using the copy move approach, often known as (CMFD) Copy Move Forgery Detection [17]. The goal of this research is to provide a CMFD strategy that employs both descriptor and feature matching methods to accurately detect manipulated photos. Aside from the proposed CMFD approaches, the state-of-the-art feature extraction method SURF is also replicated. The feature extraction approach is oriented Features from Accelerated Segment Test and rotated Binary Robust Independent Elementary Features (Oriented FAST and rotated BRIEF), while the feature matching method is 2 Nearest Neighbor (2NN) with Hierarchical Agglomerative Clustering (HAC). The proposed technique was tested on images that were subjected to various geometrical attacks. The evaluation of photos from the MICC-F600 and MICC-F2000 datasets yielded an overall accuracy rate of 84.33 percent and 82.79 percent, respectively. High dimension of feature descriptor, huge amount of computation, and low matching accuracy when the angle of rotation and angle of view are too large are all issues with SURF.

The accessibility of low-cost hardware and software tools makes it simple to create and edit digital photographs without leaving any visible traces [18]. This has resulted in a scenario where the integrity and authenticity of digital photos can no longer be taken for granted. Splicing is the process of copying images from one source to another to create a composite image. Various image editing software's, such as Photoshop and GIMP, are used to do this. As a result, the spliced photos are being scanned and must be authenticated. Image feature extraction-based classifiers such as K Nearest Neigbor (KNN), Fuzzy Logic, and Support Vector Machine (SVM) are utilized to identify spliced images in this case. The Gray Level Co-Occurrence Matrix is used to extract features in the first stage (GLCM).

A blind forensic approach has been presented by Gang Cao and others to detect median filtering (MF), which is widely used for signal denoising and digital image enhancement. The likelihood of zero values on the first order difference map in texture regions can be used to identify MF from other processes [19]. Because anti-forensic strategies like to use MF to assault existing forensics algorithms' linearity assumption, blind detection of the non-linear MF becomes more important. The efficiency of their suggested MF forensics approach is confirmed by both theoretical logic and experimental findings. Most forensics approaches lack a single measurement standard due to the poor performance of typical blind detection and hence not suitable.

Colorization is a new image editing process in which grayscale photographs are colorized with realistic colors. Unfortunately, this approach may be used on purpose to confuse object recognition engines in particular photos. No forensic approach has yet been created to determine whether an image is colorized, to the best of the authors' Yuanfang Guo, Xiaochun Cao, Wei Zhang, and Rui Wang's knowledge. They discovered that colorized photos created by three state-of-the-art algorithms have statistical variations in the hue and saturation channels when compared to natural images [20]. The authors also notice statistical anomalies in the dark and bright channels, which is understandable given that the colorization procedure will eventually change the dark and bright channel values. They suggest two simple but successful detection methods for false colorized images based on their observations: histogram-based fake colorized image detection and feature encoding-based fake colorized image detection. Experiments results show that both proposed methods perform well when compared to a variety of state-of-the-art colorization approaches.

Copy-move forgery is an image tampering technique that involves copying and pasting a region or several regions of an image into one or more parts of the same image. Because images are utilized as the key element of communication and security in many places, detecting image copy-move forgeries is critical. Used images in places of law and order must be original or authentic, and their validity must be ensured. As a result, detecting copy-move forgeries has emerged as a popular and active research topic. The authors of the research, Badal Soni and Debalina Biswas, describe the various block-based methodologies utilized in copy-move forgery detection, as well as their key findings [21]. This paper also examines the benefits and drawbacks of each approach, as well as the many picture datasets utilized for image forgery detection, as well as unsolved concerns and challenges in the field of forgery detection.

Source identification and forgery detection are the two main areas of interest in digital camera picture forensics. Van Lanh et al. [22] give a quick overview of the basic processing stages inside a digital camera before going through numerous approaches for detecting forgeries and identifying source digital cameras. Existing methods for source identification look into the various processing stages inside a digital camera to find clues that can be used to distinguish the source cameras, whereas forgery detection looks for inconsistencies in image quality or the presence of certain characteristics that could indicate tampering.

