Multimedia Tools and Applications (2024) 83:87801–87902
https://doi.org/10.1007/s11042-023-17950-x

A review of image features extraction techniques and their applications in image forensic

Dhirendra Kumar[1] · Ramesh Chand Pandey[1] · Ashish Kumar Mishra[1]

Received: 23 August 2022 / Revised: 4 October 2023 / Accepted: 17 December 2023 /
Published online: 20 March 2024

Abstract
In these modern days, digital images become prominent information on the Internet and Social Media. The images can have a number of features with many secrets. To get these secrets, the information regarding the features of the images must be known. An image passes through the pre-processing stage before retrieving these features. For pre-processing, different operations such as normalization, thresholding, noise removal, etc. are applied to get the relevant features of the image. Feature extraction is the process of converting the input image into corresponding image features with the help of some algorithms such as the key point detector algorithm, edge detection algorithm, noise retrieval algorithm, etc. The objective of this survey article is to explore the latest methods for extracting image features and utilizing them in image forensics. These features are applied to detect the different types of image tampering attacks on the image with their detection techniques. Nowadays, the manipulation of an image is a very easy task with the help of different types of tools and software such as adobe photoshop, google picasa, and GNU's Not Unix (GNU) Image Manipulation Program (GIMP), etc. To detect image tampering, features of the image play a very crucial role. A detailed review of different image features that are being utilized in image forensics has been done in the paper. The image features are colors, shape, texture, edges, noise, and key points. The different issues and challenges for detecting image tampering available with the existing techniques along with their performance have been presented in this paper. The future scope of the research work in the area of image processing has also been explored.

Keywords Image features · Image feature extraction · Image tampering attacks · Active forensic techniques · Passive forensic techniques

✉ Dhirendra Kumar
dhirendraphdit@recabn.ac.in

Ramesh Chand Pandey
rameshcse19@gmail.com

Ashish Kumar Mishra
ashish.rcs51@gmail.com

[1] Rajkiya Engineering College (REC), Ambedkar Nagar, UP, India

1 Introduction

In general, feature extraction is part of the image size reduction process in image processing, in which, we perform dimension reduction of an image by dividing the raw data into operational sub-part. Due to reduction less no of resource is required to process the large data. Processing huge data is complex due to more no of variables, and it also increases memory and time complexity. The problem of data overfitting can be a problem in classification algorithms. Extraction of image features is a usual method for constructing combinations of the feature variables for these types of queries. Some of the common visual features of the image are color, shape, texture, etc. Most of the algorithm is based on these features. However, the uses of these features decide the complexity of the algorithm. Different types of image features is represented by Fig. 1.

The remaining parts of the paper are structured as follows. Section 2 elaborates on the most common image feature extraction techniques, with their pros and cons and some modern applications in the different fields. In Section 3, Different types of attacks on digital images have been described. A description of the different types of active attacks is provided in Section 4. It is followed by describing different passive attacks in Section 5. State of art and research methodology is represented in Section 6. Finally, Section 7 Includes References.

2 Image feature extraction

Image feature extraction is a fundamental process in image processing that involves identifying and quantifying meaningful information or patterns within an image. Features are distinctive characteristics of an image that represent specific aspects of its content, such as edges, textures, colors, shapes, or other visual properties. These features serve as a basis for various tasks, including image analysis, recognition, image classification, facial recognition, medical image analysis, and machine learning applications.

2.1 Preprocessing in feature extraction

Image preprocessing is a critical step in image feature extraction (Fig. 2), and its importance cannot be overstated. Image preprocessing involves a series of operations applied to raw images before feature extraction begins. These operations enhance the quality of images, remove noise, correct imperfections, and prepare the data for more effective feature extraction. Here's why image preprocessing is important in the context of image feature extraction:

2.2 Normalization

In the image processing step preprocessing stage is one of the crucial parts. In a preprocessing step, normalization is one of the operations. In image feature extraction normalization is very impactful. Normalization is used for the manipulation of data like scaling up and scaling down which is used for further stages. There are various normalization techniques namely Z-score normalization, Min-Max normalization, and Decimal scaling normalization. With the help of this normalization technique, various other technique has been proposed. In the paper [266] a novel method is proposed based on image normalization to enhance the image recognition performance called the preprocessing process, mainly useful for electronic objects with few distinct recognition characteristics due to functional/material specificity. This paper reflects

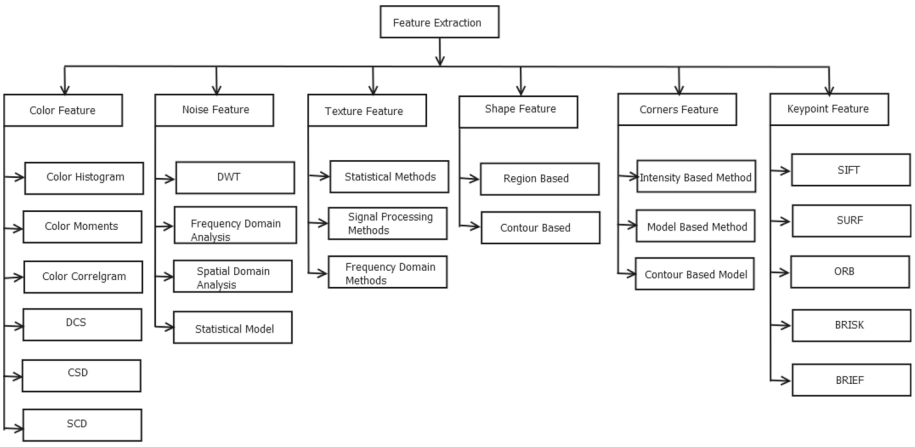

Fig. 1 Types of Image Feature Extraction Techniques

the importance of preprocessing for recognition using deep learning. There are many other paper that plays a very impactful role in feature extraction and object recognition [267–269].

Normalization plays a vital role in image feature extraction, significantly impacting the quality and effectiveness of the extracted features. It involves rescaling or transforming pixel values to ensure that they conform to a particular range or distribution. The impact of normalization in image feature extraction is multifaceted and includes the following aspects.

- Improved Comparability
- Enhanced Robustness
- Improved Convergence
- Enhanced Discriminative Power
- Facilitation of Dimensionality Reduction

In practice, various types of normalization may be applied depending on the specific characteristics of the image data and the requirements of the feature extraction task. Some common normalization techniques include min-max scaling (scaling pixel values to a specific range, e.g., [0, 1]), mean-centering (subtracting the mean intensity value from each pixel), standardization (mean-centering and scaling by the standard deviation), and histogram equalization (adjusting pixel values to achieve a desired histogram distribution). The choice of normalization method should be guided by the nature of the data and the goals of the feature extraction process.

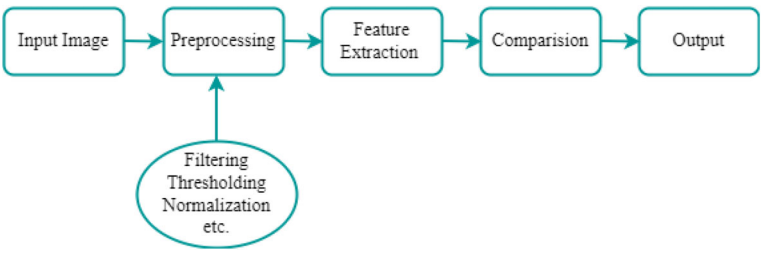

Fig. 2 Basic Diagram of Image Feature Extraction with Preprocessing

2.2.1 Thresholding

Thresholding is a fundamental technique in image processing that plays a significant role in segmenting images, separating objects from backgrounds, and extracting meaningful information from images. Its primary purpose is to simplify and binarize an image by dividing it into regions of interest (foreground) and non-interest (background) based on pixel intensity values.

In the thresholding process, segmentation of a digital image has been done based on a specific characteristic of the pixels. Thresholding is one of the essential tasks in any image processing application. Here we have mainly two types of thresholding techniques global and local. In Global thresholding generally, an image is separated by one threshold whereas a local threshold relies on the local characteristic of sub-images. In the paper, [270] a novel method is proposed for pulmonary nodule segmentation and feature extraction using multilevel thresholding. different types of thresholding techniques (Intermediate Thresholding, Micro-level Thresholding) have been used for various feature extraction. The application of the thresholding technique is explained in different articles [271–275]. Here are some key roles and applications of thresholding in image processing:

- Image Segmentation
- Object Detection
- Foreground-Background Separation
- Image Enhancement
- Noise Reduction
- Morphological Operations
- Image Registration

2.2.2 Noise reduction

Images often contain various types of noise, such as Gaussian noise, salt-and-pepper noise, or speckle noise, which can introduce unwanted variations in pixel values. Preprocessing techniques like filtering and denoising help remove or reduce this noise, leading to more accurate feature extraction.

Noise feature extraction has various stages, Prepossessing is one of them. Which performs the denoising of images. Image de-noising is a very crucial step in image processing for noise feature extraction. The goal of this step is to remove the noise from the image in such a way that the original image is visible. The application of denoising is very crucial in different areas of image processing. It is often used to enhance the features of an image.

Mainly there are two types of denoising techniques

- Linear
- Non-linear

Linear filter To remove certain types of noise we use a linear filtering technique. We reduce image noise by convolving the authentic image with a mask that represents a low-pass filter or smoothing operation. Linear methods are computationally fast but we lost many details of the image.

Non linear filter The output of a non-linear filter is not a linear function. Compared to liner filters Non-linear filters conserve more details of the image. The application of the non-linear

type of filter is to remove the non-additive type of noise. Designing of non-linear filter is more complex than a linear filter.

Types of linear and non-linear filters

Median filter The median filter is based on the order statics filter. It is a simple and powerful non-linear filter, whose reply is based on the ranking of pixel values contained in the filter region.

Mean filter The mean filter is a simple spatial filter, it is also known as an average filter or a box filter. It is a very common noise-filtering technique used in image processing to reduce the effects of noise in an image. The mean filter acts on an image by smoothing it. The mean filter operates by replacing the value of each pixel with the average value of the pixel values in its neighborhood. The primary purpose of a mean filter is to smooth or blur the image, effectively reducing the impact of high-frequency noise while preserving the overall structure and larger features of the image.

Adaptive filter In the context of noise filtering, an adaptive filter is a type of filter that adjusts its filter coefficients or parameters based on the local characteristics of the input image. Two parameter mean and variance are the two statistical measures that a local adaptive filter relies on with a defined mxn window region. Unlike fixed or static filters, which apply the same filter coefficients to the entire image, adaptive filters modify their coefficients dynamically, typically using information from the immediate vicinity of the current sample or pixel being filtered. This adaptability allows adaptive filters to better handle varying noise characteristics and image properties. Compared to linear filters, the adaptive filter is more selective and maintains edges and other high-frequency parts of an image.

Wiener filter Wiener Filter is mainly used to filter out the corrupted image noise. Wiener Filter follows the statistical approach. To design this filter one should have the knowledge of spectral properties of the original image, the noise, and the linear time-variant filter whose output should be as close to the original as possible. The Wiener filter is especially effective in reducing additive noise, such as Gaussian noise.

Max and min filter Other names of minimum and maximum filters are erosion and dilation filters, respectively. These morphological filters work by considering a neighborhood around each pixel. From the neighbor pixels list, the minimum or maximum value is extracted and stored as the corresponding resulting value. These filters are mainly effective in preserving image features, including edges, while suppressing noise. Lastly, every pixel in the image is replaced by the resulting value generated for its associated neighborhood. If we implement max and min filters one by one they remove certain kinds of noise (salt-and-pepper noise).

2.2.3 Contrast enhancement

Inadequate contrast can make it challenging to extract meaningful features from an image. Contrast enhancement techniques, such as histogram equalization or contrast stretching, can improve the visibility of details in the image. By implementing contrast enhancement, filtering techniques, and removal of noise, methods we can enhance the image quality. Enhancement of image quality and pixel intensity are also carried out after preprocessing.

2.2.4 Illumination correction

Changes in lighting conditions can affect the appearance of objects in an image. , We can use shading correction or histogram-based preprocessing methods to reduce the impact of illumination variations which is very useful for extracting more consistent and informative features.

2.2.5 Edge detection

Apply edge detection algorithms like Sobel, Canny, or Laplacian of Gaussian to highlight important features in the image. The paper [276] combination of fuzzy image edge-detection with convolutional neural network for a computer vision system based on their body model it classify guitar types. The article is mainly focused on comparing the effects of performing image-preprocessing techniques on raw data (non-normalized images) with the help of various fuzzy edge-detection methods, specifically fuzzy Prewitt, fuzzy Sobel, and fuzzy morphological gradient, before inputting the images into a convolutional neural network for the classification task.

2.2.6 Normalization for convolutional neural networks (CNNs)

In the CNN model feature extraction process from each image, we subtract the mean pixel value of the dataset and divide it by the standard deviation. This is called z-score normalization and helps to improve the model convergence.

In essence, image preprocessing acts as a crucial data preparation step, enhancing the quality of the data and making it more suitable for subsequent feature extraction processes. It ensures that the extracted features are more reliable, meaningful, and robust, ultimately leading to better performance in image analysis, computer vision, and machine learning applications. The specific preprocessing steps used will depend on the characteristics of the images and the requirements of the feature extraction task.

2.3 Extraction of color feature

Color features can be considered the most trivial as well as usable property of an image, it is the most visceral and common feature of the image to be extracted easily. Representation of color information can be described with different methods. Color is unattached from the orientation and image size, due to background complication and robustness. There are many perspectives to represent the color, in the last ten years much research on Content-Based Image Retrieval (CBIR) has been done. CBIR depends on analyzing and extracting various features from images, including color features. Based on these features CBIR systems retrieve similar images from a database based on a query image's content. Color feature extraction plays an important role in CBIR systems, as it helps in characterizing and comparing images based on their color information. Figure 3 shows the block diagram of the CBIR system.

Much research has been done on the retrieval of images in the last few years in a wide range. Kato proposed an auto retrieval of the images depending on color shape from the database, known as Content-Based Image Retrieval (CBIR). An image representation utilizes quantization to reduce the time for computation without any change in the image quality, so, less space and time are required for processing. To increase the performance of image feature extraction, and image quantization. Two factor that affect the color feature is surface

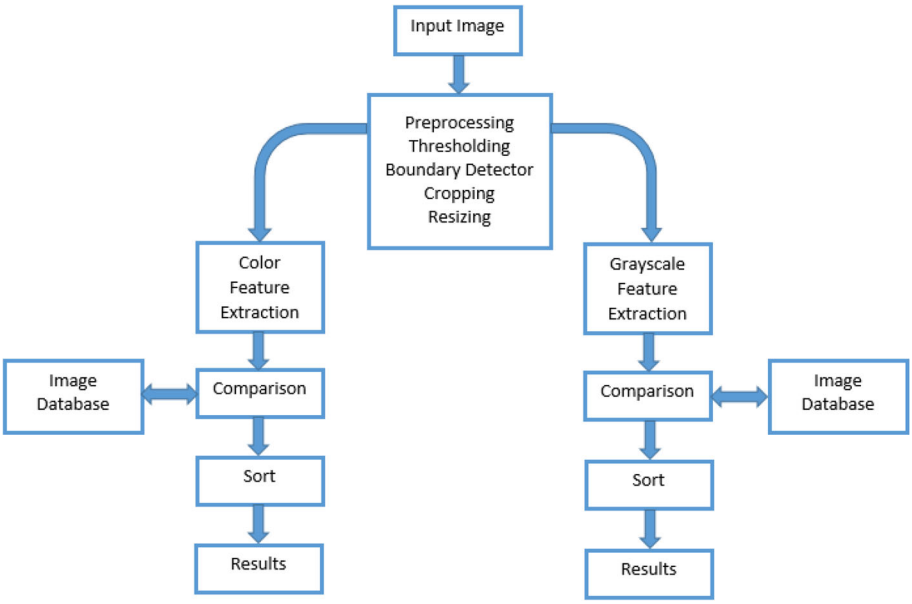

Fig. 3 Content-Based Image Retrieval (CBIR) Flowchart

reflectance property and spectral power distribution. HSV is a color model used for the color feature representation of an image. Hue, saturation, and value (HSV) are very sensitive to noise. The color feature is extracted by different techniques. For example Color Histogram and Color correlogram. Red, Green, and blue are the three main components of a color image for color space representation, and you can also see the accuracy of color spectrum [96, 97]. Table 1 describe the different color descriptor.

2.3.1 Histogram

There are two methods for color feature extraction methods, local methods and global methods. In Local methods, a portion of the image is considered, including color correlogram, local color histogram, color difference histogram, etc. On the other hand in global methods, the feature extraction process considers the complete image, including image bitmap, global color histogram, histogram intersection, etc.

Color histogram Color histograms represent the distribution of colors in an image. You can create histograms for each color channel (e.g., red, green, blue) or use a color space that separates hue, saturation, and value (HSV or HSL). For color feature extraction, mostly we use color Histograms [257, 258]. It shows the property of an image from a different view. This appears for the frequency distribution of color bins in an image. It reckon up identical pixels and store it. There are two types of color histograms. local color histogram and global color histogram. In local color histogram, we deal with a portion of an image. Local color histograms reflect on the spatial distribution of pixels which vanished in global color

Table 1 Contrast of Different Colour Descriptor

Author	Year	Method Used	Applications	Performance	Advantages
Liu, Guang-Hai, et al.	2011	Micro-Structure Descriptor (MSD) [222]	For color feature extraction by simulating human early visual processing	– The average retrieval precision and recall ratios by MSD Precision = 55.92%, Recall = 6.71 %	– MSD algorithm has high indexing performance and low dimensionality.
Blum, Manuel, et al.	2012	convolutional k-means descriptor [223]	This approach is able to learn meaningful features from RGB as well as depth data automatically.	– The improved accuracy margin is 5.9%.	– Accuracy of this descriptor is high for the used dataset.
Talib, Ahmed, et al.	2013	Correlogram [113]	Content-based Image Retrieval (CBIR)	– Performance of these descriptors shows high effectiveness and feasibility with database (Corel-10K and Cartoon-11K). – This method reduces the complexity $O(m^2d)$ to $O(m^2/2 + m/2)$.	– Two compact representations of the color correlogram, compact-generalized correlogram, and dominant color-based correlogram is presented.
Tian, Xiaolin, et al.	2014	EOD Hand Color-SIFT [224]	To achieve a weighted codeword distribution, which implements excellent retrieval performance than several previous methods	– CPU time required for features extraction of the natural image is 0.8s.	– This method gives the best result on large datasets
Lan, Rushi, Yicong Zhou, and Yuan Yan Tang	2015	Quaternionic Local Ranking Binary pattern (QLRBP) [145]	Toproposed QLRBP descriptor	– Performance of QLRBP is more than 2% compared to several state-of-the-art methods for different databases.	– QLRBP is image characteristics and shows more robustness for different aspect Compared with existing LBP-based methods.