Oleg Muratov and colleagues present a new use of saliency detection as a tool for forensic image analysis [23]. They propose combining prominent object detection with forensic analysis on the associated output because they argue that salient objects/subjects, which convey the semantic content of the image, are the regions whose integrity is more crucial. To that end, they present an updated version of a saliency map extractor based on segmentation, as well as a description of the use of a tampering detection method for digital composite detection. The outcomes of the experiments are given and debated. The comparison takes place on the entire image in camera-based techniques. Rather, a sliding-window correlation-based technique is used for forgery detection and localization. As the reference image is missing, when a fake area is present, there is a low correlation observed which a demerit in this approach is.

3 SIFT Algorithm for Copy-Move Forgery Detection

Object SIFT key points are extracted and saved in a database after being extracted from a series of reference images. An object in a new image is recognized by comparing each feature in the new image to this database and discovering candidate matching features based on the Euclidean distance between their feature vectors. The goal of SIFT is to compare two images based on important points, or points of interest. The algorithm is split into two parts: detecting Key Points and extracting descriptors from Key Points [24]. The image's scale-space family must first be computed. This family simulates all possible image zooms: it depicts a set of the same image that has become increasingly fuzzy as if the same shot had been taken at various distances from the subject [25]. The first step is to use the input image to create a Gaussian "scale space" function. Convolution (filtering) of the original image with Gaussian functions of varied widths produces this. D(x, y), is determined as the difference between two filtered pictures, one with k multiplied by the scale of the other.:

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \tag{1}$$

We'll use a set of images with varied blur levels and sampling frequencies to apply the scale space framework to a discrete setup. These images, L(x, y, σ), are created by combining Gaussian functions, G(x, y, kσ), with an input image, I. (x, y):

$$L(x,\ y,\ \sigma) = G(x,\ y,\ \sigma) * I(x,\ y) \tag{2}$$

We'll need to evaluate three more extremal images per octave because we'll be computing a difference of scale space images. The extrema of the Laplacian scale space described by (σ, x) are considered to include important points. The following approximation can be derived using the heat Eq. (3)

$$\sigma \Delta v = \frac{\partial v}{\partial \sigma} \approx \frac{v(k\sigma,\ X) - v(\sigma,\ X)}{k\sigma - \sigma} = \frac{w(\sigma,\ X)}{(k-1)\sigma} \tag{3}$$

The process of detecting keypoints is divided into four steps:

1. Computation of the Difference of Gaussian (DoG) by subtracting one Gaussian blurred version of an original image from another which is a less blurred one.
2. The difference between Gaussians can be used to enhance the visibility of edges and other detail in a digital image. Select the discrete DoG extrema from them.
3. The invariant features can be compared to a huge database of known object features from a variety of images and that helps in the fine-tuning of the location of key points.
4. To characterize each key point region, use local image gradients at a specific scale and rotation. This way, the unstable key points are removed.

The DoG operator represents the Gaussian Difference, which can be simply computed as a simple difference. We can determine the discrete extrema of the DoG in a first approximation by considering all the computed difference of Gaussian. This first method has a number of flaws, including noise sensitivity, unreliable keypoints identification, and position and scale accuracy limitations due to the presence of sample grid [26].

The number of DoG extrema can be increased arbitrarily by picture noise. To avoid this, we must delete the keypoints for which the DoG value is less than a certain threshold. As with the DoG computation, this threshold is determined by the number of scales per octave. To boost the speed of the procedure, another method is to execute softer thresholding on the discrete extrema.

The points on the edges are also untrustworthy because they cannot be accurately located: they can be effortlessly translated along the edges to which they belong. By thresholding the Eigen values ratio, the 2D Hessian matrix can be used to find those sites. The normalizing of local gradients based on the orientation reference, i.e., the gradient's primary direction, ensures translation and rotation robustness. Fig. 3 represents the SIFT algorithm in flow chart.

A three-step technique is used to obtain the reference orientation of key points.

1. On a local normalized patch, compute the histogram of local orientation.
2. Smoothing the histogram: three circular convolutions with the same kernel are applied to the histogram.
3. Using the smoothen histogram to extract one or more reference directions: the reference directions are picked from local gradient maxima.