Table 1 continued

Author	Year	Method Used	Applications	Performance	Advantages
Bora, Dibya Jyoti, and Anil Kumar Gupta	2016	Histogram [95, 97]	For feature extraction and image segmentation	– PSNR value for the proposed approach is 12.0306.	– Combination of Otsu's thresholding with Sobel Filter makes Computation easy.
Lu, Ze and Jiang, Xudong and Kot, Alex	2017	Ternary-Color LBP (TCLBP) [200]	Face recognition	– Face recognition accuracy for different database is FERET (86:7292%), FRGC (75:3234%), Georgia Tech (94.5714 %) and LFW (78:5667%)	– Compare to other state-of-the-art Color LBP, color LBP descriptors, CLBPT-CLBP gives visibly better face recognition performance on all 4 public face databases.
Ashraf, Rehan, et al.	2018	Automatic image retrieval [201]	Image retrieval	– Precision = 0.735 and Recall = 0.1485	– In this paper color features are used with the histogram and applied Haar wavelet transform to effectively reduce computational steps and help improve search speed.
Patruno, Cosimo, et al.	2019	Soft biometric features [202]	Avoiding the problem of re-identification	– Recognition accuracy for different database: BIWI RGBD-ID = 97.84%, KinectREID = 61.97 and RGBD-ID = 89.71%	– RGB-D data gives the best matching among a dataset of possible users.
Xie, Guangyi, et al.	2020	Dominant Color Descriptor and Hu Moments [203]	CBIR	– Precision (%) and Recall (%) for dataset: Corel-1K (78.75 and 9.45). – Precision (%) and Recall (%) for dataset: Corel-5K (60.05 and 7.26). – Precision (%) and Recall (%) for dataset: Corel-10K (53.17 and 6.38).	– In the proposed method, the Combination of DCD and HM has the advantages of shape and color detection.

histograms [259]. In the global color descriptor, we analyze every statistical color frequency in an image [260].

Histogram intersection Descriptor of the image was proposed in [261] Histogram Intersection (HI) which look at global color features using histograms. For a given pair of color histograms, X and Y with k bins for each, the HI is defined as shown in (1):

$$HI(X, Y) = \sum_{i=1}^{k} min(X_i, Y_i) \tag{1}$$

The performance of HI mostly depends on color space selection. HSV (Hue, Saturation, and Value) or CIELab color spaces [261] enhance the performance because they are orthogonal among the three color components.

Color histogram for K means (CHKM) For a color pixel 2^{24} different colors combination are possible. The color histogram is used for the K-mean (CHKM) feature representation. This technique has, a single color for a common color palette which is almost identical to a particular pixel color is considered to replace that pixel color. The output of this algorithm is the mean of all pixels in every cluster, This becomes the new initial value for the next step training. For k^{th} color bin the CHKM feature was defined as shown in (2):

$$g_k = \frac{N_k}{N} \tag{2}$$

Where N_k is the number of the pixels in the k^{th} cluster and N is the total pixel numbers.

Dominant color descriptor (DC) Dominant Color depends on compactly representing an image with the help of a small number of its well-known colors. This technique is incompetent for object-based image retrieval (CBIR). Dominant color extraction procedures are a collection of different tasks such as color quantization, color space selection, calculating a histogram of the quantized image, dominant color determination, and computing each dominant color percentage. Article [262] HSV color space was selected and a quantized image having 72 different colors and DC descriptor as shown in (3):

$$DC = \{C_i, P_i\}, i = 1.....N \tag{3}$$

C_i represents N most representative colors are selected as dominant colors and P_i denotes their percentages.

There are many algorithms for color feature extraction, Color histogram is one of them for features of different types of colored images. Two color spaces RGB and HSV are represented by color histogram [2]. It represents the ratio of each color corresponding to that image. CBIR is used for image feature extraction of image feature concerning color and shape. Histogram also uses this technique to extract the image feature[10]. CBIR is one of the most used techniques by many researchers to get the image properties [9]. In the image histogram, the grey level values vary from 0 to 255, because of large dimensional data this range of values cannot be utilized as a feature vector. Image pixels are subdivided into bins to represent image information accurately. We utilize the width of very small bins by dividing the pixel of the image. So, if you sample in a large bin, this will increase the frequency of each bin and will not be able to differentiate two or more objects in the image. Which in turn decreases the accuracy of the histogram. One of the most basic color content representations is the color histogram, In the histogram, the statistical color distributions are described by quantizing color space. The work in the paper [89], describes a different color-based method of image differentiation

that contains the same color histograms, and also shows a novel method of image retrieval by utilizing the technique of auto segmentation. A stochastic algorithm approach is used with the brightness of the regions for image segmentation. This approach finds the group of different equally bright element regions [90]. Image is segmented based on particular brightness in a class and based on that particular class feature extraction has taken place [91].

2.3.2 HSV

HSV color space is based on three factors Hue (H), 105 Saturation (S), and Value (V) [93]. This color model represents both chromatic and achromatic information of an image. Hue describes the detail of the colors, saturation shows the level of color dominance and the value represents the brightness level. HSV color space can be considered one of the differences between HSV and RGB color space. It differentiates the Intensity value from the color space. Hue contains a value range of 0 to 1.0, Value from 0 to 1.0 corresponds to the color starting from Red, going through many colors in between, and ending with again at Red. Green, cyne, blue, and magenta are in between colors. This model represents saturation variation from 0 to 1.0 with color variation from unsaturated to fully saturated. CBIR systems [12] commonly use HSV color space. A different HSV color representation in three dimensions is hexane. In this representation, intensity is shown by the central vertical axis [94]. The Hue is presented in the angular form with a range of $[0, 2\pi]$ varies from red, green, blue, and again red with corresponding angel 0, $2\pi/3$, $4\pi/3$ and 2π [95].

In many techniques, mean, skewness, and standard deviation are computed in (4), (5), and (6) respectively. So that the features can be extracted easily.

$$\mu_i = \frac{1}{M}\sum_{k=1}^{M} f_{kl} \tag{4}$$

$$\sigma_i = \left(\frac{1}{M}\sum_{l=1}^{M}(f_{kl} - \mu_k)^2\right)^{\frac{1}{2}} \tag{5}$$

$$\gamma_i = \left(\frac{1}{M}\sum_{l=1}^{M}(f_{kl} - \mu_k)^3\right)^{\frac{1}{3}} \tag{6}$$

where f_{kl} is the color value

2.3.3 RGB

RGB is based on three primary colors Red-Green-Blue, this color model can not mimic the human eye's perception and is not capable of differentiating the different luminance components from the chrominance ones. This color model can be utilized in many applications such as digital images, digital cameras, digital screens, etc. The work in the article [98] shows RGB color model importance in different industries and research areas in today's scenario. The following Fig. 4 shows the RGB color Model.

2.3.4 Application of color feature extraction in image forensic and security

Color feature extraction techniques play a very important role in image forensics and security by helping to analyze and verify the color properties of images and videos. Color feature

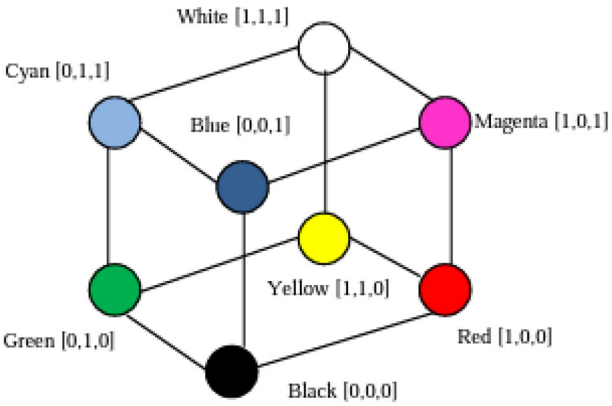

Fig. 4 RGB color Model

extraction techniques are essential for verifying the authenticity and integrity of visual content, detecting anomalies, and aiding in security-related applications across various domains. These techniques can be used for various applications in these fields. when combined with other image analysis methods, contribute to robust image forensic and security solutions. Following are a few major applications of color feature extraction in image forensics and security.

- During image forgery detection when images are tampered with, color inconsistencies may occur. The use of color feature extraction can identify such discrepancies, making it useful for image forgery detection.
- In biometric systems, such as iris recognition, where the color and texture of the iris are analyzed for identification and authentication purposes color features can be used.
- For digital image databases used for security and law enforcement, color features can be used to tag and retrieve images based on their dominant colors, facilitating efficient image search and retrieval.
- In fire safety and security systems, color-based feature extraction can be used to detect the presence of flames or smoke in real-time video feeds, enabling prompt responses.
- Implementation of these features can be employed to determine the source of an image or video, helping to verify the authenticity of visual evidence in legal cases.
- Color analysis is applied to video frames to detect any temporal inconsistencies or color alterations that may indicate video tampering or deepfake creation.
- Color-based watermarking techniques can be used to embed and extract information in images for copyright protection and authentication.

2.4 Noise feature

During the transmission, coding, processing, and acquisition noise is present in the digital image. Noise in the digitalize is primarily introduced during image acquisition and transmission [3]. It is difficult to remove noise without knowing the noise model. That is why, it becomes very essential to know the noise models of image denoising techniques. Following are some noise models that demonstrate different types of noise in an image.

2.4.1 Noise models

Noise means unwanted information in a digital image. It shows the change in a given image. The noise changes the authenticity of an image. It creates some unwanted brightness, lines, corners, blurred objects, and changes in the background scene. To resolve all these unwanted effects knowledge of noise models is necessary. Here are some important useful noise models are given below.

Gaussian noise model Gaussian noise is produced by some natural incident, another name for Gaussian noise is normal noise. The Gaussian noise model shows the Gray values distribution of digital images. Statistical terms are used to show the Gaussian noise model. For gray value probability density function or normalized histogram is used. The probability density function is represented as follows.

$$P(g) = \sqrt{\frac{1}{2\pi\sigma^2}} e^{-\frac{(g-\mu)^2}{2\sigma^2}} \tag{7}$$

Where g = Gray value, $\sigma = Standard\ deviation$ and $\mu = Mean$

Speckle noise The behavior of Speckle noise is very similar to the Gaussian noise. Application of Speckle noise is laser, radar, and in many other coherent imaging areas. It has multiplicative properties. Speckle noise is not useful for ultrasound imaging. Following (8) is the probability density function that follows a gamma distribution.

$$F(g) = \frac{g^{\alpha-1} e^{(\frac{-g}{a})}}{\alpha - 1! a^{\alpha}} \tag{8}$$

White noise Due to white light correspondence, its name is white noise, although light white does not have flat power spectral density property over the visible band. The strength of this noise is very similar to the power spectral density function. Correlation is not possible in white noise because pixel values are different from their neighbors. This results in the zero value of correlation.

Brownian noise It is also known as red noise. Brownian motion takes place due to the arbitrary placement of particles that are suspended in the fluid. This noise can also be generated by white noise.

Photon noise (Poisson noise) This is a physics phenomenon, in which natural vibration of the incident photon flux creates noise. Electromagnetic waves like gamma rays, x-rays, and visible lights show statistical nature properties. Due to these properties photon noise is produced. The application of these rays is in the field of medicine and many other fields also. Equation (9) shows the Poisson noise formula.

$$p(f_{p_i} = k) = \frac{\lambda^k i e^{-\lambda}}{k!} \tag{9}$$

Rayleigh noise Following (10) represents probability density function in Rayleigh noise:

In the images of radar range, Rayleigh noise can be seen.

$$P(f) = \begin{cases} \frac{2}{b}(f-a)e^{\frac{-(f-a)^2}{b}} & \text{for g} \geq \text{a} \\ 0 & \text{for f} < \text{a} \end{cases} \tag{10}$$

Where $\mu = a + \sqrt{\frac{\pi b}{4}}$ and variance $\sigma^2 = \frac{b(4-\pi)}{4}$

2.4.2 Noise feature extraction techniques

In image processing, Noise feature extraction involves quantifying and identifying the noise present in an image. The applications of image feature extraction techniques are in various fields, including quality assessment, image denoising, and computer vision tasks. Here following are some steps involved in noise feature extraction.

Steps in the image feature extraction

- First we take the input image
- The Second step is the preprocessing of the image using normalization and segmentation.
- Further different noise estimation has been done. In local noise estimation, we calculate the noise variance or standard deviation in small local regions (e.g., blocks or patches) of the image. In Global Noise Estimation, we Calculate an overall estimate of noise for the entire image.
- In this step noise model selection is done to determine the type of noise (Salt-and-pepper noise, Gaussian noise, speckle noise, and Poisson noise) present in the image.
- In this step we extract the noise feature using different methods like Frequency-domain Features, Statistical Features, Wavelet Transform, Filter-Based Features, and Local Variance Features.
- After feature extraction we select the appropriate feature according to your applications.

Thresholding techniques based on Wavelet are very effective in denoising. Nonsignificant wavelet coefficients a preset threshold values that are discarded as noise and the image is built from the rest of the significant coefficients. Compared with the linear denoising methods that blur images as well as smooth noise, the nonlinear wavelet thresholding schemes preserve image singularities better.

Generally, thresholding is classified into two parts soft thresholding and hard thresholding. In the hard thresholding technique coefficients smaller than threshold value T are set to zero while other coefficients are remained unchanged. In Soft thresholding, a coefficient value less than the threshold value T is set to zero while important coefficients reduced by an absolute threshold value.

$$G(x, y) = I(x, y) + N(x, y) \tag{11}$$

Where I(x,y) represents the original signal, G(x,y) is the noisy image, and N(x,y) indicates the additive noise. Let W and $W^{(-1)}$ be the forward and inverse wavelet transform operators respectively. We intend to wavelet shrinkage denoise G(x,y) in order to recover I(x,y). D(Y, λ) is the shrinkage filter used where λ is the threshold for data Y (wavelet coefficient). Figure 5 is the example of an image with noise and denoise using soft thresholding method

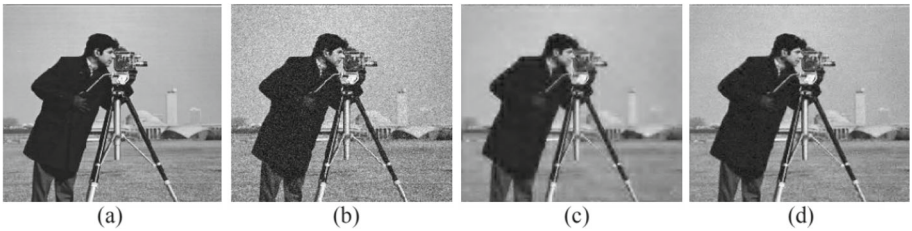

Fig. 5 Cameraman Image: (a) Original (b) Noisy Image with Noise Level 20 (c) Denoising using VisuShrink (d) Denoising Soft Thresholding Method

The three steps involved in the procedure are as follows:

$$Y = W(G) \tag{12}$$

$$Z = D(Y, \lambda) \tag{13}$$

$$I = W^{(-1)}(Z) \tag{14}$$

$$D(Y,) = sgn(Y) * max(0, |Y| - \lambda) \tag{15}$$

$$\mathrm{D(Y, \lambda)} = \begin{cases} \mathrm{Y, if|Y| > \lambda} \\ \mathrm{0,\ otherwise} \end{cases} \ldots\ldots \tag{16}$$

Equations 15 and 16 are Soft, Hard thresholding respectively and Y is the wavelet coefficient). The operator D nullifies all values of Y for which $|Y| < \lambda$ and shrinks toward the origin of all values of Y for which $|Y| > \lambda$. It is the latter aspect that has led to D being called the shrinkage operator in addition to being a Thresholding operator.

Algorithm

- Take the input image and convert it to grayscale in order to obtain a single-color channel matrix from the three-color channel matrix.
- The image is then filtered, i.e., denoised by using DWT(Discrete Wavelet Transform). Denoising an image using DWT and soft and hard thresholding.
- Get the noise residue features by subtracting the denoised image from the original one.
- The noise residue features of images are divided into blocks of 8 Ã- 8 pixels and correlation is calculated amongst adjacent blocks of the image and stored in a correlation matrix.
- Finally, the Gaussian Mixture Density(GMD)based Bayesian classifier, classifies the values of the correlation matrix and the threshold of the Bayesian classifier is set by the Expectation Maximization algorithm.
- The forged or spliced blocks are separately marked out in the output result.

Figure 6 shows the description of image forgery detection using the noise feature. There are many articles based on forgery detection which is based on noise feature extraction techniques. In this article [263], a novel method for nonintrusive scanner forensics that uses intrinsic sensor noise features is proposed to confirm the source and integrity of the digitally scanned images. Scanning noise is analyzed from several aspects using only scanned image

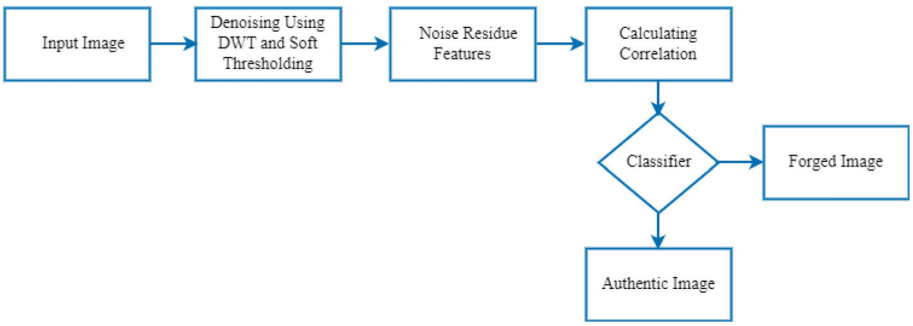

Fig. 6 Image Forgery Detection using Noise Feature [281]

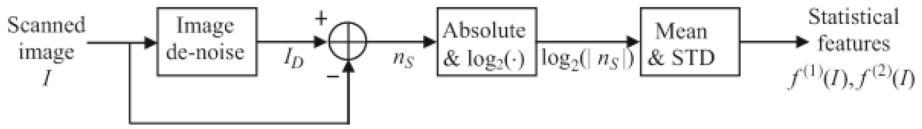

Fig. 7 Denoising algorithm for Statistical noise feature extraction

samples, including through different methods of image denoising, wavelet analysis, and neighborhood prediction, then from each characterization statistical features.

Statistical feature extraction With the help of the following three aspects we statistically characterize the scanning noise. First, for scanned images to estimate their noise we apply different denoising algorithms and extract moment-based features of the noise. In the second step, we pivot on histograms of wavelet coefficients at the finest scale, where scanning noise affects the shape of the histogram. Finally, we see the impact of noise effect on the local smooth part of a scanned image via linear prediction modeling.

Statistical noise features from denoising algorithms The following figure 7 is used to extract the statistical features of scanning noise. In the figure, we apply the denoising operation on image I to obtain denoised version I_D. By pixel-wise subtraction $n_s(i,j) = I(i,j) - I_D(i,j)$ we found scanning noise n_s at the pixel location (i, j). Statistical properties of the estimated noise $n_s(i,j)$ can then be used as features. In this test, we use the absolute value of scanning noise and take the log transform shown in (17).

$$
\begin{aligned}
f^{(1)}(I) &= \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\log_2\left(|n_S(i,j)|\right) \\
f^{(2)}(I) &= \left(\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(\log_2\left(|n_S(i,j)|\right) - f^{(1)}(I)\right)^2\right)^{\frac{1}{2}}
\end{aligned}
\tag{17}
$$

Statistical noise features from wavelet analysis In this part, with the help of wavelet analysis, we characterize scanning noise. In 2-D wavelet decomposition after one stage, the partition of an input image has been done in four different subbands, namely, lowa â€“low (LL), lowâ€“high (LH), highâ€“low (HL), and highâ€“high (HH) subbands. Among these four subbands, the LL subband carries low-frequency components, while the other three contain high-frequency components.

we extract statistical noise features in the wavelet domain following the steps shown in Fig. 8. Given a scanned image I, we first normalize it to be $\tilde{I}$ with unit power

$$
\tilde{I}(i,j) = \frac{I(i,j)}{\left(\frac{1}{MN}\sum_{k=1}^{M}\sum_{l=1}^{N} I(k,l)^2\right)^{\frac{1}{2}}}
\tag{18}
$$

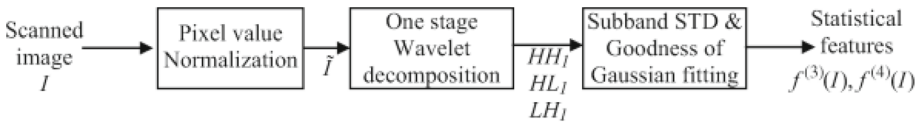

Fig. 8 Statistical noise feature extraction via wavelet analysis

Statistical noise features from neighborhood prediction In this section, we focus on smooth regions of scanned images and apply linear prediction modeling to study how scanning noise affects the relationship between a pixel in a smooth region and its eight neighbors. In the following, we characterize scanning noise by applying linear prediction modeling to smooth regions of scanned images, and the major steps are shown in Fig. 9.