Interpolating the gradient values is required once more to improve the accuracy of the reference orientations. The refined reference orientation can be calculated by maximising the interpolation function. As a result, each local key point might have multiple reference orientations associated with it. As a result, we use more descriptors than key points in practice. The SIFT descriptors are calculated in each

Key points normalized neighbors. In terms of translations, rotations, and zoom-outs, normalization assures invariance. The histogram values of the gradient orientations in the normalized patch make up those descriptors.

$$\theta K = 2\pi \left(K - 1\right)/nbins \tag{4}$$

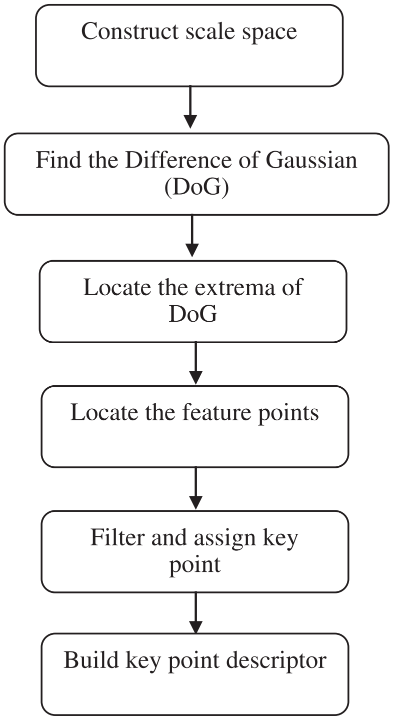
Construct scale space
Find the Difference of Gaussian (DoG)
Locate the extrema of DoG
Locate the feature points
Filter and assign key point
Build key point descriptor

Figure 3: SIFT algorithm

To ensure that each patch is thoroughly incorporated into the image, key points that are too close to the image boundary are eliminated. The angle between the normalized gradient's orientation and the reference orientation determines the remaining key points. The matching procedure is divided into two stages:

1. Finding pairs for each image's Key points: we look at the set L A of key point descriptors and try to locate the pairs.
2. Choosing the Most Reliable Key points: If the distance between two key points is less than C relative times the second closest match, the key points are preserved.

3.1 Challenges with SIFT Approach for Forgery Detection

Although relative thresholding is more efficient than absolute thresholding, it has the disadvantage of failing to identify duplicate copies [27]. While contemporary descriptors like SIFT can discover matches between features with distinct local neighbors, they often overlook the global context when resolving ambiguities that can arise when an image has numerous similar regions.

The goal is to develop a feature descriptor that complements SIFT with a global context vector that includes curvilinear shape information from a much bigger area, reducing mismatches when many local descriptors are comparable.

Other issues include enhancing the detection phase for cloned picture patches with highly uniform texture, when SIFT-like approaches fail to retrieve significant keypoints. Integration with other forensics techniques used locally on flat zones, in particular, must be considered.

Volume changes, rotation, lighting, and point of view have no effect on the SIFT algorithm. It also works well with noise and wide areas of affine conversions, and it can distinguish between points. The SIFT method has a number of drawbacks, including the fact that it is still relatively sluggish, takes a long time to complete, and is ineffective for low-power devices [28]. Hence, we propose to use an enhanced SIFT method with new similarity measures in this work which provides better results than the traditional methods in the literature.

4 Proposed Approach Through Enhanced SIFT Detector Method

When comparing histograms, the Euclidean distance frequently performs worse than the chi-squared distance or Hellinger kernel. RootSIFT is a simple algebraic extension to the SIFT descriptor that allows SIFT descriptors to be “compared” using a Hellinger kernel while still employing the Euclidean distance. The algorithm for extending SIFT to RootSIFT is as follows:

Step 1: The first step is to compute the regular SIFT descriptors as detailed in Section 3 of this work.
Step 2: Normalize each SIFT vector using the L1 method.
Step 3: Take the square root of each SIFT vector element. The vectors are then L2 normalized.

The Euclidean distance between two RootSIFT descriptor vectors h′ and w′ becomes identical to the Hellinger’s distance between the corresponding original SIFT descriptor vectors h and w, defined as:

$$BD_i^h = \frac{\sum_j^s \sum_t^n \left(X_{itj}^h - \bar{X}_{i,j}^h\right)^2}{(n-1)} \tag{5}$$

These metrics produce a number between 0 and 1, with numbers closer to 0 indicating a lesser ‘distance’ and hence a greater similarity. When comparing histograms, of which SIFT descriptors are one example, the Hellinger’s distance is superior to the Euclidean distance. The reason behind this is that when two histograms are matched, the Euclidean distance emphasizes large mistakes on a few bins over tiny errors on the majority of bins, but the Hellinger’s distance does the opposite. In order to avoid this problem, the lower order Manhattan distance might be used instead of the Euclidean distance.