Given a scanned image I, we identify its smooth regions based on local image gradient

In the paper [249] we measure the performance of three popular image feature extraction methods as Speeded-Up Robust Features (SURF), Scale Invariant Feature Transformation (SIFT), and Histogram of Oriented Gradient (HOG) by experimenting with the images tampered by three types of noise such as salt & pepper, gaussian, and speckle. In this process, we measure the efficiency of different feature detection methods such as SURF, SIFT, and HOG with input noisy images. To identify key points, the first stage of computation uses the Difference of Gaussians (DoG) function. The scale space function is represented by (19).

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{19}$$

Where $G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}}$

In the second stage by eliminating low contrast keypoints we identified keypoints in the image.

In SURF within a rectangular region the integral image I(x) corresponds to the sum of all pixels in the input image I.

$$I(X) = \sum_{i=0}^{x} \sum_{j=0}^{y} I(i, j)$$

In the end, With the help of creating a square region associated with the selected orientation SURF descriptor is extracted.

In the fie;d of computer vision and image processing HOG descriptor is used for object detection. HOG descriptor is implemented on a part of the image. The histograms are stacked into a single vector and this resulting vector is known as a descriptor. We calculate the gradient by filtering the standard grayscale image I with the help of subsequent filter kernels [250].

$$D_x = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$$

The x and y derivatives of image I are derived by performing convolution with filter kernels as follows:

$$I_x = I * D_x and I_y = I * D_y$$

In this presented method we have seen that the HOG descriptor focuses more on the textural information of the image. Between SIFT and SURF, SIFT detects more features points and matching points nevertheless the effectiveness of SURF is better than SIFT. However. In features detected of the original image. Input is tested one by one against the noisy image. When the same information is found then a new match is originated.

This article [251] presents a Pareto-based evolutionary multi-objective approach, which optimizes the functionals in the Trace Transform for extracting image features that are robust

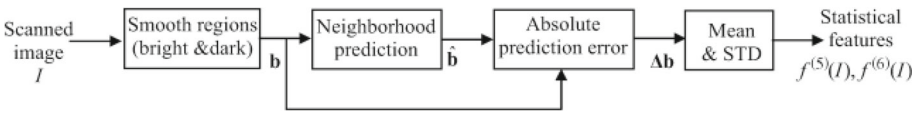

Fig. 9 Statistical noise feature extraction via neighborhood prediction

towards invariant to geometric deformations such as rotation, scale, noise, and translation (RST).

Because of their less computation time Global support region feature extraction methods are more efficient and reliable among all support region feature extraction approaches. Trace transform has main two objectives, namely, minimization of the within-class variance. , maximization of the between-class variance. The optimization of the Trace transform can now be represented as a bi-objective optimization problem by the following (20).

$$\begin{aligned} min\{f1, f2\} \\ f1 = S_w, \\ f2 = \frac{1}{(S_b + \epsilon)}, \end{aligned} \tag{20}$$

In this method experiment, images from the COIL-20 database are transformed using a scaling factor (from 1.0 to 0.3.) with respect to Robustness to Scale (COIL-20 Database).

Both pepper noise & Gaussian noise are added to the whole image, with the help of translation, random rotation, and a predefined scaling factor is performed for RST transformations.

With the help of three strategies (sampling parameters and pixel locations, symmetry of the Trace functionals, and Triple features) The computational efficiency of the evolutionary optimization of the Trace transform was enhanced.

In this article [252] a new technique to correlate statistical image noise features with three EXchangeable Image File format (EXIF) header features for manipulation detection has been proposed.

Image tampering such as contrast adjustment and brightness can affect these noise features and expedite enlarged numerical differences between its estimated EXIF feature and the actual from the noise features. By using the numerical difference as a manipulation indicator, we achieve excellent performance in detecting common brightness and contrast adjustment.

Following are the, three EXIF header features [253] related with DCSI:

$$\begin{aligned} Aparture : y_1 = \log_2(F^2) \\ Shutterspeed : y_2 = \log_2(\frac{1}{t}) \\ ISOspeedrating : y_3 = \log_2(\frac{I}{3.215}) \end{aligned} \tag{21}$$

After combining the three detection results We achieve an average accuracy of 99.8%. For the other two different combinations of the image sources, the average accuracies of combining the three detection results are 99.2% and 98.1%. We waged a least squares mixture to solve the regression weights from the intact images By formulating each EXIF feature as a weighted average of its selected noise features. By combining performance results attain 99.7% average accuracy for contrast adjustment detection and 99.8% average accuracy for brightness adjustment detection. Comparison with state of art show that the proposed algorithm not only performs well in detecting contrast manipulations but also works well to detect pure brightness adjustment. It is hard to meet the three EXIF-image constraints related to ISO, aperture, and shutter speed, simultaneously.

2.4.3 Application of noise feature extraction in image forensic and security

Noise feature extraction techniques in image forensics and security are essential for analyzing and identifying different types of noise or disturbances in images and videos. When combined

with noise feature extraction techniques with other image analysis methods, it is very helpful for image forensic experts and security professionals to assess the authenticity, integrity, and quality of visual content in various applications related to security, law enforcement, and digital media authentication. In the field of tamper detection, image authentication, and overall image quality assessment this technique is very useful. The applications of noise feature extraction in these fields are as follows:

- Noise analysis can help identify inconsistencies in noise patterns introduced during image tampering or forgery, such as copy-paste operations or image splicing. Inconsistent noise levels or patterns can indicate areas of manipulation.
- Noise features are useful to detect the splicing or pasting of one image onto another. Differences in noise characteristics between the spliced regions and the rest of the image can be indicative of tampering.
- It is used to assess the quality of images and videos, identifying noisy or low-quality segments that may affect the reliability of visual evidence.
- Noise feature extraction technique is used to assess the quality of images and videos, identifying noisy or low-quality segments that may affect the reliability of visual evidence.
- In legal cases, noise feature extraction helps verify the authenticity and integrity of digital evidence, ensuring its admissibility in court.
- Deepfake videos often contain subtle inconsistencies in noise patterns. Noise analysis can assist in detecting such discrepancies, aiding in the identification of deepfake content.

2.5 Texture feature

The texture feature of an image is one of the relevant features for image processing. This feature is very important for image processing. Many researchers have used texture properties for the purpose of image discrimination. The texture method is used by the human visual system for object identification. The following Fig. 10 shows the types of texture.

To get the pixel properties, the color feature is used, but for the group of pixels, the texture feature is used. The application of texture features is in the search and feature extraction of an image. Like color features, texture also comes in the low-level feature. Analysis of texture properties of Texture methodology has been investigated by a number of researchers. The main purpose was an extraction of lower-order feature vectors and a reduction of computation load.

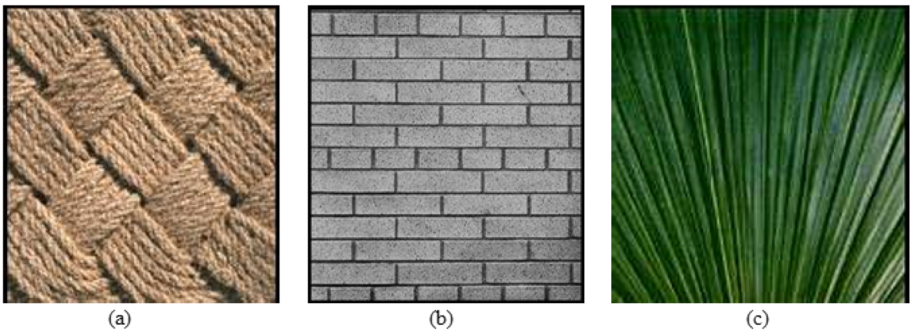

Fig. 10 Different types of Texture Examples

In the last decade, texture classification, analysis, and segmentation have an interesting topics in the image processing area. But still, there is scope for further research in this field. In grey-level image analysis, Local binary pattern (LBP) texture is the most frequently utilized pattern. Figure 11 classifies the texture feature extraction techniques.

Local binary pattern (LBP) was proposed by author [100]. LBP is robust in nature, for image feature extraction it uses pixel-level intensity. The work in the article [101], demonstrates a standard method, called a Local tri-directional pattern. It's abbreviated as LTriDP. In the LTriDP method pixel comparison of neighbour and center pixel is required.

2.5.1 Texture feature extraction method

In statistical texture analysis, texture features are calculated from the statistical distribution of observed fusion of intensities at particular places in proportion to each other in the image. Level Co-occurrence Matrix (GLCM) method is a technique to extract second-order statistical texture features. In this paper [264] four important features, Angular Second Moment (energy), (inertia moment), Correlation, Entropy, and using Xilinx ISE 13.4 the Inverse Difference Moment are selected for implementation (Table 2).

In the article [277], a new method of image forgery detection based on DRLBP and SVM has been proposed. Firstly partitioning of chrominance components of an input image is done into overlapping blocks, after that, the DRLBP code of each block is calculated. Later, For each block histograms of both Cb and Cr components are used as features. We use SVM as a classifier. The result of this method was considerably calculated on individual and combined datasets in terms of training and testing on splits of the same dataset. The method proposed

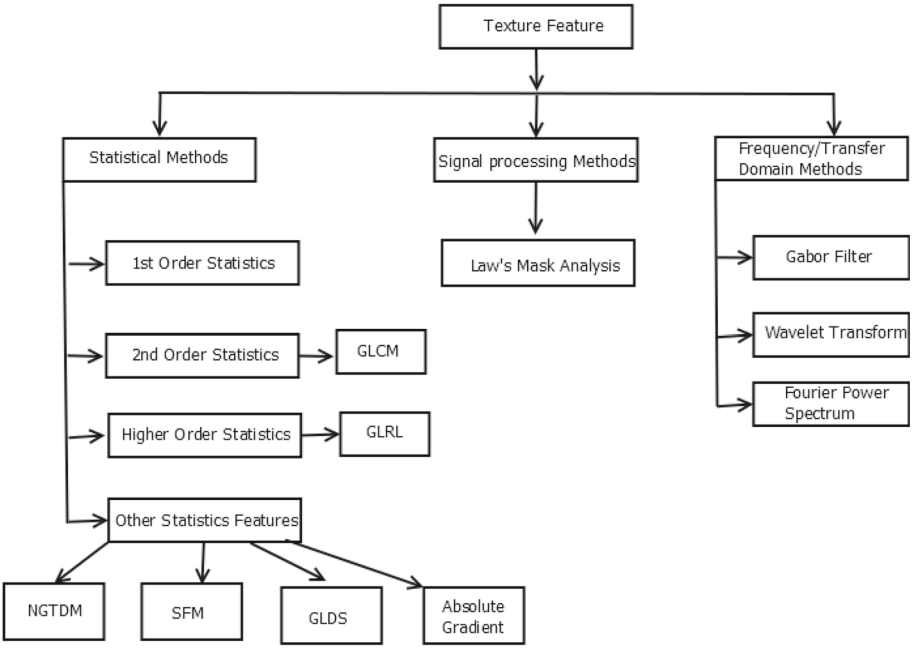

Fig. 11 Different Texture Feature Extraction Techniques

Table 2 Different Types of Texture Methods

S.No.	Method	Texture Method	Input	Performance	Limitations
1	Wavelet [25, 114–116, 146, 147]	Temporal	KSC, Botswana, Pavia-U datasets [146]	– Overall accuracy of all four data set for mixed lasso is 93.15%. – Testing time for lasso is 1.3+25.2 seconds.	– Wavelet is sensitive to rotation, so the robustness is not good.
2	Curvelet [121, 122, 160, 161]	Temporal	Brodatz and the KTH-TIPS databases [122]	– Curvelet-based texture features improve the classification accuracy of different small objects with Multi-resolution and multi-orientation. – Curvelet implementation complexity is $O(n^2 log(n))$ for $n \times n$ Cartesian.	– This experiment can perform for small data set only.
3	Gabor [117, 118, 149, 150]	Temporal	CH and NBI images[117], IKONOS images [118]	– Pair of Gabor texture features and SVM classification give the best accuracy (89.8%). – Gabor textures are important for spatial feature extraction of the image with good accuracy and computational time.	– Classification accuracy can be further improved with optimized parameters.

Table 2 continued

S.No.	Method	Texture Method	Input	Performance	Limitations
4	FT/DCT [16, 23, 151, 162]	Temporal	CASIA database [162]	– A novel work is described for human iris recognition based on using 1D Discrete Cosine Transform (DCT) with two data sets (CASIA and Bath). – Recognition rate is one hundred percent for both the database. – Complexity of this method is low.	– Because of the diversity of iris image sources and the lack of public data set, Not possible to extrapolate the output too large.
5	Fractal dimension(FD) [135, 164, 165]	Spatial	Brodatz album[165]	– Comparatively FD-based approach performs better than the traditional algorithm for the capability of nonlinearly enhancing complex texture details in smooth areas.	– This technique requires high computation cost, which is not good for scaling.
6	GLCM based method [131, 132, 156–159]	Spatial	Brodatz album [132]	– This method uses 16 different features extracted. Which is based on GLCM to capture the texture information. – This method has been tested with a big range of different textures with a small percentage error of 3.57% and a high percentage of accuracy.	– With some classifiers GLCM can only perform for a limited type of image.

Table 2 continued

S.No.	Method	Texture Method	Input	Performance	Limitations
7	Tamura SAR [134, 155, 163]	Spatial	Brodatz [163]	– This feature is effective in image texture directionality strength and mirroring the degree of the image texture direction.	– Processing cost of this technique is high. – Sometimes it is hard to find the pattern in the object.
8	Texton [133, 152, 153]	Spatial	PolSAR images [152]	– Combination of Textons with sparse coding, SVM, and wavelet shows effectiveness and adaptability for PolSAR image Classification. – Accuracy of all five dataset are 87.88%, 91.02%, 97.79%, 92.55% and 85.88%.	– This method is sensitive toward rotation, noise, and scale.

in this article was evaluated using various benchmark datasets: CoMFoD, DVMM, CASIA v1.0, CASIA v2.0, MICC-F2000, MICC-F220, UNISA, FRITH, Set-A, and Set-B.

In the preprocessing stage, a forged image is shown in Fig. 12 with respective components of RGB, HSV, and YCbCr color spaces. We have noticed that each part explores the image content in detail excluding chroma components (CbCr), which emphasize the weak signal content (little image detail) of the image. , this paper uses the YCbCr color space for different operations.

In feature extraction stage new texture descriptor called DRLBP (Directional Robust Local Binary Pattern) is introduced. DRLBP assigns a weight factor carrying gradient and texture information, enabling it to better represent microstructure patterns by combining edge and texture details. This makes DRLBP suitable for detecting image tampering.

To identify an image as authentic or forged is a two-class problem. The process of training a classification SVM mode. which is shown in Fig. 13. To conduct these experiments, a cross-validation (CV) protocol is employed, where each dataset or combination of datasets is divided into k (typically 10) folds. The SVM (Support Vector Machine) parameters are fine-tuned on nine of these folds, and the chosen parameterization is then applied to the remaining fold. This process is repeated for each fold, resulting in k iterations.

These extracted features are then fed into the trained image classification model. The primary objective is to determine whether the model can accurately classify these test images as either authentic or forged. Several publicly available benchmark datasets, including Columbia

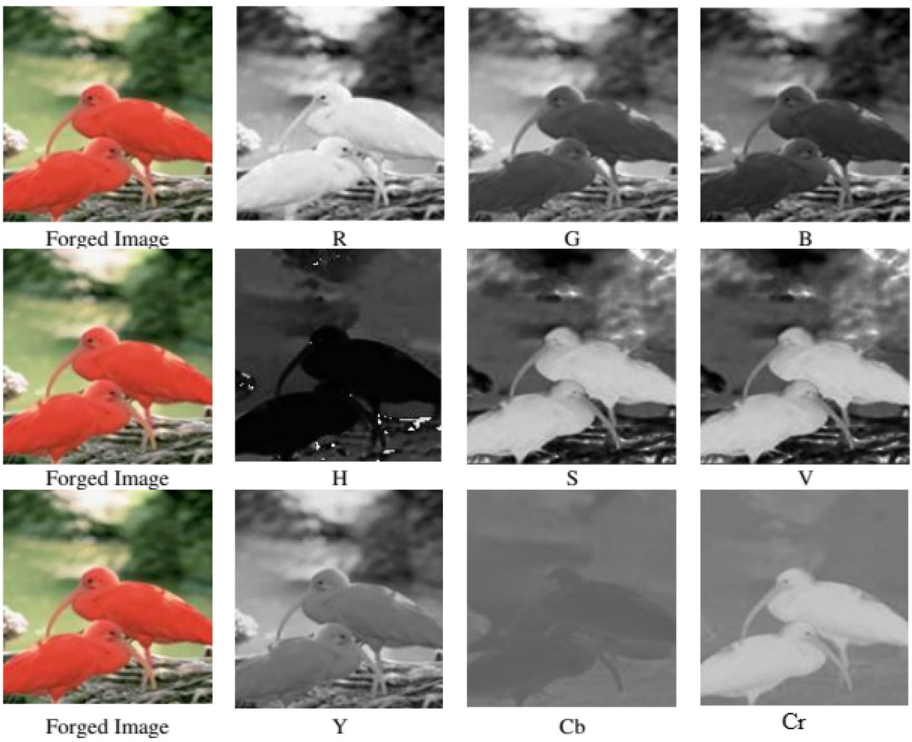

Fig. 12 Visualization of R, G, B, Y, Cb, Cr, H, S, V channels of a Forged Image, using RGB (Row1), YCbCr (Row2) and HSV (Row3) Color Spaces from Left to Right

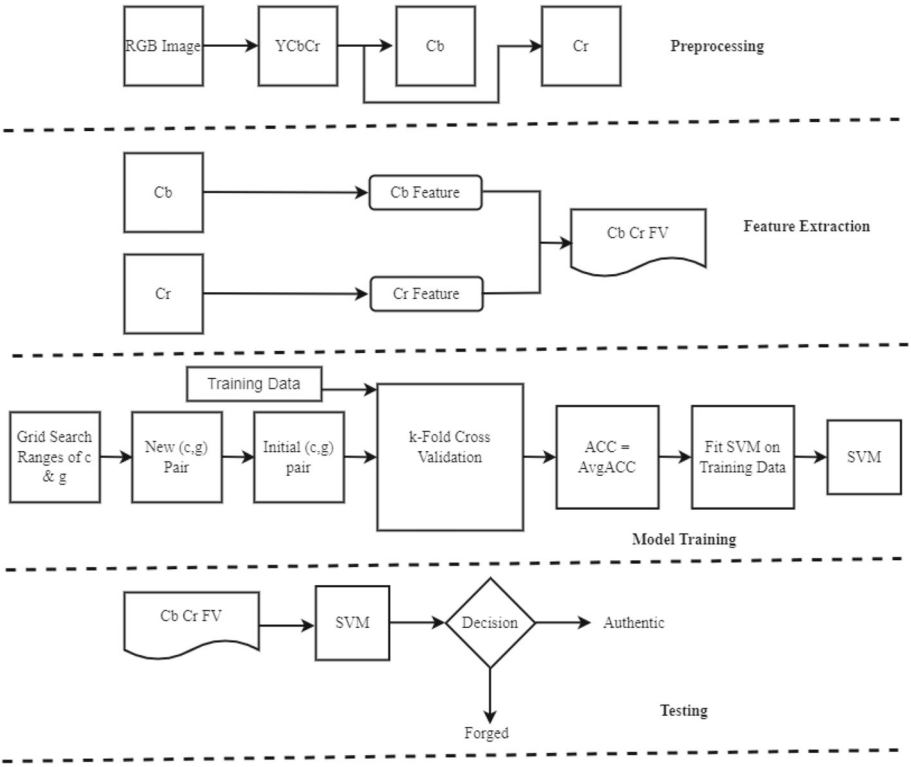

Fig. 13 Image Forgery Detection using Edge-Texture Feature

color DVMM, CASIA v1.0, CASIA v2.0, CoMFoD, UNISA, MICC-F220, and MICC-F2000, are employed to evaluate the proposed approach. These datasets are known in the field of image forgery detection and serve as standard reference points for testing and validation

2.5.2 Protocol to select texture extraction algorithms

Following are various algorithms for texture feature extraction. To select these algorithms some characteristics should be followed, these attributes are as follows.

- **Illumination (gray-scale) invariance:** The sensitivity of the algorithm to change in grayscale. This aspect is important when the lighting condition in industrial machine vision may be unstable.
- **Rotation invariance** Does the texture of images change if we change the rotation of the images?
- **Robustness in the presence of noise** The robustness ability of the algorithm is in a noisy environment which has an effect on the input image.
- **Computational Complexity:** Many algorithms are computationally intensive, for example in retrieval applications for large databases.
- **Generatively:** Can the algorithm facilitate texture synthesis, i.e. by regenerating the texture that was captured using the algorithm?
- **Popularity:** Which of them are more popular and more practical?

- **Easy to implement:** The algorithm should be simple to implement.

Angular second moment Another name for Angular Second Moment is Uniformity or Energy. It is the sum of squares of entries in the GLCM Angular Second Moment that measures the image homogeneity. The value of Angular Second Moment is high when image homogeneity is good or when pixels are very identical.

$$ASM = \sum_{i=0}^{Ng-1} \sum_{j=0}^{Ng-1} P_{ij}^2 \tag{22}$$

Where N_g is the gray tone and i, j are the spatial coordinates of the function p (i, j).

Inverse difference moment Inverse Difference Moment (IDM) is the local homogeneity. It is high when inverse GLCM is high and the local gray level is uniform.

$$IDM = \frac{\sum_{i=0}^{Ng-1} \sum_{j=0}^{Ng-1} P_{ij}}{1 + (i-j)^2} \tag{23}$$

The inverse value of Contrast weight represents the IDM weight value.

Co-occurrence matrix for gray level To classify a digital image using texture feature, a method has been proposed in the article [99]. This article represents easily computable texture features based on gray tone spacial dependencies. Further, a modified algorithm based on GLCM fusion and direction measure is given by author [102]. To fetch the image feature in GLCM second order statistic calculation is required. GLCM is used in many algorithms. In GLCM, the frequency for a pixel is calculated with the grayscale intensity in which the location of the m value is horizontally next to the pixel element (m, n). Where m represents the frequency of pixels that occurred horizontally adjacent to a pixel with a value of n. Extracted features like Energy, Homogeneity, Contrast, and Correlation as shown in Table 2.

2D-DFT and Hamming distance method In this method, the first method is image segmentation, which was done using a texture database. then image segmentation, the Transformation of every segmented image is done by utilizing a 2D-DFT transformation [4]. The transformation equation for 2D-DFT is shown in (24):

$$(Ff)(a,b) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) exp\left(-j2\pi \left(\frac{ax}{M} + \frac{by}{N}\right)\right) \tag{24}$$

Feature vector estimation Mean and standard deviation are used to decrease the input vector size by utilizing ANN. Equation (25) and (26) represent the mean and standard deviation respectively:

$$\mu = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{25}$$

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)^2} \tag{26}$$

Where x = Value of the pixel in the sub-image, n = Number of pixels. Xi = mean=0, s.d.=0, mean=1, s.d.=1,â€¦.mean=127, s.d.=127 is derived.

Wavelet transform Wavelet transform is used to get the detailed resolution of a small region of interest. In this transformation, the signal is subdivided into a basic function from a base

function. Translations and Scaling of a fundamental function in wavelets transform from the mother wavelet. $s \in \mathscr{R}^+ \quad u \in \mathbb{R}$

$$\psi_{s,u}(x) = \frac{1}{\sqrt{s}}\psi\left(\frac{x-u}{s}\right) \tag{27}$$

where s = Scaling and u = Translation

Wavelet original function is given in (28) as follows:

$$Wf(s,u) = \int_{\Re} f(x)\frac{1}{\sqrt{s}}\psi^*\left(\frac{x-u}{s}\right)dx \tag{28}$$

Energy filters and edginess Energy filters are an algorithm, which is based on convolution. To increase the performance of the textual property of an image (I) we apply Energy filters on the image with the help of different types of filter $g_1, g_2, ...g_N$, therefore, obtaining N new images $J_n = I * g_n (n = 1, ..., N)$. Then, the energy in the neighborhood of each pixel is calculated.

The following formula with Equation 29 calculates the gradient of an image I as a function of the distance "d" between neighbor pixels:

$$\begin{aligned} g(i,j,d) = \sum_{(i,j)\in N} & \{|I(i,j) - I(i+d,j)| + |I(i,j) - I(i-d,j)| \\ & + |I(i,j) - I(i,j+d)| + |I(i,j) - I(i,j-d)|\} \end{aligned} \tag{29}$$

where g(i, j, d) = edginess per unit area surrounding a general pixel (i, j)

Primitive length texture features Primitive length texture feature extraction is a technique in image processing used to describe the texture of an image by analyzing the lengths of primitive structures or elements present in the image. These primitive structures can be lines, curves, or other simple geometric shapes that collectively contribute to the overall texture pattern. Coarse texture is a collection of a large number of neighboring pixels with the same gray level value, whereas a less number of pixels is called fine texture. A continuous set of the maximum number of pixels in the same direction (with the same gray level) is called primitive. Each primitive is defined by its gray level, length, and direction. For example, if we represent B(a, r) as the number of primitives of all directions with length r and gray level with symbol a, N_r the maximum primitive length in the images, and K the total number of runs let M, N be image dimensions, L the number of gray levels given by $\sum_{a=1}^{L}\sum_{r=1}^{N_r} B(a,r)$, then we can define the following 5 features defining image texture: $a = 1 r = 1$

Short primitive emphasis: $\frac{1}{K}\sum_{a=1}^{L}\sum_{r=1}^{N_r}\frac{B(a,r)}{r^2}$; Long primitive emphasis: $\frac{1}{K}\sum_{a=1}^{L}\sum_{r=1}^{N_r} B(a,r)r^2$

Gray level uniformity: $\frac{1}{K}\sum_{a=1}^{L}\left[\sum_{r=1}^{N_r} B(a,r)r^2\right]^2$; Primitive length uniformity: $\frac{1}{K}\sum_{a=1}^{L}\left[\sum_{r=1}^{N_r} B(a,r)\right]^2$ Primitive percentage: $\frac{K}{\sum_{a=1}^{L}\sum_{r=1}^{N_r} rB(a,r)} = \frac{K}{MN}$

Autocorrelation-based texture features Autocorrelation is a mathematical concept that measures the similarity between a signal and a time-delayed version of itself. The textural character of an image depends on the spatial size of texture primitives. Small primitives give fine texture (e.g. silk surface) and large primitives give rise to coarse texture (e.g. rock surface). An autocorrelation function evaluates the linear spatial relationships between primitives along with the measurement of coarseness [229]. The function decreases slowly if the

primitives are large and the distance is increasing whereas it decreases rapidly if the texture consists of small primitives. However, if the primitives are periodic, then the autocorrelation increases and decreases periodically with distance. The set of autocorrelation coefficients shown below are used as texture features: In image processing, this idea is extended to measure the similarity between an image and a shifted version of itself. The autocorrelation-based texture features utilize the spatial relationships between pixel intensities to capture information about the structures present in an image or repeating patterns.

$$C_{ff}(p,q) = \frac{MN}{(M-p)(N-q)} \frac{\sum_{i=1}^{M-p}\sum_{j=1}^{N-q} f(i,j)f(i+p,j+q)}{\sum_{i=1}^{M}\sum_{j=1}^{N} f^2(i,j)} \tag{30}$$

where in i, j direction p, q is the positional difference, and M, N are image dimensions. Variation of (p, q) from $(0, 0)$ to $(9, 9)$ gives 100 features output.

2.5.3 Application of texture feature extraction in image forensic and security

Application of texture feature extraction techniques is widely applied in image forensics and security for various purposes like analyzing and identifying textural patterns and anomalies within images and videos. For image authentication, tamper detection, and object recognition these feature extraction techniques play a crucial role in various applications. We can see various applications of texture feature extraction techniques in different research articles. Some of texture feature extraction are as follows:

- In content moderation and filtering on the internet, texture features can be used to identify and filter out inappropriate or explicit content based on its visual texture characteristics.
- In medical imaging, texture features are used to identify anomalies and regions of interest in images such as X-rays, MRIs, and CT scans.
- In the fashion and textile industry, texture analysis is used to identify different fabrics and materials, helping to verify the authenticity of products.
- Texture features are used to detect tampering or alterations in images by analyzing inconsistencies in patterns and textures within the image.
- In matching and comparing images, especially when images are partially occluded or transformed texture feature is utilized. This is useful for image retrieval and recognition in security and forensic databases.
- Texture analysis can help detect discrepancies in texture patterns introduced during image forgery or tampering. Inconsistent textures may indicate areas of manipulation, such as cloning or texture synthesis.

2.6 Shape feature

A human being generally identifies objects with the help of some basic properties like color, size, and shape. The shape feature is one of the easy ways to recognize an object in daily life. In this algorithm, the object is converted into some other format like straight lines in different directions. For image identification some useful tool is required, shape is used as one of the basic features of an image. However, sometimes it is too complex to represent and describe an image's shape properties. Representation of shape uses a descriptor that explores the property of the object easily, shape representation depends on interior content with the boundary of the object or shape boundary. Article [103] describes the efficient use of curvature scale space (CSS), Fourier descriptors, B-splines, polygonal approximation, and

invariant moments for the feature of shape. After applying different descriptors it was shown that the efficiency of the Fourier descriptor is higher in comparison to CSS in [1]. A new method based on partial shape matching (PSM) is developed for x-ray image retrieval [90]. Features like shape signature, moments, curvature, shape context, shape matrix, signature histogram, shape invariant, spectral features, etc. [104]. With the help of these features, performance has been measured. In the paper [105], real-life implementation is shown by tracking humans based on shape features in a crowded area, with the help of other additional features complete information can be achieved.

In the medical imaging field shape features play a vital role in finding different diseases [5]. In the last few years, many methods have been developed. Contour-based is one of them. This algorithm is concerned with the boundary information of an object. Another technique is the Fourier Descriptor Method. This method has been utilized for closed curve shape feature extraction and disjointed contour shapes. Contour-based methods are performed better compared to other methods. Shape-based image retrieval is the measuring of similarity between shapes represented by their features. The shape is an important visual feature and it is one of the primitive features for image content description. Shape content description is difficult to define because measuring the similarity between shapes is difficult. Therefore, two steps are essential in shape-based based image retrieval, they are feature extraction and similarity measurement between the extracted features (Fig. 14). The following Fig. 14 describes the different shape description techniques.

2.6.1 One-dimensional function for shape representation

One-dimensional functions for shape representation in image processing can be used to capture essential information about the shape of an object or contour in a concise manner. This function which is derived from shape boundary coordinates is also called shape signature [230, 231]. The shape signature usually captures the perceptual feature of the shape. Complex coordinates, Centroid distance function, Tangent angle (Turning angles), Curvature function,

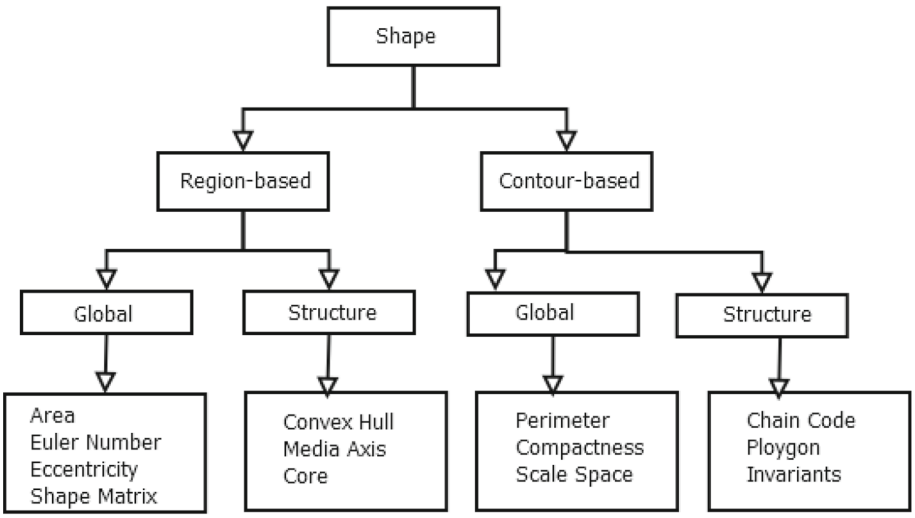

Fig. 14 An Overview of Shape Description Techniques

Area function, Triangle-area representation and Chord length functions are the commonly used shape signatures.

Shape signature can describe a shape all alone; it is also often used as a preprocessing to other feature extraction algorithms, for example, Fourier descriptors, and wavelet description. In this section, the shape signatures are introduced.

2.6.2 Complex coordinates

A complex coordinates function is simply the complex number generated from the coordinates of boundary points, $P_n(x(n), y(n)), n \in [1, N]$:

$$z(n) = [x(n) - g_x] + i\left[y(n) - g_y\right] \tag{31}$$

where $\left(g_x, g_y\right)$ is the centroid of the shape, given by (31).

2.6.3 Centroid distance function

The centroid distance function is expressed by the distance of the boundary points from the centroid $\left(g_x, g_y\right)$ (32) of shape

$$r(n) = \left[(x(n) - g_x)^2 + \left(y(n) - g_y\right)^2\right]^{1/2} \tag{32}$$

Due to the subtraction of the centroid, which represents the position of the shape, from boundary coordinates, both complex coordinates and centroid distance representation are invariant to translation.

2.6.4 Tangent angle

The tangent angle function at a point Pn(x(n); y(n)) is defined by a tangential direction of a contour

$$\theta(n) = \theta_n = \arctan \frac{y(n) - y(n - w)}{x(n) - x(n - w)} \tag{33}$$

since every contour is a digital curve; w is a small window to calculate (n) more accurately.

2.6.5 Contour curvature

Curvature is a very important boundary feature for humans to judge the similarity between shapes. It also has salient perceptual characteristics and has proven to be very useful for shape recognition. In order to use K(n) for shape representation, we quote the function of curvature, K(n), as:

$$K(n) = \frac{\dot{x}(n)\ddot{y}(n) - \dot{y}(n)\ddot{x}(n)}{\left(\dot{x}(n)^2 + \dot{y}(n)^2\right)^{3/2}} \tag{34}$$

Therefore, it is possible to compute the curvature of a planar curve from its parametric representation. If n is the normalized arc-length parameter s, then the above equation can be written as:

2.6.6 Chord length function

The chord length function is derived from the shape boundary without using any reference point. For each boundary point p, its chord length function is the shortest distance between p and another boundary point p' such that line pp' is perpendicular to the tangent vector at p.

The chord length function is invariant to translation and it overcomes the biased reference point (which means the centroid is often biased by boundary noise or defections) problems. However, it is very sensitive to noise, there may be drastic burst in the signature of even smoothed shape boundary.

2.6.7 Triangle-area representation

The triangle-area representation (TAR) signature is computed from the area of the triangles formed by the points on the shape boundary. The curvature of the contour point (x_n, y_n) is measured using the TAR as follows.

For each three consecutive points $P_{n-t_s}\left(x_{n-t_s}, y_{n-t_s}\right)$, $P_n(x_n, y_n)$, and $P_{n+t_s}\left(x_{n+t_s}, y_{n+t_s}\right)$, where $n \in [1, N]$ and $t_s \in [1, N/2 - 1]$, N is even. The signed area of the triangle formed by these points is given by:

$$\mathrm{TAR}\,(n, t_s) = \frac{1}{2}\begin{vmatrix} x_{n-t_s} & y_{n-t_s} & 1 \\ x_n & y_n & 1 \\ x_{n+t_s} & y_{n+t_s} & 1 \end{vmatrix}$$

When the contour is traversed in counterclockwise direction, positive, negative, and zero values of TAR mean convex, concave, and straight-line points, respectively.

2.6.8 Splitting methods

Splitting methods work by first drawing a line from one point on the boundary to another. Then, we compute the perpendicular distance from each point along the boundary segment to the line. If this exceeds some threshold, we break the line at the point of greatest distance. We then repeat the process recursively for each of the two new lines until we don't need to break anymore. See Fig. 15 for an example.

Sometimes this is recognized as the "fit and split" algorithm. For a closed contour, we can find the two points that lie farthest apart and fit two lines between them, one for one side and one for the other. Then, we can apply the recursive splitting procedure to each side.

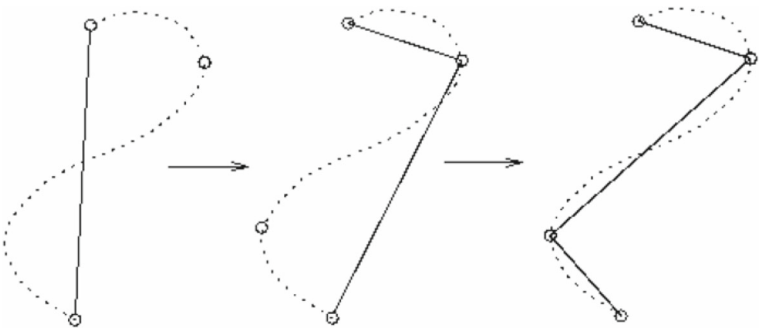

Fig. 15 Splitting methods for polygonal approximation

2.6.9 Fourier descriptor method

In image processing, the Fourier descriptor is used especially for boundary shape information. This comes from the Mathematical word Fourier and is used to compute the shape complexity and also another geometric attribute. The shape signature function is used for the computation of the Fourier descriptor. After calculating boundary pixels we group each pixel to get exact information of the object. It is represented by (35).

$$P = \{(f(n), g(n))|n \in [1, M]\} \tag{35}$$

where M = Count of pixels at the boundary.