The speeded up robust features (SURF) algorithm, like the SIFT algorithm, searches for the point’s orientation by assigning directions and sizes to each keypoint. It then calculates the description that represents the keypoints proximity. The distance is then calculated based on the descriptors that aren’t on the keypoint locations.

The SURF algorithm is a faster version of the SIFT method. With the same SIFT properties, it is more robust to scale changes. SURF is also a patent-protected algorithm. The SURF algorithm has the following drawbacks: it is not rotationally stable, and it does not operate well with illumination. Considering the speed and robustness as our objective, we propose to use a modified or an enhanced SIFT algorithm that helps to meet both of these challenges as shown in Fig. 4 below.

SIFT and SURF descriptors represent the histogram of oriented gradient (of the Haar wavelet response for SURF) in a neighborhood, hence histogram-based metrics are alternatives to the Euclidean distance. The average response time, the distribution of the best key points (BKP), and the BKP accuracy are all used to evaluate performance. The average reaction time refers to the time it takes for algorithms to recognize and provide findings. The BKP distribution refers to how the generated BKP are distributed across the image’s total surface. The proposed enhanced RootSIFT algorithm snippet is described below with the class RootSIFT as follows:

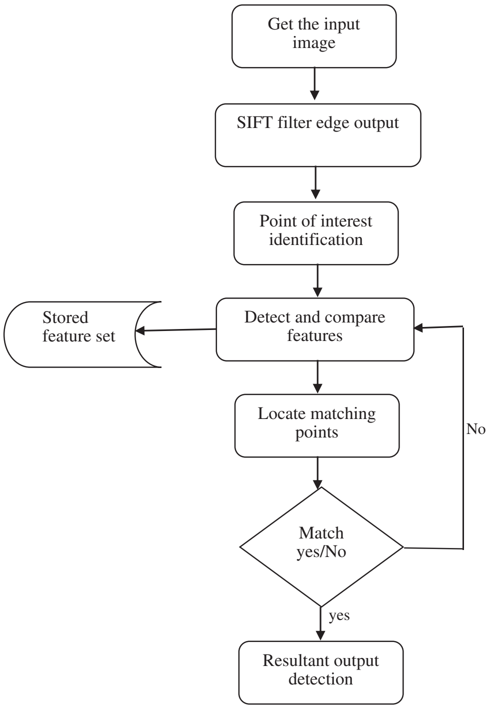

Figure 4: Proposed enhanced SIFT algorithm

Algorithm 1: SIFT

The algorithm works by taking the input image, converting it to gray scale, computing the difference of Gaussian, extracting the SIFT descriptors, finding the RootSIFT values as per the proposed algorithm and then finally return the output values.

```
# Input image that requires descriptor extraction and convert in to grayscale
image_input = cv2.imread(“./images/tree_512.jpg”)
gray_output = cv2.cvtColor(image_input, cv2.COLOR_BGR2GRAY)

# Difference of Gaussian (DoG) keypoints detection from the image
Detector_output = cv2.FeatureDetector_create(“SIFT_Data”)
Kps_value = detector.detect(gray_output)

# SIFT descriptors extraction
Extract_values = cv2.DescriptorExtractor_create(“SIFT_Data”)
(kps_value, descs) = extractor.compute(gray_output, kps_value)
Print “SIFT_Data: kps_value = %d, descriptors = %s” % (len(kps), descs.shape)

# RootSIFT descriptors extraction
rs_output = RootSIFT()
```

(continued)

Algorithm 1: (continued)

```
(kps_value, descs) = rs.compute(gray_output, kps_value)
class RootSIFT:
    def __init__(self):
        # SIFT feature extractor Initialization
        self.extractor = cv2.DescriptorExtractor_create("SIFT_Data")
    def compute_value(self, gray_output, kps_value, eps = 1e−7):
        # SIFT descriptors computation
        (kps_value, descs) = self.extractor.compute(gray_output, kps)
        # if no keypoints or descriptors are found, then return an empty value
        if len(kps_value) == 0:
            return ([], None)
        # Now Hellinger kernel is applied by first L1-normalizing and then taking the
        # square-root of the value
        descs_output /= (descs.sum(axis = 1, keepdims = True) + eps_value)
        descs_output = np.sqrt(descs_output)
        #descs /= (np.linalg.norm(descs, axis = 1, ord = 2) + eps)
        # When keypoint/descriptors are found, return a tuple of the same
    return (kps_value, descs_output)
```

5 Results and Discussion

The Copy-Move Forgery Dataset is provided by the CVIP group [29]. The dataset is broken into many datasets and is made up of medium-sized images (D0, D1, D2). The initial dataset, D0, consists of 50 non-compressed images that have been simply translated. We chose 20 non-compressed images representing simple scenes (single item, simple background) for the other two groups of images (D1, D2), rather than complicated scenarios, because the research goal is in examining the robustness against specific attacks. The similarity between source and destination regions is measured using zero mean normalized cross-correlation. The relationships between the source group and the target group are depicted in Fig. 5 below, when the Helmert transform matrix is used.