Centroid distance is measured through the use of the signature function of another shape. This function is represented in terms of overall performance. The coordinate of the centroid is represented by (f_c, g_c), where f_c and g_c can be represented as in Equation 36 respectively.

$$f_c = \frac{1}{N}\sum_{n=0}^{M-1} x(n) \; and \; g_c = \frac{1}{N}\sum_{n=0}^{M-1} y(n) \tag{36}$$

distance for $(t+1)^{th}$ is represented by r(t) is from the centroid (x_c, y_c), then $r(t)$ can be denoted by (37).

$$\mathrm{r(L)} = \left([\mathrm{x(L)x_c}]^2 + [\mathrm{y}(\ell) - \mathrm{yc}]^2\right)^{\frac{1}{2}}, \quad \mathrm{L} = 0, 1, \ldots\ldots \mathrm{M} \cdot 1 \tag{37}$$

Where a_n = Fourier Transformed Coefficients of r(n). The following Equation 38 shows Fourier transformed coefficients.

$$a_n = \frac{1}{N}\sum_{k=0}^{M-1} r(n) \exp\left(\frac{-2j\pi nl}{M}\right), n = 0, 1, \ldots . M-1 \tag{38}$$

Standardized Fourier transformed coefficients thatares represented by following (39). Where b_n = Fourier transformed coefficients.

$$b_n = |\frac{a_n}{a_0}| \tag{39}$$

Equation 40 represents the vector of shape feature as shown below.

$$FD = \{b_i|i \in [0, \frac{M}{2} - 1]\} \tag{40}$$

2.6.10 Hu-moment shape features

Rotation, translation, scaling, and (RTS) are some Algebraic Moment invariant properties of Hu-Moment related to connected regions. Moment invariants output is used for feature vector creation [8]. For class and shape identification region property has been used. Hue describes the set of properties of a region. This technique is old and standard for producing invariants. It was proposed by Hu.

Suppose I is an image, a+b central moments or I form as below (41)

$$\mu_{a,b} = \sum_{x,y} (x - x_c)^a (y - y_c)^b \tag{41}$$

(x_c, y_c) represents centre of object. Following (42) and (43) represent central moments for scale-independent nature.

$$\eta_{a,b} = \frac{\mu_{a,b}}{\mu_0^2, 0} \tag{42}$$

$$\gamma = \frac{a+b+2}{2} \tag{43}$$

The above moments are useful in feature extraction and pattern recognition, which are free from orientation, size, and position.

2.6.11 Application of shape feature extraction in image forensic and security

Shape feature extraction is a valuable tool in image forensic and security applications. These techniques are helpful in the visualization of contours, shapes, and geometric properties in images. Various articles based on shape feature extraction show the different applications of shape features in the field of image forensics. Different applications in various fields are given below:

- Shape-based image retrieval systems use shape features to search and retrieve images from large databases based on their shapes or contours, assisting investigators in finding relevant visual content.
- In automated license plate recognition (ALPR) systems to identify vehicles and read license plate numbers for security and law enforcement applications shape feature is used.
- We also use shape features for object recognition, allowing for the identification of specific objects or patterns within images and videos, even when they appear at different orientations or scales.
- In surveillance and security applications, shape features are used to identify and track vehicles based on their unique shapes, such as license plate recognition.
- Shape features are employed in object tracking systems, allowing for the continuous monitoring and tracking of objects or individuals in video streams.

2.7 Corners/interest points

A sudden change in the direction of the edge in image processing is called “Corner". Initially to find out the corner of image detection of edge was required. [6].

But as time goes on new algorithm has been developed. In recent times there is no need for edge detection to find out the corner of an image. For example to find a high level of curvature in an image gradient. The intersection of two edges is called a corner, and the Word “corner" is used by tradition. This term is used by many methods like camera tracking and visual odometry. Many new techniques related to this method have been developed For corner detection Fast Event-based Corner Detection method performs very accurately in an event stream. There are mainly two types of Corner detection algorithms. First, the corner detection algorithm depends on the edge and another one is dependent on the gray change. In the first method difference between two adjacent code value is calculated, based on this value extraction of the chain code for the image edge is done, the difference between these two values determine whether it is a corner or not, this method takes more time and space to perform the calculation [106]. In the second method, for corner detection, we calculate curvature and gradient. To perform this we can use the Harris algorithm, the SUSAN algorithm,

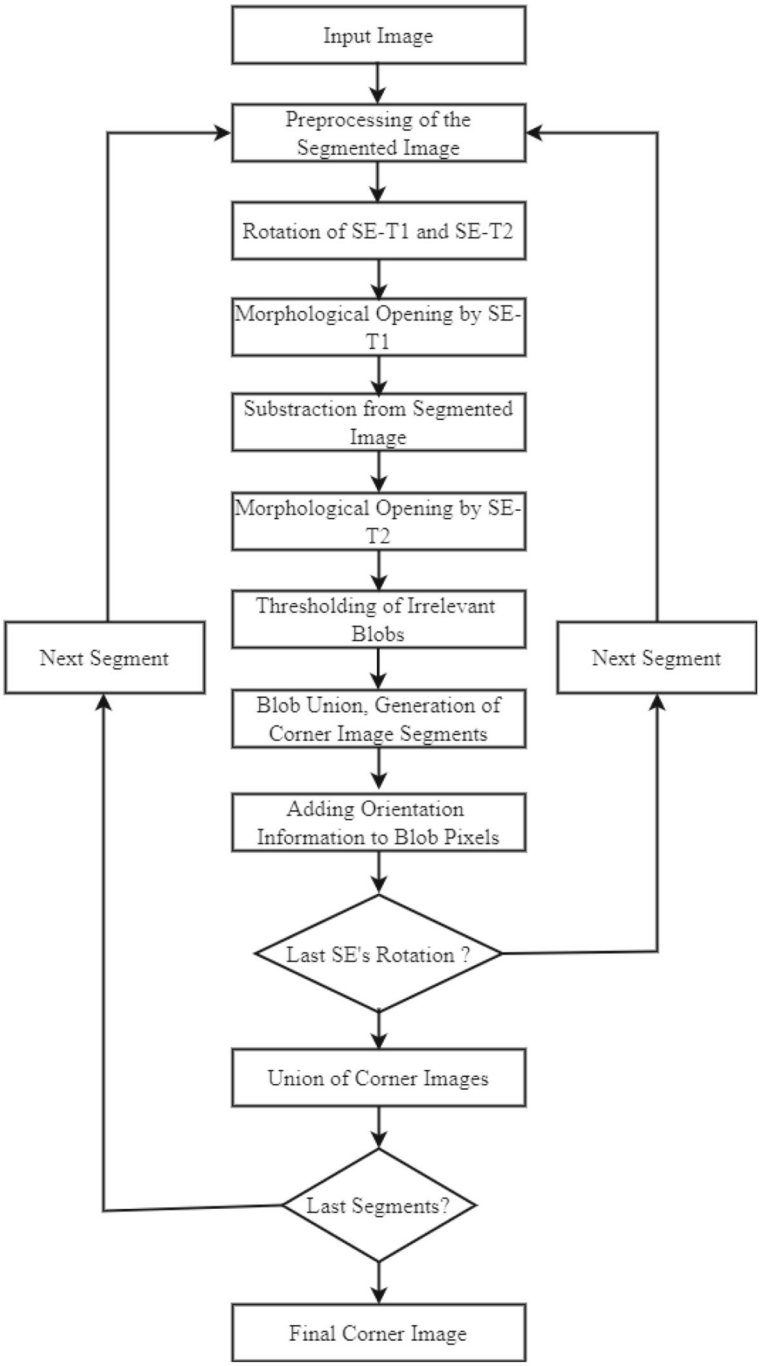

Fig. 16 Flowchart of the Corner Detection Algorithm

Moravec algorithm [107]. In the paper [265] image corner detection from segmented areas using mathematical morphology employing paired triangular structuring elements. The algorithm highlights TriSE02 detects the inner corners of a segmented area and saves information regarding each cornerâ€™s angular orientation and position. The developed method is used to identify conjugate corners in stereoscopic dual-energy X-ray images which is generated by an experimental system for security of aviation screening. Figure 16 shows the flow diagram of the corner detection algorithm.

2.7.1 Corner detection by FAST method

In the FAST method, we assume a pixel is a corner. If there is a center pixel with a brighter or darker pixel having a threshold value (typically, b = 9 on a circle of radius 3 with 16 pixels) compare to their circumference values. Where n represents contiguous pixels along the circumference of a circle. Figure 17 describes the FAST algorithm local detection area.

In the event camera brightness is encoded in the form of temporal contrast. Here event e = (a,b: t, pol) is triggered at a pixel (a,b) at time t if brightness I equal to contrast threshold C. Relationship between variables is shown in Equation 44.

$$I(a, b, t) - I(a, b, t - \Delta t) = pol.C \tag{44}$$

where $t - \Delta t$ represents the time when the last event at that pixel was triggered, the change in brightness is shown by the polarity of the event. Following are Surface of Active Events (SAE), SAE shows the function derived by the timestamp of the latest event at every pixel:

$$SAE : (a, b) \mapsto t \tag{45}$$

The information gained from the comparison of the circumference pixel with the center pixel is not given useful information. That is why a different comparison method is required. It is required to check the current events with the help of the local neighborhood.

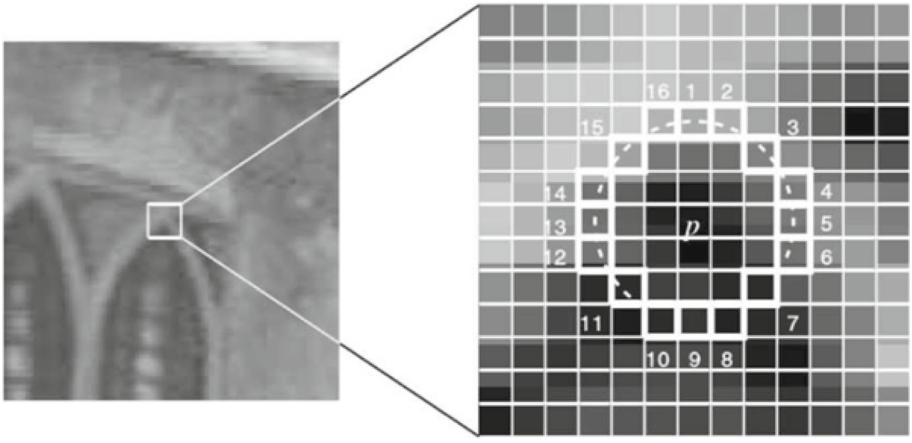

Fig. 17 FAST Local Detection Area

2.7.2 Shearlet corner detection

The corner is placed in an object which has large intensity changes in all directions. In the last decades, many corner detection algorithms have been developed. The Wavelet method is one of them which is used for corner detection [7]. For corner detection, authors Harris and Stephens developed a method based on the image gradients structure tensor which is also called the autocorrelation matrix. In Shearlet Corner Detection, a weighted sum of Shearlet coefficients with respect to shears is done to find a good corner. The shearlet response of a point can be calculated by, how perpendicular is the orientation of the shear with the orientation of the shear with the maximum shearlet response for that point.

CM stands for Cornerness Measure which is computed for a point $m \in \tau$ and j = fixed scale shown in Equation no 46.

$$CM_j(m) = \sum_{u \in W(m)} \sum_{k} |SH(j,k,u)| sin(|\theta_k - \theta_k max|) = 1 \tag{46}$$

where SH(j, k, u) represents the discrete shearlet transform coefficient for a point u in a neighborhood of m, at scale j and shearing k.

2.7.3 Canny edge detector

Canny Edge Detector depends (CSS) [13] corner detector. The thickness of the edge in the canny detector depends on the orientation of the angel. A thick edge has been produced when edge position 135 ° or 45°. In this process, two similar gradient value is produced, because of the same brightness level on either side of the edge. When the edge line is thinned corresponding only one pixel wide, it is meaning that Non-maximum suppression. For neighboring pixels information Cannyâ€™s non-maximum suppression uses the direction of the gradient at an edge point. Mainly two information border gradient direction and intensity [92] is extracted. For the convolution of an image, the Canny operator uses templates of different directions.

In the paper [278] the original Canny algorithm, while effective, is computationally demanding and has higher latency due to its frame-level processing. The proposed mechanism enables the Canny algorithm to be applied at the block level without sacrificing edge detection performance. To overcome the challenges of applying the Canny algorithm at the block level, the authors introduce a distributed Canny edge detection algorithm. This new approach adaptively computes edge detection thresholds based on the type of block and the local distribution of gradients within each block. The algorithm consists of several key steps:

- Calculation of Gradients
- Gradient Magnitude and Direction
- Non-Maximal Suppression (NMS)
- Threshold Computation

The Canny algorithm is a multi-stage process that aims to identify edges by analyzing gradient information while suppressing non-maximal responses. It then applies thresholds to distinguish between strong and weak edges. This algorithm is known for its effectiveness in detecting edges, especially in the presence of noise, and has been widely used in image processing and computer vision applications. Figure 18 describes the canny edge detection using lenna image.

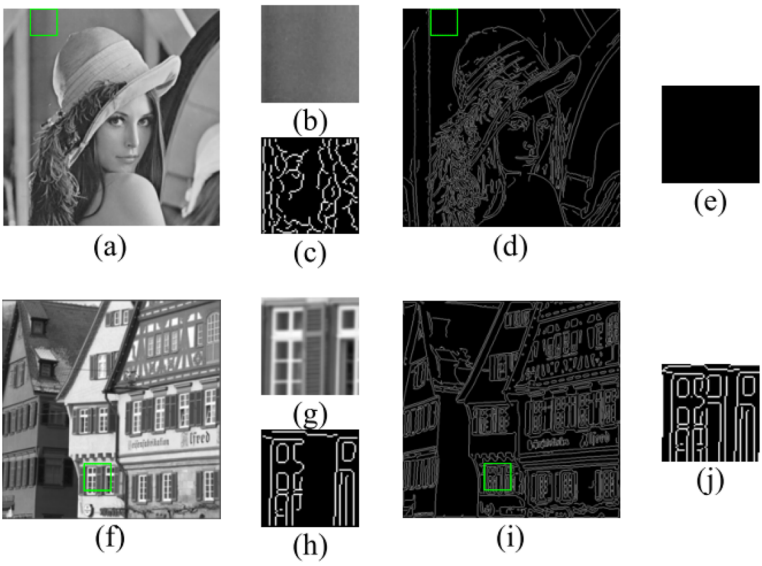

Fig. 18 (a), (f) Original Lena and Houses Images; (b), (g) Zoomed-in blocks of the Lena and Houses Images; (c), (h) Edge Maps of Block Images in (b) and (g) obtained by Applying the Classical Canny Edge Detector to Blocks Separately; (d), (i) Edge Maps Using the Canny Edge Detector Applied to the Entire Lena Image and Houses Image; (e), (j) Zoomed-in Blocks of the Edge Maps (d) & (i) [278].

Following are step-by-step description of the Canny Edge Detector Algorithm:

- The First step is Smoothing
- In the second step we find gradients
- Non-maximum suppression has been calculated in step third
- In the fourth step Double the thresholding value
- In the last step Edge tracking was done with the help of hysteresis

2.7.4 Harris corner detection algorithm

In the area of computer vision and image processing, the important operator used for the corner detection of an image is the Harris Corner Detector. This came into the picture first in 1988, when Chris Harris and Mike Stephens first time used this algorithm. The author in the article [109] implemented the Scale-invariant feature theory with the Harris feature detection algorithm. This paper [108], concluded that the Harris corner algorithm is dependent on a threshold value. A moderate value of threshold value finds the best corner information. A very large value will lose some corner information, and a very small one will detect wrong corners. There is another Harris that has been based on algorithm [110] a novel encrypted image retrieval scheme has been proposed based on Harris Corner preference and optimization of LSH in cloud computing. Harris corner detection is dependent on how strong the gradients are in all directions for corner [11]. The algorithm contains the steps shown below:

Gradient computation The derivatives for x and y axis at each pixel are computed based on the vertical and horizontal gradients, the ${D_x}^2 = D_x \times D_x$, ${D_y}^2 = D_y \times D_y$ and $D_x \times D_y$ values.

Gaussian smoothing In this step, the application of Gaussian smoothing is done over the previously obtained three gradients ($D_x{}^2$, $D_y{}^2$, and $D_x \times D_y$).

Harris measure Harris measure is the detection of whether a pixel belongs to a corner or not. Eigenvalues of the calculated matrix show that a pixel belongs to which reason flat, edge, or corner region. For matrix calculation following Equation 47 is used.

$$S = \begin{pmatrix} I_x^2 & I_{xy} \\ I_{xy} & I_y^2 \end{pmatrix} \tag{47}$$

Thresholding A moderate value of the threshold will give the best result. In Harris measure, a corner pixel is located by a high positive value is used to locate a corner pixel.

2.7.5 SIFT

The scale-invariant feature transform (SIFT) was first introduced by Lowe in 2004 [233] and has become one of the most widely used methods for detecting and describing distinctive features in images, especially for tasks such as image stitching, object recognition, and 3D reconstruction. Since then, many variations have been presented, such as BF-SIFT [232]. The key steps involved in the SIFT feature extraction process are as follows:

- Scale-Space Extrema Detection
- Keypoint Localization
- Orientation Assignment
- Keypoint Descriptor
- Feature Matching

Due to the advent of deep learning and more advanced feature extraction techniques like CNN-based descriptors, SIFT's popularity has somewhat decreased in recent years. However, it remains a valuable tool, especially in scenarios where robustness to geometric transformations and invariance to changes in scale are crucial. The SIFT feature detector operates by finding the difference of Gaussians (DoG), which is an approximation of the Laplacian of the Gaussian(LoG). The SIFT descriptor contains 128 bin values and it is represented as a vector. Equation (48) shows the convolution of the difference between two Gaussians that were computed at different scales with an image I(x, y).

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma) * I(x, y)) \tag{48}$$

where G represents a Gaussian function.

2.7.6 KAZE

KAZE was proposed by Alcantarilla et al in the year 2012 which was a feature detector based on a nonlinear scale-space through nonlinear diffusion filtering [234]. Similar to SIFT, KAZE is used for object recognition, image stitching, and image matching tasks. It computes distinctive feature descriptors that can be used to match key points between images and perform further analysis. The KAZE descriptor achieves rotation invariance by finding the dominant orientation in a circular region that is centered at the location of the feature point. Equation 49 shows the standard nonlinear diffusion formula.

$$\frac{\partial L}{\partial t} = \text{div}(c(x, y, t) \cdot \nabla L) \tag{49}$$

where c represents the conductivity function, "div" is divergence, ∇ is the gradient operator, and L is image luminance.

2.7.7 Application of corner feature extraction in image forensic and security

In image forensic and security applications, the corner feature extraction technique plays very important as it helps identify and characterize distinctive corner points or interest points within images and videos. These corners are further used as key reference points for various tasks related to image forensics, image analysis, object recognition, and security. The versatility of corner features makes them a fundamental component of computer vision and image processing, enabling a wide range of tasks and applications that involve the analysis and interpretation of visual data Here are several specific applications of corner feature extraction in these fields:

- In the process of face recognition systems to locate facial landmarks, such as eyes, nose, and mouth corners, facilitating accurate identification and authentication corner features play an important role.
- Corner features are employed in forensic analysis to match photographs with sketches, aiding in the identification of suspects or missing persons.
- It is essential in creating large-scale image mosaics from multiple images, which is useful for wide-area surveillance and monitoring.
- Corner-based visual odometry and localization techniques enable robots and autonomous vehicles to navigate and map their environment, ensuring safe and reliable operations in security and defense applications.
- For object recognition, allowing for the identification of specific objects or patterns within images and videos we use the corner feature. These features provide distinctive landmarks that aid in matching objects against reference images or templates.
- Corner features help in identifying inconsistencies in geometric structures introduced during image forgery or tampering. Inconsistent corners may indicate areas of manipulation, such as cloning or perspective distortion.
- We use this feature in image-matching tasks, such as image retrieval, content-based image retrieval (CBIR), and reverse image search, which is used in security and forensic databases.
- In content moderation and filtering on the internet, corner features can be used to identify and filter out inappropriate or explicit content based on the presence of specific shapes or corners in images and videos.