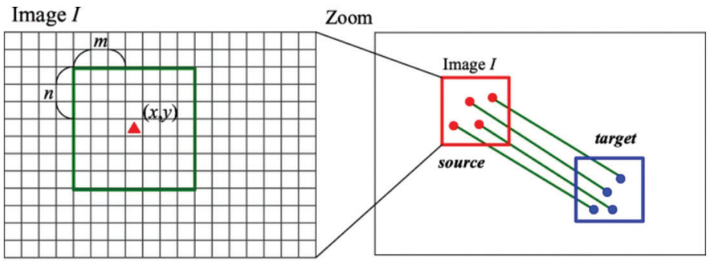

Figure 5: Copy-move transformation from image I to image W

The different parameters used in our setup are listed in Tab. 1 below.

Table 1: Parameter values set in the algorithm for copy-move detection

S.No.	Parameter	Value
1	Threshold for distance calculation	0.5
2	Threshold for making a group	7n where n = Number of key points in a group
3	Rectangle search area	0.125
4	Threshold for the group number of key points	5
5	Pixel size	300

Tab. 2 and Fig. 6 below shows the average response time with the increasing of the number of tested images in the database. When the number of BKP is between 25 and 45, the findings reveal that the suggested augmented SIFT algorithm delivers the fastest average reaction time. We also observed that, as the number of BKP increases further, the proposed SIFT algorithm outperforms the SIFT algorithm while remaining extremely near to the SURF algorithm [30].

Table 2: Average response time in msec for copy-move detection with the proposed algorithm and traditional methods

Method (down)/BKP value (right)	5	15	25	35	45
SIFT algorithm	12.34	13.74	14.02	14.88	15.07
SURF algorithm	4.27	3.87	3.54	3.56	3.99
Proposed algorithm	2.45	3.02	2.90	2.71	2.82

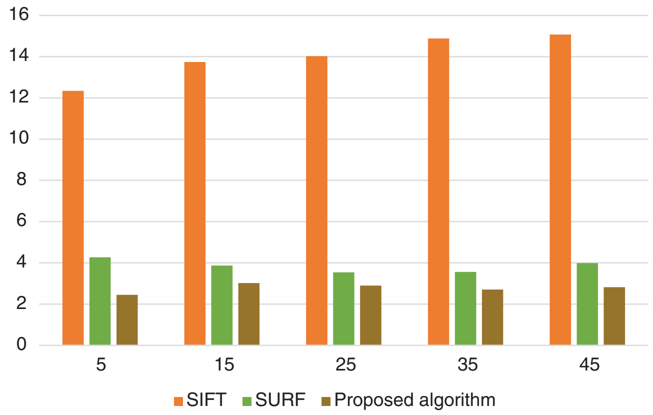

Figure 6: Average response time comparison across methods

The ability of an image hash to identify malicious changes of the incoming image is measured by the tamper detection rate (TDR) Tab. 3. Not only should a good perceptual hash have a higher TDR, but it should also be capable of correctly locating tampered parts. Because the manipulated image may also be subjected to standard image processing, the TDR should be unaffected.

Table 3: Image manipulation and the TDR value observed

Image manipulation	TDR value with the proposed method
Gaussian noise impact	0.913
Speckle noise addition	0.904
Motion Blur detection	0.921
JPEG compression effect	0.951
Image Rotation	0.926
Scaling output	0.905

The received testing image could be authentic or intentionally tampered with, replaced, or manufactured. It could have also been subjected to standard image processing, such as noise reduction, filtering, or geometrical alterations such as rotation and scaling. To analyse the performances, precision and recall metrics are used. The proportion of successfully identified counterfeit images/pixels to all detected images/pixels is known as precision. The proportion of accurately detected forgery images/pixels to all forgery images/pixels is called recall. F1 score is a combination of precision and recall and is defined as follows:

$$F1 = 2 \times (\text{precision} \times \text{recall})/(\text{precision} + \text{recall}) \tag{6}$$

There are 48 high-resolution base images in the dataset, with an average size of 1920 × 1080 pixels. Rotation, scaling, JPEG compression, and extra Noise assaults are all present in all 48 photos. The duplicated snippets are rotated with rotation degrees of 0°, 2°, 4°, 6°, 8°, 10°, 20°, 60°, and 180° in rotation attacks. The comparison of different architectures in terms of F1 score for this dataset is listed in Tab. 4 below.

Table 4: Architecture efficiencies between different methods

Architecture	F1 score
Dense-InceptionNet model	92.5%
AR-Net	88.5%
Base-Atten-P	87.3%
Proposed model	92.65%

We use Gaussian noise (mean is zero, variance is 0.01 and 0.05), blur (window size is 3 but different variances are 0.1 and 0.5), and JPEG recompression (compression quality factors are 30, 60, and 90) to test the resilience of the post-processing operations on the dataset. The results of this experimentation is detailed in Tab. 5 below.

Table 5: Robustness analysis across methods

Operation	F1 score
Noise	45.22%
Blur	43.36%
JPEG	44.45%
Contrast	43.71%

From all these experimental results, it is evident that the proposed method outweighs existing methods in terms of accuracy, speed and robustness.

6 Conclusion

Image forgery detection has clearly been the subject of a lot of research in the past. However, as image alteration technologies get more complex, there is still a need to pay close attention in this subject. The challenge of detecting forgeries can be made much more difficult by applying post-processing techniques to altered images. As the SIFT feature descriptor is invariant to uniform scaling, orientation, illumination changes, and partially invariant to affine distortion, it can reliably recognize objects even among clutter and under partial occlusion. SIFT has the ability to produce large numbers of features that densely cover the image across a wide range of scales and locations. But SIFT comes with limitations which are addressed in this work through enhanced SIFT approach. Our experimental results prove that the proposed method is much better in terms of results, accuracy and robustness as compared to the literature methods. By applying the interest point operator to both images, the cost of a two-dimensional search over the full image is lowered, and the search space is considerably improved. With only a few iterations, the technique converges quickly and may be used on images with a large disparity range. It can withstand a wide variety of disparities. The method is unaffected by 2D image plane rotation or scaling. In order to increase tapering detection, future research should focus on developing more effective hybrid descriptors and matching approaches, such as geometric moments and texture-based features.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] J. A. Redi, W. Taktak and J. -L. Dugelay, “Digital image forensics: A booklet for beginners,” *Multimedia Tools and Applications*, vol. 51, no. 1, pp. 133–162, 2011.

[2] K. Asghar, Z. Habib and M. Hussain, “Copy-move and splicing image forgery detection and localization techniques: A review,” *Australian Journal of Forensic Sciences*, vol. 49, no. 3, pp. 281–307, 2017.

[3] V. Nath, G. Gaharwar and R. Gaharwar, “Comprehensive study of different types image forgeries,” *International Journal of Computer Science and Information Technology*, vol. 6, no. 6, pp. 5413–5416, 2015.

[4] O. O. Evsutin, A. S. Melman and R. V. Meshcheryakov, “Digital steganography and watermarking for digital images: A review of current research directions,” *IEEE Access*, vol. 8, pp. 166589–166611, 2020.

[5] T. Julliand, “Automatic noise-based detection of splicing in digital images,” Université Paris-Est, 2018.

[6] E. Karami, S. Prasad and M. Shehata, “Image matching using SIFT, SURF, BRIEF and ORB: Performance comparison for distorted images,” *ArXiv Preprint ArXiv:1710.02726*, 2017.

[7] M. Olivier, “On a scientific theory of digital forensics,” in *Proc. IFIP ICDF*, Springer, New Delhi, India, pp. 3–24, 2016.

[8] S. Maheswaran, S. Sathesh, P. Priyadharshini and B. Vivek, “Identification of artificially ripened fruits using smart phones,” in *Proc. I2C2*, IEEE, Tamilnadu, India, pp. 1–6, 2017.

[9] M. Stoyanova, Y. Nikoloudakis, S. Panagiotakis, E. Pallis and E. K. Markakis, “A survey on the internet of things (IoT) forensics: Challenges, approaches, and open issues,” *IEEE Communications Surveys & Tutorials*, vol. 22, no. 2, pp. 1191–1221, 2020.