2.8 Keypoint features

In image processing, keypoints play an essential role in many computer vision tasks. This is used in image processing applications due to their capability to represent significant image information efficiently and accurately. Keypoints play a very important role because they come up with a compact and revealing representation of an image. This property makes it simple to analyze and compare images for various tasks such as image matching, object recognition, and image alignment.

2.9 Keypoint feature extraction techniques

In the current research of image processing somehow it is very challenging work to design a keypoint-based algorithm [243] to detect forgeries involving small smooth regions.

For image feature extraction using different keypoint descriptors is as follows:

2.9.1 Scale invariant features transform (SIFT)

Scale Invariant Features Transform (SIFT) [233] is a technique for extracting distinctive invariant features from images that can be used to perform reliable matching between different views of an object or scene. In the image processing area researcher uses the SIFT algorithm because of its performance and robustness to changes in scale, rotation, and illumination. SIFT is divided into four parts.

Scale-space extrema detection This is the first step of the SIFT algorithm. In this step detection of key points or interest points that are stable across different scales has been taken. To identify these extrema, We calculate the Difference of Gaussians (DoG) between adjacent scales (Fig. 19). DoG of the image is represented as

$$D(x, y, \sigma) = D(x, y, k_i\sigma) - D(x, y, k_j\sigma) \tag{50}$$

Keypoint localization After keypoint identification, localize the key points accurately and refine weak localized points. We localize their positions by fitting a 3D quadratic function to the DoG scale space.

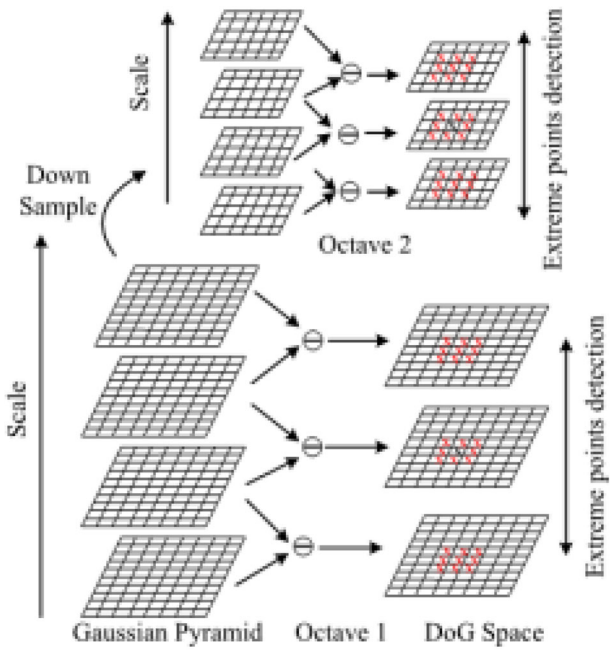

Fig. 19 Gaussian Pyramid and the DoG Pyramid

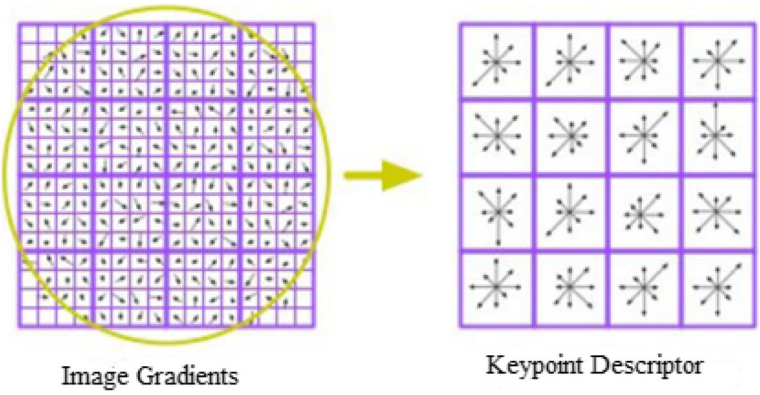

Fig. 20 Representation of Descriptors

Orientation assignment In this step, the SIFT algorithm determines a presiding orientation for every key point to make the descriptor rotation-invariant. We have to use the following equations 51 and 52 for orientation and magnitude calculation.

$$\theta(x, y) = atan2(L(x, y+1) - L(x, y-1), L(x1, y)L(x1, y)) \tag{51}$$

$$M(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \tag{52}$$

The top of the histogram correlates with the dominant orientation of the key point.

Descriptor generation In the fourth stage for every key point, computation of the descriptor vector is done and the created descriptor is highly distinctive and partially invariant illumination changes, 3D viewpoint, etc. SIFT creates a local feature descriptor for every key point that encodes information about the key point's surroundings. SIFT performs a ratio test to filter out wrong matches. Performing the ratio test we compare the distance from the closest match to the distance of the second-closest match. After outlier elimination, the rest of the key point matches can be utilized for different computer vision tasks, such as image stitching, image alignment, or object recognition (Fig. 20).

2.9.2 Speed up robust features (SURF)

In [255], Herbert Bay et al. proposed the SURF algorithm which is similar as based on the principles and steps of the SIFT algorithm, but the details in each step are different. Speed Up Robust Features (SURF), is a novel scale- and rotation-invariant interest point detector and descriptor. The SURF algorithm is based on the Hessian Matrix and uses very basic approximation. The Hessian matrix H(x, σ) can be defined at a point x and scaleσ, for image I as follows:

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \tag{53}$$

Where $L_{xx}(x, \sigma) = I(x) * \frac{\partial^2 g(\sigma)}{\partial x^2}$, $L_{xy}(x, \sigma) = I(x) * \frac{\partial^2 g(\sigma)}{\partial xy}$, $L_{yy}(x, \sigma) = I(x) * \frac{\partial^2 g(\sigma)}{\partial y^2}$, $L_{xx}(x, \sigma)$ represents convolution of the image with the second derivative of the Gaussian g(σ) at scale σ

To reduce computational time, it relies on integral. The descriptor describes a distribution of Haar-wavelet responses within the interest point neighborhood. The indexing step is based

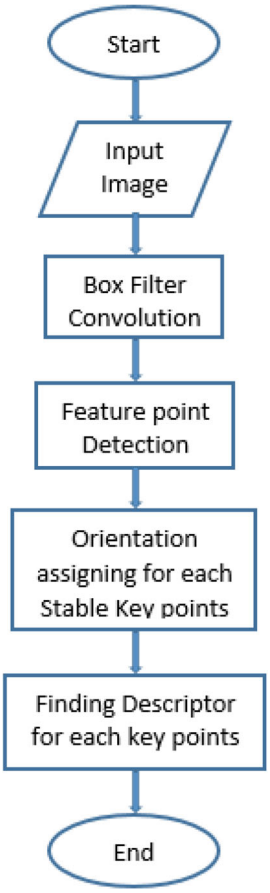

Fig. 21 SURF Algorithm Flowchart

on the sign of the Laplacian, which increases the matching speed and the robustness of the descriptor (Fig. 21).

ORB (oriented FAST rotated and BRIEF) feature extraction ORB was developed by Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary R. Bradski in 2011. ORB is known for robustness and computational efficiency, making it well-suited for real-time applications. In the paper [256] ORB image-matching algorithm is proposed. This ORB-based algorithm is divided into three steps: The first step is feature point extraction, the second step is generating feature point descriptors, and the third step is feature point matching. Figure 22 shows the working flow of the ORB image-matching algorithm.

In feature point detection first, we select an image with pixel p and brightness I_p. We consider the brightness threshold as T. pixel p is considered as a feature point if the brightness of consecutive N points on the selected circle is greater than I_p + T or less than I_p - T.

ORB algorithm enhances the performance of the original FAST algorithm by calculating the Harris response values for the original FAST corner points arranging them according to the gray value and taking the first N points. The following equation represents the Harris

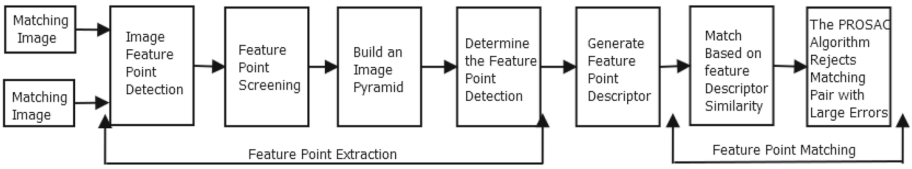

Fig. 22 Flow chart based on ORB Image Matching Algorithm

response value calculation formula

$$R = det(M) - k(trace(M))^2 \tag{54}$$

$$M = \sum w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \tag{55}$$

where R is the Harris response value, M is a 2*2 matrix,

After getting the Oriented FAST feature points, the ORB algorithm uses the enhanced BRIEF algorithm for descriptors calculation of every point. BRIEF is a binary vector descriptor whose vector consists of a number of 0 and 1:

$$\tau(p; x, y) = \begin{cases} 1, & \mathrm{p}(x) < p(y) \\ 0, & \mathrm{p}(x) \geq p(y) \end{cases} \tag{56}$$

where p(x) represents the gray value at the field x around the image feature point, and p(y) is the gray value at the field y around the image feature point.

In the third step, we find feature point matching. The previous step gives rotation and scale information about image feature points. After that, it becomes mandatory to regulate the similarity between the feature point descriptors in the two different time images to determine their match. Lets consider feature points x_m^t, where m = 1,2,.....M is extracted in image I_t and feature points x_n^t, n = 1,2,...N is extracted in image I_{t+1}. This method calculates the distance between every feature point x_t^t and x_{t+1}^n. Compared to Brute-Force Matcher this method is more efficient. Additionally, the PROSAC algorithm is utilized to remove some matching pairs with large matching errors, which improves the matching accuracy.

Binary robust invariant scalable keypoints (BRISK) BRISK was developed by Stefan Leutenegger, Margarita Chli, and Roland Siegwart in 2011, BRISK is designed to be both robust and computationally efficient. The BRISK detector evaluates the original scale of each keypoint in the continuous scale space. In the BRISK framework, the scale-space pyramid layers consist of n octaves c_i and n intra-octaves d_i, for i = 0, 1, . . . , n âˆ’ 1 and typically n = 4. In BRISK, commonly we use the mask of size 9-16, which intrinsically requires minimum 9 successive pixels in the 16- pixel circle to either be sufficiently brighter or darker than the central pixel for the FAST criterion to be satisfied.

In Key Point Description Given a set of key points (consisting of sub-pixel refined image locations and associated floating-point scale values), the BRISK descriptor is composed as a binary string by concatenating the results of simple brightness comparison tests. In the BRISK algorithm, the local gradient is assumed to obliterate each other and the local gradient is not mentioned in the computation of the final gradient mode. So, by the set L final mode direction of the feature points can be estimated.

$$g(x) = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \frac{1}{l} \dot{\sum}_{(p_i, p_j) \in W(m)} g(p_i, p_j) \tag{57}$$

where l represents the length of a subset of long-distance pairings L. g_x and g_y are gradients sum of the long-distance point pair set on respectively x-axis and y-axis direction.

For building a descriptor with rotation invariance, for θ angle rotation around feature point k is calculated as:

$$\theta = octan2g(p_i, p_j) \tag{58}$$

In the descriptor matching part, the matching of two BRISK descriptors is a simple computation of their Hamming distance as done in BRIEF. The Hamming distance is calculated using a bitwise XOR operation with the help of two values. The number of bits different in the two descriptors is a measure of their dissimilarity. Notice that the respective operations reduce to a bitwise XOR followed by a bit count, which can both be computed very efficiently on todayâ€™s architectures. Output is "0" if the corresponding bit is the same otherwise "1". Let's assume X and Y are two BRISK descriptors. Hamming distance is calculated by following equations.

$$HD(X, Y) = \sum_{i=1}^{N} x_i \bigoplus \gamma_i = \sum_{i=1}^{n} b(x_i, \gamma_i) \tag{59}$$

Where x_i, γ_i are the i^{th} bits of the descriptors X and Y respectively.

Figure 23 shows the different steps involved in the keypoint detection Process.

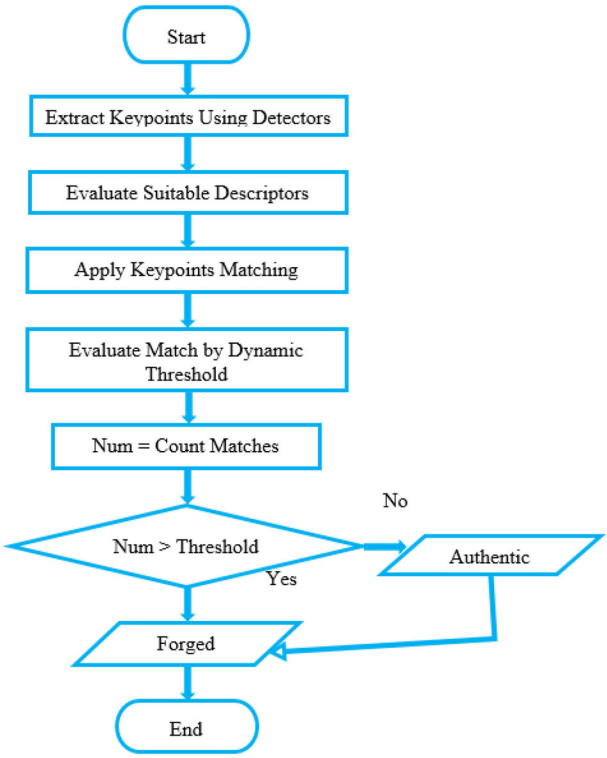

Fig. 23 Forgery Detection with Keypoint Feature

In the paper [238] the author proposes a novel keypoint-based copy-move forgery detection method for small smooth regions. In this process segmentation of the tempered image takes place into nonoverlapping and irregular superpixels, and then the classification of superpixels into texture, smooth firm texture based on local information entropy has been done. In superpixel segmentation, the author uses the entropy rate superpixel (ERS) algorithm for the segmentation of the host image into meaningful irregular superpixels. In this process, we assign adaptively the optimal number K of superpixels according to the tampered image size and K as follows:

$$K = \frac{n_1}{n_2} \times \lfloor \frac{Max(M, N)}{5} \rfloor \tag{60}$$

In superpixel classification, we first classify the superpixels into smooth, texture, and strong texture ones according to their local information entropy, and then define adaptively the Hessian response threshold based on superpixel content. Keypoint feature construction is a step for CMFD and accuracy detection. After matching a set of image key points x_m has been obtained, it is important to cluster these points to distinguish the different matched regions. RANSAC algorithm is used to determine the type of geometrical transformation used between the copy-moved version and the original area.

Comparison and future scope After comparing existing state-of-the-art, the results of this paper conclude that this proposed approach can achieve more accurate detection results for copy-move forgery images under various challenging conditions, such as JPEG compression, geometric transforms, and additive white Gaussian noise. Higher computational complexity is a problem of this proposed copy-move forgery detection method in the future we can use this approach to detect regions with nonaffine transformations. Following Fig. 24 demonstrates the copy-move forgery detection using keypoint feature

In the paper author "Sunitha, K., and A. N. Krishna" proposed a novel keypoint-based method. This method extracts hybrid features to detect efficient copy-move forgery techniques using key points and hierarchical clustering methods.

- In this paper an efficient keypoint-based copy move forgery detection method using a hybrid feature extraction method (i.e., SURF is used to extract the feature and SIFT is utilized as feature descriptor).
- In the clustering method agglomerative hierarchy is used to map the feature. After that, an outliers removal algorithm (RANSAC) is used to transform the image.
- If we compare the proposed method (recall (92.5), FPR (8.9), and F1-score (91.7) to the existing one with respect to recall, FPR, and F1-score. The outcome shows that the proposed model performance is better.

Comparison and future scope From the experiment output, it can be seen that EKF-CMFD attains a FPR performance of 8.9% which is 8.37% better than the existing CMFD model presented by Zandi et al. Then further, EKF-CMFD attains a TPR performance of 92.5% which is 3.36% better than the existing CMFD model presented by Raju et al. Another EKF-CMFD attains an F1-score performance of 91.7% which is 0.9% lesser than the existing CMFD model presented by Raju et al. In the future with further optimization, we can improve the accuracy and TPR and F1 score of the EKF-CMFD.

Semma, Abdelillah, et al [15] proposed a writer identification system that uses key points for feature extraction from handwriting and feeding small patches around these key points with the help of convolutional neural network (CNN)for classification and feature learning. Three datasets namely QUWI, IAM, and IFN/ENIT are used. For end-to-end classification, the identification process uses the following steps.

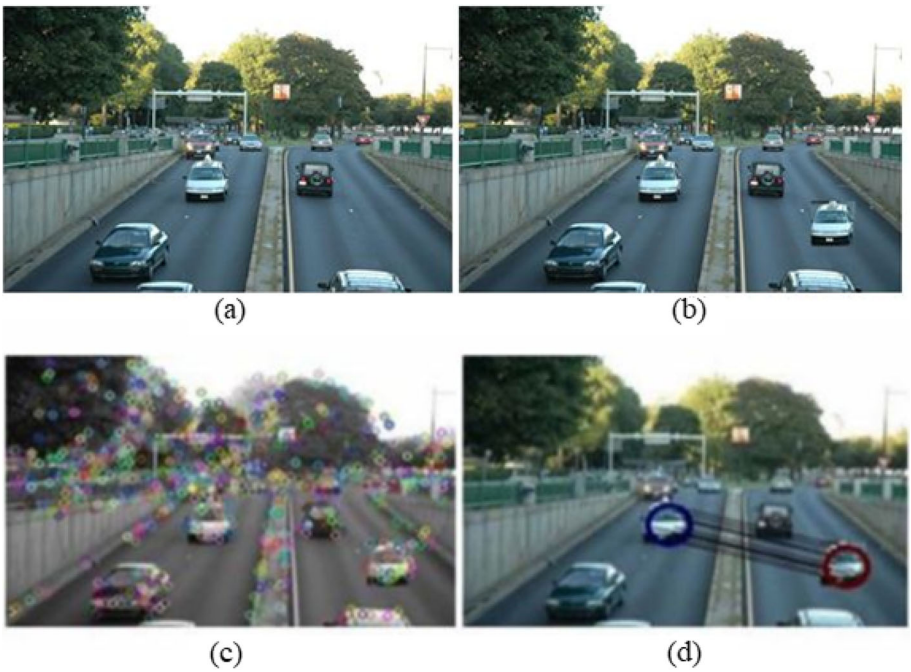

Fig. 24 Copy-Move Forgery Detection (a) Authentic Image (b) Tampered Image (c) Keypoints Detection in the Tampered Image (d) Forgery Detected Image [280]

- Pre-process the images in the training and test sets.
- With the help of sampling methods (Harris, FAST, or random sampling) extract patches of handwriting from training and test images.
- In this step we train the CNN model with the help of patches of writing in the training set.
- We get the Classified images of writing in the test set by feeding the corresponding writing patches to the trained CNN model.