[10] S. Maheswaran, B. K. Paul, M. A. Khalek, S. Chakma, K. Ahmed *et al.,* “Design of tellurite glass based quasi photonic crystal fiber with high nonlinearity,” *Optik*, vol. 181, pp. 185–190, 2019.

[11] Y. Luo, H. Zi, Q. Zhang and X. Kang, “Anti-forensics of jpeg compression using generative adversarial networks,” in *Proc. EUSIPCO*, IEEE, Rome, Italy, pp. 952–956, 2018.

[12] 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.

[13] R. Abdal, Y. Qin and P. Wonka, “Image2stylegan++: How to edit the embedded images?,” in *Proc. CVPR*, IEEE, Seattle, WA, USA, pp. 8296–8305, 2020.

[14] T. -C. Wang, M. -Y. Liu, J. -Y. Zhu, A. Tao, J. Kautz *et al.,* “High-resolution image synthesis and semantic manipulation with conditional gans,” in *Proc. CVPR*, IEEE, Salt Lake City, UT, USA, pp. 8798–8807, 2018.

[15] A. Piva, “An overview on image forensics,” *International Scholarly Research Notices*, vol. 2013, pp. 1–23, 2013.

[16] S. Qu, “An approach based on object detection for image forensics,” in *Proc. IAI, 2019*, IEEE, Shenyang, China, pp. 1–6, 2019.

[17] Y. Y. Yeap, U. Sheikh and A. A. -H. Ab Rahman, “Image forensic for digital image copy move forgery detection,” in *Proc. CSPA, 2018*, IEEE, Penang, Malaysia, pp. 239–244, 2018.

[18] V. Vijayalakshmi, B. Shwetha and S. Sathyanarayana, “Image classifier based digital image forensic detection-a review and simulations,” in *Proc. ICERECT*, IEEE, Mandya, India, pp. 23–28, 2015.

[19] G. Cao, Y. Zhao, R. Ni, L. Yu and H. Tian, “Forensic detection of median filtering in digital images,” in *Proc. ICME*, IEEE, Singapore, pp. 89–94, 2010.

[20] 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, 2018.

[21] B. Soni and D. Biswas, “Image forensic using block-based copy-move forgery detection,” in *Proc. SPIN*, IEEE, Noida, India, pp. 888–893, 2018.

[22] T. Van Lanh, K. -S. Chong, S. Emmanuel and M. S. Kankanhalli, “A survey on digital camera image forensic methods,” in *Proc. ICME*, IEEE, Beijing, China, pp. 16–19, 2007.

[23] O. Muratov, D. -T. Dang-Nguyen, G. Boato and F. G. De Natale, “Saliency detection as a support for image forensics,” in *Proc. ISCCSP*, Rome, Italy, IEEE, pp. 1–5, 2012.

[24] Y. Zhao, Y. Zhai, E. Dubois and S. Wang, “Image matching algorithm based on SIFT using color and exposure information,” *Journal of Systems Engineering and Electronics*, vol. 27, no. 3, pp. 691–699, 2016.

[25] I. R. Otero, “Anatomy of the SIFT method,” École normale supérieure de Cachan-ENS Cachan, 2015.

[26] S. Setumin and S. A. Suandi, “Difference of Gaussian oriented gradient histogram for face sketch to photo matching,” *IEEE Access*, vol. 6, pp. 39344–39352, 2018.

[27] A. D. Warbhe, R. Dharaskar and V. Thakare, “A survey on keypoint based copy-paste forgery detection techniques,” *Procedia Computer Science*, vol. 78, pp. 61–67, 2016.

[28] F. Bellavia and C. Colombo, “Is there anything new to say about SIFT matching?,” *International Journal of Computer Vision*, vol. 128, no. 7, pp. 263–273, 2020.

[29] E. Ardizzone, A. Bruno and G. Mazzola, “Copy–move forgery detection by matching triangles of keypoints,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 10, pp. 2084–2094, 2015.

[30] D. S. Aljutaili, R. A. Almutlaq, S. A. Alharbi and D. M. Ibrahim, “A speeded up robust scale-invariant feature transform currency recognition algorithm,” *International Journal of Computer and Information Sciences*, vol. 12, no. 6, pp. 346–351, 2018.