In the Harris Corner detector, we use the Hessian matrix for detection of interest points. For example, given pixel p in the image, the intensity variation in the x, y, and xy directions.

$$H(p) = \begin{bmatrix} I^2_{x(p)} & I_{xy(p)} \\ I_{xy(p)} & I^2_{y(p)} \end{bmatrix} \tag{61}$$

Interest points were detected using FAST. In which we choose a candidate point p with intensity I_p, selecting a threshold value t and considering a circle of 16 pixels with p as center. Representation of scores of adjacent candidate key points by (62):

$$score\left(p\right) = \sum_{k=1}^{16} |I_k - Ip| \tag{62}$$

Training of network in this model in an end-to-end manner with writer IDs of patches as the target class labels.

Comparison and future scope The proposed technique reports the highest identification rate (99.5%) on the complete set of 657 writers. After analysis, IFN/ENTI and QUWI datasets are comparatively fewer as compared to those employing the IAM dataset. Considering a relatively larger number of unique writers in the QUWI dataset (1017), a Top-1 identification rate of nearly 100% is indeed very promising. For future research writer identification, the system can also be explored for other similar problems like identification of writers' demographics from handwriting, and writer verification. A major drawback of the described technique is that it detects a large number of key points that are very close to one another.

A new keypoint-based CMFD technique has been proposed by Yue, Guangyu, et al. [194]. In this paper, a novel keypoint-based CMFD method: second-keypoint matching, and double adaptive filtering (SMDAF) are used. The double adaptive filter (DAF) algorithm is based on the KANN-DBSCAN clustering algorithm and AdaLAM algorithm and for wrong filter keypoint matches are proposed, according to the distinct distribution of key points in each image. In this proposed method following steps are used:

- AdaLAM outpoint filter which takes a large number of matched key points as input.
- We use KNN-DBSCAN algorithm which is based on the DBSCAN clustering algorithm Which has eps and minpts as key parameters.
- SIFT descriptor is used to extract key points. For each keypoint, P_j in the keypoint set KP, where j ϵ $\{1, 2..., s\}$, The extraction of local binary pattern features has been done, that contain rotation invariance weight uniformity.
  The following (63) and (64), Show the details where S represents the number of keypoints, and j ϵ (0, S):

$$KP = \{P_1, P_2, P_3, \ldots, P_S\} \tag{63}$$

$$P_j = \{PT_j, A_j, S_j, D_j\} \tag{64}$$

- In second keypoints matching, for matching keypoints in multi-copyâ€“move forgery we use the following expression:
  $P_o^L \rightarrow P_1^R$: P_o^L match P_1^R as the second similar keypoint, but not vice versa. $P_o^L \leftrightarrow P_1^R$: The two keypoints match with each other.
- Next step is double adaptive filtering: Which contain two steps: first is First adaptive filtering and second is Adaptive clustering.

Comparison and future scope The HFPM method grows to 0.500 and 0.901 of the F1 score. However, program errors occur when detecting 1313 authentic and 838 forgery images. However, the F1 score on the MICC-F220 dataset is as disappointing as BusterNet, which only reaches 0.746. On the MICC-F220 dataset, the F1 score of the proposed method can reach 0.904, which held a slender lead to HFPM (0.901) and SMDAF-SURF (0.869). After using the SMDAF-SURF and the proposed method, the F1 scores notably improved at the image level. On the CASIACMFD dataset, the result of SMDAF-SURF improves from 0.629 in DOA-GAN to 0.682, and the proposed method is even more excellent, achieving 0.714. Another deep learning-based method called DOA-GAN, gets better results on the CASIA-CMFD dataset, which F1 score is 0.629. In future work, we can improve the following aspects. keypoints extraction and the other is to design a better forgery localization method to locate the copyâ€“paste regions. Further, we can improve the stability of deep learning-based methods against attacks of postprocessing. The scope of improvement is also in keypoint-based and deep-learning methods.

2.9.3 Application of keypoint feature extraction in image forensic and security

Keypoint feature extraction, also known as interest point or salient point extraction, is widely used in image forensic and security applications to identify distinctive and stable points within images and videos. For keypoint feature extraction, different descriptors like SIFT, SURF, ORB, and BRISK have been used. Which performs very well with other methods. These keypoints serve as landmarks for various tasks related to image analysis, authentication, and tamper detection. Following are some specific applications of keypoint feature extraction in the field of image forensics and security:

- These features are used for efficient image retrieval and database search, making it easier to find relevant images in large collections, such as security camera archives.
- For image matching and similarity comparison, aiding in identifying duplicate or altered images in large databases keypoint feature is used.
- Keypoints play a vital role in detecting image forgeries, such as copy-paste operations or object insertion. Inconsistencies in keypoint distribution can indicate areas of manipulation.
- By using a keypoint we can identify and verify the authenticity of images. They help in ensuring that an image has not been tampered with or manipulated.
- It is helpful for automatic license plate recognition (ALPR) systems to locate and identify license plates, aiding in security and law enforcement applications.
- Utility of key points in biometric systems for facial recognition, iris recognition, and palmprint recognition to identify and authenticate individuals.

3 Different types of digital image tampering attacks

In the modern era digital images play a very important role in human life. Images are available on the internet, social media, and many other platforms. It is too complex to check whether an image object is real or not in modern days. Making an image forged is a very easy task these days, due to the availability of free software, people must know that seeing does not always imply believing. Sometimes you can not differentiate two images with bare eyes, detection of forgery has become very important today. Survey paper [226] presents useful facts about image and video forensics based on source camera identification. In the current AI-based age, deep learning plays a very important and crucial role in image manipulation and its detection. A systematic and step-wise survey has been carried out in the review [228]. This article represents a deep-learning-based method in a comprehensive way for image forensics techniques.

Most of the attackers use attacks like copy-move, slicing, re-sampling, resizing, noise variations and/or blurring, retouching, JPEG compression, lighting inconsistencies, and luminance non-linearities. These types of attacks are applicable to both images and video. Following are some important attacks.

3.1 Copy-move (cloning)

This is one of the commonly applied attacks given its simplicity and effectiveness. It concerns all techniques that manipulate an image by copying certain region(s) and pasting them into another place on the same image (or video). Following Fig. 25 is an example of copy-move forgery detection.

(a) (b)

Fig. 25 (a) Original Image (b) Copy-Move Forged Image [68]

Copy-Move Forgery (CMF) is a type of image tampering attack in which a part of an image is copied and pasted into another region of the same image. The copied region is then moved to a new location, making it appear as if there are multiple instances of the same object in the image. As a result, some details will be hidden as well as others being duplicated in the same image.

3.1.1 Splicing

In an Image splicing attack two or more images are combined to create a new composite image. In this attack, portions of multiple images are spliced together to form a single manipulated image. This technique is used to manipulate the content of an image, often with the intention of creating a false or misleading representation of a scene. In image splicing some image (or video) objects from one or more different images (or videos) into another image (or video) in order to generate a composite image (forged). This affects the original image pattern. It is one of the most aggressive and frequently used attacks. In modern days it is a very challenging task, especially when the attacker is skilled in image editing and takes care to hide any visual artifacts. The following Fig. 26 shows the example of image splicing.

Fig. 26 Osama Bin Ladin's Internet viral Spliced Image [62]

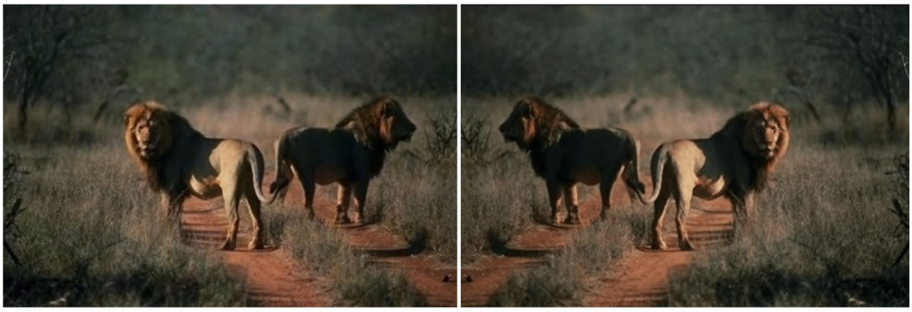

Fig. 27 a) Authentic Image b) Resampling Image through Horizontal flipping

3.1.2 Re-sampling

Resampling is the process of applying some geometric transformations (like rotation, scaling, or skewing operations) or any interpolation algorithms in order to create a malicious transformed image or a portion of the image and therefore a visually convincing forgery by, for example, increasing or decreasing the image size. In this attack, the image is resampled using interpolation techniques to change its size or resolution, which can introduce visual artifacts and potentially deceive viewers. The resampling process involves changing the number of pixels in the image, either by upsampling (increasing the resolution) or downsampling (decreasing the resolution). Interpolation methods are used to estimate the pixel values at the new locations after resampling. Common interpolation techniques include nearest neighbor, bilinear, bicubic, and Lanczos. The following Fig. 27 shows the example of image resampling.

3.1.3 Retouching

A retouching attack in image processing refers to the act of using image editing techniques to modify an image with the aim of improving its appearance or hiding specific details. It is usually applied as a post-processing operation of image tampering. Unlike some other types of image tampering attacks that involve inserting or removing objects, a retouching attack focuses on enhancing the image's visual quality, retouching imperfections, or altering certain elements to create a more appealing or misleading version of the original image. In this case, the original image (or video) will not be modified significantly, but only a few reductions in certain properties and characteristics of the image. The following figure 28 show the example of image retouching.

3.1.4 Inpainting

It is a type of image tampering attack where certain regions or objects in an image are removed or obscured by filling them in with visually plausible information. This is the process of drawing some missing content over the image or video using, for example, the "brush" software tool in order to repair damage. The following figure 29 shows an example of image inpainting.

The inpainting process is used to fill the gaps left by the removed objects, making it appear as though the tampered regions are a natural part of the image. inpainting attacks can

(a) (b)

Fig. 28 (a) Input Image (b) Retouched Image from MIT-Adobe FiveK dataset

be performed using various algorithms, such as texture synthesis, exemplar-based inpainting, and deep learning-based methods. The choice of inpainting algorithm affects the quality of the forgery and the difficulty of detecting it. The inpainting attack is particularly concerning because it can remove or hide critical information from an image, potentially leading to misinformation or misleading viewers. As a result, inpainting detection is an important area of research in digital image forensics.

3.1.5 Attacks on digital watermarking

Attacks on digital images are the processes that can degrade the strength of digital watermarks. Watermark attacks are divided as presented in Fig. 30.

Protocol attacks In protocol attacks, attacker adds their own watermark signal at the host side [18, 19, 27]. One example of a protocol attack is a copy attack, in which changing of the watermark takes place. In this process, the attacker intentionally removes or alters

(a) (b)

Fig. 29 (a) Corrupted Image (b) Inpainted Image

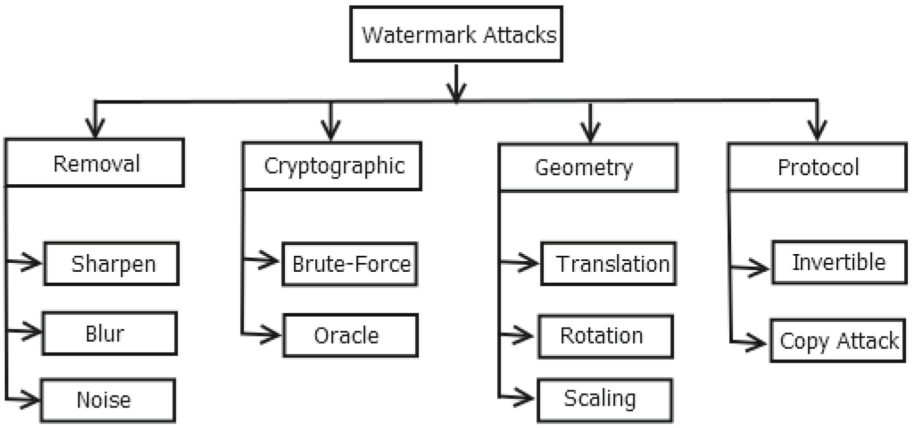

Fig. 30 Classification of Watermark Attacks [19]

the image watermark. Protocol attacks focus on attacking the whole concept related to the application of watermarking. Invertible watermarks are another example of protocol attack [81]. In this method, the attacker replaces his/her own watermark with the original watermark and tries to show the ownership of the data. In a copy attack, a watermark was imitated from the watermark data and copied to the target data by the attacker [82]. Copy attack does not involve the removal of watermark, but stake the application for which digital watermarks are used. Hackers are continuously developing more impact attacking techniques which may cause more damage in comparison to this attack. It can be further classified into two types:

- Invertible
- Copy attack

Cryptographic attacks The cryptographic attack deals with the security mechanism of the watermarking technique. This attack is used for breaching the security method utilized in the watermarking schemes [18, 19, 27]. In these attacks, the attackers try to find the loopholes in the main embedding algorithm and vanish the watermark information. Examples of this type of attack are Brute force attack and oracle attack. However, if the embedding algorithm is complex, these attacks can be easily restricted. Further, it can be classified into two types.

- Oracle
- Brute-force

Removal attacks Removal attacks are an attacking technique in which the entire watermark is removed from image data without affecting the security of watermark algorithms [18, 19, 27]. The removal attack completely removes the watermark from the object. Image quantization, denoising, collusion, and demodulating attacks are some examples of removal attacks. In these attacks, operations such as quantization or denoising should be optimized to the maximum distortion of the embedded watermark. At the same time, the quality of the distorted image kept to be high [27]. This can be classified into three parts.

- Sharpen
- Blur
- Noise

Geometric attacks Geometric attacks are considered to be the most challenging when we propose a robust image watermarking scheme. Some schemes [76–78] are robust against

geometric attacks with a combination of DFT and log-polar mapping (LPM). In a geometric attack, instead of removing the entire watermark, the attacker only distorts the embedded watermark of the data [18, 19, 27]. The author in the article [79] shows how the distortion of the watermark can be resisted with the help of a feature-based image watermarking scheme as well as common operation. Ohbuchi et al. in the paper [80] describe how some of the geometrical aspects affect geometrical transformation attacks. These types of attacks can target the pixels of the image for attacking, like pixels shifting, scaling of the image, and rotation of the image without any higher visual changes. The objective of these kinds of attacks is to degrade the quality of the watermark. It can be classified into three types.

- Rotation
- Scaling
- Translation

3.1.6 Other tampering attacks

There are also some other types of operations, such as filtering, Deepfakes, cropping, GAN-based tampering, or histogram adjustments, which are often applied without malevolent intention.

Figure 31 shows the digital image forensic techniques category.

4 Active forensic techniques

The active methods may be categorized into two classes: the digital watermarking and the digital signature. In the Active forensics technique, a new watermark or fingerprint is created for the image content of an image and these are added to the content of the digital image [26].

4.1 Digital image watermarking

Digital image watermarking is a mechanism to protect ownership of an image or video [16]. In this process, some information is added in the form of a logo, image, or text [17]. Watermark represents accurate and authentic information about image ownership. The process in which

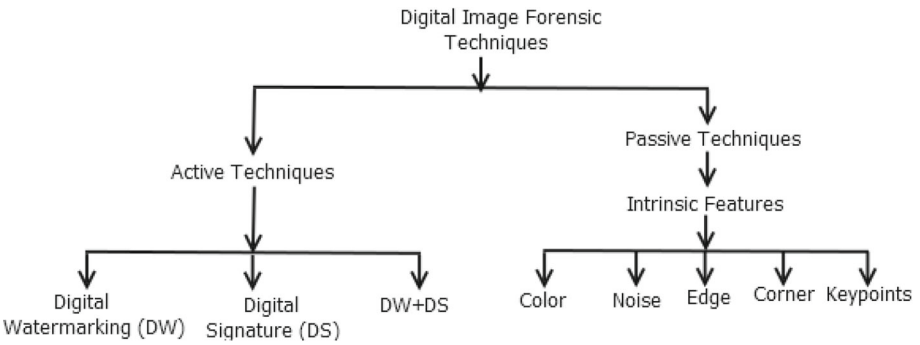

Fig. 31 Digital Image Forensic Techniques

(a) (b)

Fig. 32 (a) Original Image (b) Watermarked Image [79]

the watermark of an image is violated is called watermark attack [14]. There are many types of attacks like rotation, cropping, noise (salt & pepper, Gaussian), etc. which cause watermark image violation [18]. More detailed descriptions can be found in the articles [20–22, 27].

Watermarking is primarily used in the identification of copyright ownership for digital content (Image, Video, Text, etc.). In watermarking digital information in a carrier signal is concealed. It may or may not be possible that hidden information should contain a relation to the carrier signal. Watermark can be used in the verification of carrier signal authenticity as well as in the identification of its proprietor. It is primarily used in the identification of copyright violations. The two images (Baboon and Lena) with their watermarked images are shown in Fig. 32(a) and (b), respectively.

4.2 Watermarking detection technique

In the paper [235] new algorithm for digital watermarking and tampering detection technique, by combining these techniques, we can improve the security of image. We worked on RGB components such as red, green, and blue to enhance security and robustness. 2-DWT applied on RGB components for better results. In the tampering process, we used the watermarked image as a reference image for detecting tampering.

In the process of the watermarking detection algorithm, the RGB element of the input original image is taken and 2-level DWT is applied, which divides the image into high-frequency

components and low-frequency. The equivalent process is followed for the watermark which is to be fixed in the input original image. In this process, the separated components of the input original image and the watermark are multiplied using a scaling factor, and a new image is obtained.

This paper [237] describes a combination of Discrete Wavelet Transform (DWT), Discrete Cosine Transform (DCT), and Singular Value Decomposition (SVD) of digital image watermarking based on a zigzag order. From DWT we choose the high band to embed the watermark that facilitates more information, giving more invisibility and robustness against some attacks. Such as geometric attacks. The Zigzag method maps DCT coefficients into four quadrants representing low, mid, and high bands. Finally, SVD is applied to each quadrant (Fig. 33). Figure 33 illustrates the watermark extraction flowchart using DCT, DWT, and SVD.

Applying DWT for 2-D images in each dimension. The filters subdivide the input image into four non-overlapping multi-resolution sub-bands, a lower resolution approximation image vertical (LH1), (LL1), horizontal (HL1), and diagonal (HH1) detail components. The

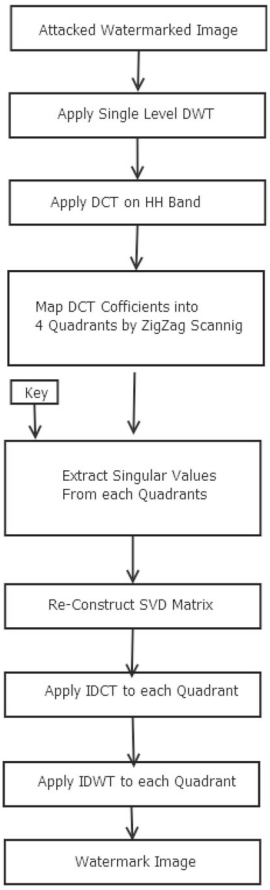

Fig. 33 Watermark Extraction Flowchart using DCT, DWT, and SVD

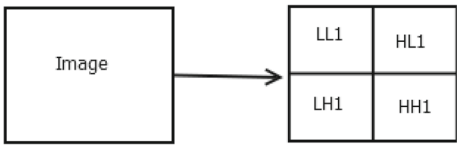

Fig. 34 Single level DWT

frequency districts of LH, HL, and HH respectively represent the level detail, the upright detail, and the diagonal detail of the original image. Figure 34 show the one dimension DWT.

DCT-based watermarking is based on two important points. The first one is that most of the signal energy lies at the low-frequencies sub-band which contains the most important visual parts of the image. The second one is by using compression and noise attacks high frequency components of the image are usually removed. In the below formula, DCT is given by (65), and inverse DCT is given by (66). Equation of 2-D DCT:

$$F(u,v) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} C(u)C(v)f(i,j)\cos\left[\frac{\pi(2i+1)u}{2N}\right] * \cos\left[\frac{\pi(2j+1)u}{2N}\right] \tag{65}$$

Equation of 2-D inverse DCT:

$$f(i,j) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} C(u)C(v)F(u,v)\cos\left[\frac{\pi(2i+1)u}{2N}\right] * \cos\left[\frac{\pi(2j+1)u}{2N}\right] \tag{66}$$

SVD is an effective numerical analysis tool used to analyze matrices. Without loss of generality, if A is a square image, denoted as $A \in R^{n\times n}$, where R represents the real number domain, then SVD of A is defined as $A = USV^T$ where U and V are orthogonal matrices, and S is a diagonal matrix, as

$$S = \begin{bmatrix} s_1 & & \\ & \ddots & \\ & & s_n \end{bmatrix} \tag{67}$$

Here diagonal elements i.e. s's are singular values and satisfy $s_1 \geq s_2 \geq \ldots . s_r \geq s_{r+1} \geq = \ldots .. = s_n = 0$

The following flowchart describes the watermark extraction process by using watermark extraction which shows how the embedded watermark is extracted from the attacked watermarked image.

4.3 Applications of digital watermarking

In this modern digital age, people use internet platforms for different purposes like entertainment, education, and many others. The authenticity of data is a very crucial topic in this modern day. Different technology has been developed to authenticate the data, image, and videos, watermarking is one of the best methods to do this. Application of watermarking is in many fields like, copyright protection [136–139], content identification & management [136, 138, 139], digital forensic [137–139, 144], broadcast monitoring, military, [136–139], archiving of media file [137, 138], medical applications, remote education, covert communication, [137, 139–143], fingerprinting [136–139], real-time audio/video and robotics, Secure E-Voting and different licenses, information security solutions in smart cities [23]. Different

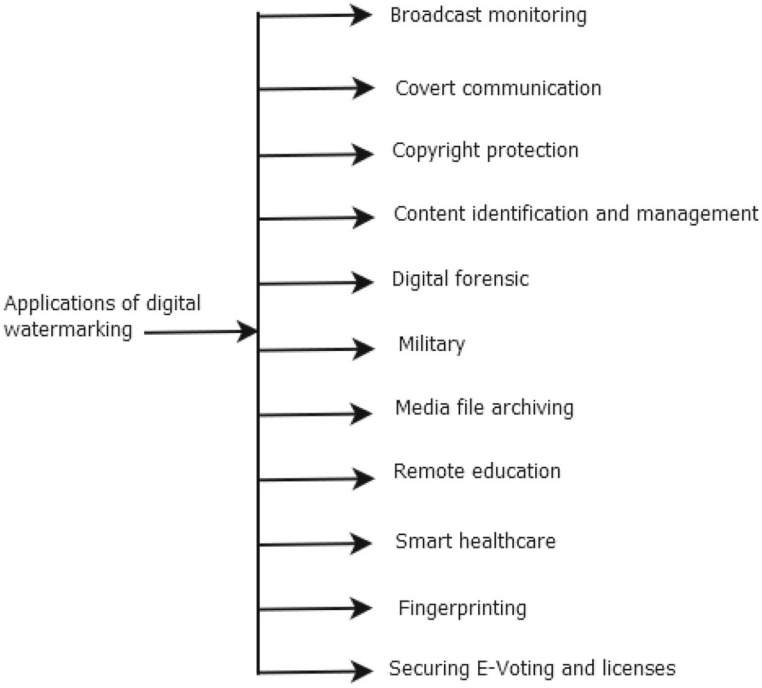

Fig. 35 Summary of Some Reported Secure Watermarking Techniques

types of watermarking techniques have been explained in the article [24]. Different varieties of watermarks with respect to their document types are shown in Fig. 35.

4.4 Digital signature

Verification of Digital signatures and detection of forgery are the methods to verify the authenticity of a signature. Digital signatures are currently used by many software for authentication purposes. Digital signatures are commonly used for contract management software and financial transactions, and in other cases, they can be utilized as a forgery detector. Digital Signatures are used for authentication purposes of an image. There are two steps of image authentication. One is the signature generation of the image, and in the second step verification of the generated signature is done by authenticating it with the original image [55].

Further, Digital signature generation is processed with two steps, In the first step, extraction of the coordinates of the image structures which are visually important, is done. These coordinates are used to find the feature points over which the authentication scheme is dependent. In the second step, a public key encryption scheme like RSA is used to encrypt the feature point set [56]. In the article [57], a novel digital signature-based authentication of image technique has been proposed by utilizing a private encryption key technique. The performance of this method is better in comparison to the previous one.

5 Passive forensic techniques

Passive digital image forensics is the process of image forgery detection based on technique, which finds some basic patterns in the master image that are left in the process of image acquisition, storage, or editing. By this method, the authentication of forged images is done. In the last decades many research article has been published based on passive forensic techniques, review paper [227] did a systematic survey to extract the global forensic techniques for image tamper detection (Fig. 36). Figure 36 depict the various image tampering attacks.

5.1 Copy move attack

Copy move attack is a method, in which attackers copy one region of an image and paste it at different locations to conceal some primary information. The objective of such type of attack is to hide some region of an object by copying some regions [28]. Copy-Move Forgery Detection (CMFD) is dependent on key points. It is a mechanism in which feature points of the image are extracted and local features of the image are employed to find the regions which are duplicates. This detection technique performs best with respect to computational cost, robustness, and memory requirement. The examples of copy-move forgery detection are shown in Fig. 25(a) and (b).

5.1.1 Copy-move forgery detection

Division of image forgery detection is based on whether the original image is present or not. There are mainly three categories of copy-move forgeries, complex, plain, and affine [29]. Most of the previous Copy Move Forgery Detection (CMFD) techniques were related to plane cloning. Sometimes human decision is more accurate in comparison to machine learning or any other algorithm, if image forgery is based on a computer [30]. For Copy Move Forgeries (CMFs) with the help of Generative adversarial networks, a new technique has been developed. This algorithm uses a deep neural architecture approach. In this algorithm, for training purposes, no forged images are accessible [31].

In the paper [32], a new technique based on passive copy-move forgery detection is demonstrated. It exploits the usability and effectiveness of the Halftoning-Based Block Truncation Coding (HBTC) image feature. In this algorithm, for image feature descriptor extraction, the forged image is partitioned into several parts which are overlapping. Further, for processing of HBTC, the image is divided into the non-overlapping block. The HBTC performs the image compression of each block f(x,y) by using (68).

$$\mathcal{H}\{f(x, y)\} \Rightarrow \{q_{\min}, q_{\max}, b(x, y)\} \tag{68}$$

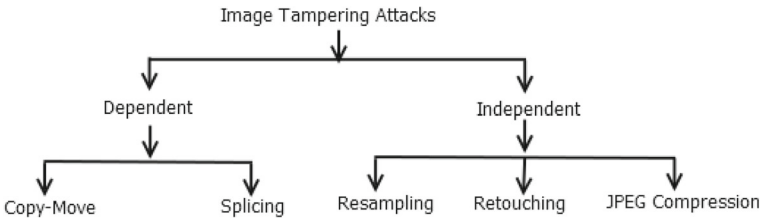

Fig. 36 Image Tampering Attacks

for every pixel location $x = 1, 2, \ldots, m$ and $y = 1, 2, \ldots, n$. The $q_m in$ and $q_m ax$ denote the color quantizers, i.e. min and max quantizer, respectively. The $q_m in$ and $q_m ax$ are represented by (69) and (70)

$$q_{\min}\& = \min_{\forall x,y} f(x, y) \tag{69}$$

$$q_{\max}\& = \max_{\forall x,y} f(x, y) \tag{70}$$

In this process Firstly partition of the suspicious image into several overlapping image blocks of fixed size has been done. For feature extraction image is sent for HBTC processing, Performs similarity matching for all image blocks, and Performs post-processing to remove the isolated regions in the last step.

Comparision and future scope After comparing with the previous method, This proposed method proves that the HBTC feature is useful for forgery detection, not only for image compression and retrieval. For future work, we can consider multi-resolution conditions, rotation invariants as well as translation problems.

Copy-move forgery detection technique which is based on Keypoint, has been demonstrated in the article [33]. In the keypoint-based approach, Speeded-Up Robust Features (SURF) descriptor algorithms and scale and rotation-invariant interest point feature detectors are used. Another approach for copy move forgery detection is Block-based techniques. Example of this technique is Discrete Wavelet Transform (DWT), Principle Component Analysis (PCA), and Discrete Cosine Transform (DCT). In this survey paper, keypoint-based copy-move forgery detection schemes involve, detecting and describing the local features of the images by using the algorithms like SURF and SIFT.

Comparision and future scope The Main disadvantage of the block-based technique is more time consumption. The block size, image size, and offset chosen for dividing an image affect the forgery detection significantly. With the help of this survey, it concludes that to exact forgery region block-based approach is good, keypoint based approach excels in all other aspects as discussed above. Hence, For large-size images, the keypoint-based approach becomes the ideal choice over the block-based approach.

- In the paper [34], for image local statistical features calculation scale-invariant feature transform (SIFT) descriptor is used. The Best-Bin-First nearest-neighbor is utilized to match the keypoints in the given algorithm. The performance of scale invariant and rotation over SIFT is not good. In this paper SIFT algorithm extracts features of local image patches mainly in four steps: In the first step of the computation searches for extrema over all scales and image locations. Given an input image I(x,y) then the scale space of image I is defined as follows:

  $$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$$

  where * is the convolution operation in x and y directions, and the Gaussian function

  $$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{\sigma^2}}$$

  where σ is the factor of scale space
  In order to detect the SIFT framework, the method used the scale-space extrema in the difference-of-Gaussian (DoG) function convolved with the image, $D(x, y, \sigma)$, and with

the help of a constant multiplicative factor k :

$$D(x, y, \sigma) = [G(x, y, k\sigma) - G(x, y, \sigma)] * I(x, y)$$
$$= L(x, y, k\sigma) - L(x, y, \sigma)$$

In the second step, keypoints are filtered so that only stable keypoints are retained. After getting a keypoint candidate compares a pixel to its neighbors, next is to perform a detailed fit to the nearby data for location, accuracy, the ratio of principal curvatures, and scale. The next step is the Orientation assignment. This is the key step to achieve invariance to image rotation, in this step based on local image gradient directions keypoint is assigned one or more orientations. In the last step, we want to compute descriptor vectors for each keypoint such that descriptors are distinctive and robust to other variations, such as illuminations etc. Compute the feature descriptor as a set of orientation histograms on 4 x 4-pixel neighborhoods (Table 3). Table 3 summaries the different copy move forgery detection techniques based on various method and block size.

- This process contains three steps. In step one, for locating & detecting the forged region, an efficient Modified Sub windows Search algorithm is used [35]. In step two, for segmentation, planar homography constraint and duplicate regions are used. In the last step, with the help of contours, the differentiation of authentic region and forged region has been done.
- In the article [36], three step method has been proposed. In the very first step, Image feature extraction with the SHIFT descriptor and feature matching process is done. For similar keypoint matching, a 2NN test which is generalized is used repeatedly. Further, in step two, for feature clustering agglomerative, hierarchical clustering is used. In the last step, estimation of Geometric Transformation has been done. In this method, multiple clone region has been detected. The disadvantage of this algorithm is that it is not able to detect tampering precisely.

SIFT features extraction and multiple keypoint matching In this step, we input a test image with key points x=x1, x2,...Xn with the corresponding descriptor f1, f2,.....fn is extracted. We calculate distance between two descriptors with respect to a global threshold does not perform well to match two keypoints. The key point is matched only if

$$d_1/d_2 < T \text{ where } T \in (0, 1). \tag{71}$$

Clustering and forgery detection In the further step of this algorithm we assign each keypoint to a cluster; after that compute all the reciprocal spatial distances among clusters, We merge the closest point to form a single cluster.

Geometric transformation estimation For unauthentic images, by using this method we can determine which type of geometrical transformation was used between the authentic area and its copy-move version. Let the matched point coordinates be, for the two image portion, $\tilde{\mathbf{x}}_i = (x, y, 1)^{\mathrm{T}}$ and $\tilde{\mathbf{x}}_i' = \left(x', y', 1\right)^{\mathrm{T}}$ respectively, their geometric relationships can be defined by an affine homography which is represented by a 3×3 matrix H as:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \mathrm{H} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{72}$$

The above matrix is computed by resorting to at least three matched points. In this way, we determine H using Maximum Likelihood estimation of the homography. This method seeks homography H and pairs of perfectly matched points $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}_i'$ this minimizes the

Table 3 Copy Move Forgery Detection Technique on the Basis of Different Techniques and Block Size

Author & Year	Technique Used	Input	Performance	Strength	Weakness
Bayram, Sevinc, Husrev Taha Sencar, and Nasir Memon, 2009	Fourier–Mellin Transform (FMT) [123]	Block size = 32 × 32	– Detection of the duplicate region is very good whether the image is manipulated in several ways. – DCT coefficients can successfully detect up to 5° and rotations up to 10°.	– The proposed method is more robust and accurate, and also introduces a new detection technique counting bloom filter.	– If the input image is highly compressed then counting bloom filter performance decreases.
Shivakumar, B. L., and S. Santhosh Baboo., 2010	Speeded-Up Robust Features (SURF) [129]	Color Images Database, and edited using the GIMP software	– Precision=72.41% Accuracy=43.75%	– This proposed copy-move forgery detection method is robust, and detects duplication regions with different sizes.	– Sometimes it is not capable to detect small copied regions successfully.
Goncalves, Hernani and Corte-Real, Luís and Goncalves, Jose A., 2011	Scale Invariant Feature Transform (SIFT) [34]	Remote sensing images (The Landsat and Hyperion images)	– Precision=88.37% Recall=79.17%	– Stability of SIFT feature descriptors is very strong, due to this the performance of the proposed method is very good for different post-image processing.	– For low SNR and small size tampered region robustness is not good.
Li, Leida, et al., 2013	Local binary pattern (LBP) [124]	Block size = 80 × 80	– The Robustness of this method is high for different types of distortion like flipping, rotation, blurring, and JPEG compression.	– The proposed method is robust to the traditional signal processing attacks as well as rotation and flipping. – We get the more promising result by using the low pass filter before image segmentation.	– This method has a problem detecting forgery when the image region is rotated by general angles.

Table 3 continued

Author & Year	Technique Used	Input	Performance	Strength	Weakness
Ardizzone, Edoardo, Alessandro Bruno, and Giuseppe Mazzola., 2015	ZERNIKE method [130]	Medium-sizedd images with 1000x700 or 700x1000	– Precision=94.0% Recall=75.0%	– The Advantage of this method is that it can be used for copy-move recognition and detection, and it is also able to locate the copy-move region.	– In the case of transformation (anisotropic deformations), the performance of the proposed method is not as good as we expected. – There is no proper filling of holes between triangles.
Yu, Liyang and Han, Qi and Niu, Xiamu., 2016	Harris detector and MROGH [173]	MICC-F2000 dataset	– Precision=87.80% Recall=63.10%	– Not only copy-move forgery detection but this method can also be implemented for computer vision, such as video event detection and multimedia indexing problems.	– This method gives moderate results on additive noise, degree of scaling, and combined effects. – Performance declines rapidly when the above attacks are strong.
Pandey, Ramesh Chand, et al., 2016	SURF + Histogram of Oriented Gradients [125]	Blocks of 64x128	– Precision=96.3% Accuracy=95.5%	– Used of hybride feature provides copy-move area completely instead keypoint as shown in the SIFT and SURF CMFD.	– By this proposed method salient key points are not recovered by the SIFT-like technique.
Zheng, Jiangbin, et al., 2016	Dyadic Wavelet Transform (DyWT)[127]	Block size = 16 × 16	– For good efficiency of detection block-based method is preferred and for reliable detection method key point-based method is used. – For database one Precision = 0.7759, Recall = 0.9375 and F1 = 0.8491. – For database two Precision = 0.8851, Recall = 0.8648 and F1 = 0.8717.	– This method effectively detects image forgery from both non-smooth regions and smooth ones whilst reducing the computation cost.	– This method also is not able to handle the same problem: For copied smooth regions, keypoint-based methods fail to work.

Table 3 continued

Author & Year	Technique Used	Input	Performance	Strength	Weakness
Zhang, Ying, et al., 2016	Deep learning approach[209]	Block size = 32 × 32	– Detection accuracy of the proposed method with jpeg image is 87.51% – Overall accuracy to detect tampered regions is 91.01%	– we can use this method for multi-format image forgery detection.	– In this paper there are mainly two drawbacks: we identify the tampered region manually, due to this author does not provide a threshold for detecting the regions. – Secondly, By using this test database the tampered regions are not detected accurately.
Soni, Badal, Pradip K. Das, and Dalton Meitei Thounaojam., 2017	Fast Walsh-Hadamard Transform (FWHT) [128]	Block size = 8 × 8	– Performance of this method is: Detection accuracy (TPR = 99.2% and FPR = 7.8%) and time processing time is 29.3 sec.	– The proposed system performs very well on the different types of attacks like color reduction, blur movement, brightness changes, and contrast adjustment.	– For compression attacks and highly similar regions accuracy of forgery detection is not good.
Warif, Nor Bakiah Abd, et al., 2017	SIFT and symmetry-based matching [174]	NB-CASIA	– F-score = 83.4%	– By using the SIFT-Symmetry method we got the symmetry matching technique, by doing so we achieved high robustness against CMF with a combination of reflection and other transformations.	– This method has mismatched a region in an original image.
Jin, Guonian, and Xiaoxia Wan., 2017	Optimized J-Linkage and Non-maximum value suppression [172]	Dataset MICC-F600	– Precision=90.20% Recall=93.70%	– In this method we enhance the discriminative power of SIFT keypoints. – With the use of the J-Linkage algorithm we can reduce the clustering time in the case of a mass of matched pairs.	– In the proposed method, average precision is very low for multiple copies of tampered images.

Table 3 continued

Author & Year	Technique Used	Input	Performance	Strength	Weakness
Bunk, Jason, et al., 2017	Convolutional Neural Networks (CNNs) and LSTM networks [207]	NIST Nimble 2016 dataset	– AUC of the proposed method is 0.9138.	– In this article both LSTM and Convolutional Neural Networks (CNNs) based networks are effective in extracting resampling features to detect tampered regions.	– Some state-of-the-art is still performing better than the proposed method.
Yang, Bin, et al., 2018	CMFD-SIFT [126]	Image with various resolutions varies from 400 × 300 to 4000 × 3000 pixels.	– TPR=95.88% FPR=9.02%	– For highly uniform texture this method detects copy-move forgery based on the CMFD-SIFT.	– Still needs an improvement for transformation like non-affine transformations.
Wang, Huan, and Hongxia Wang., 2018	Discrete cosine transform (DCT) [111]	First is MICC-F2000 and second is MICC-F220 datasets	– Output is irregularly tampered and multiple duplicated regions in forged images. – Precision = 0.902 Recall = 0.910.	– The proposed algorithm perceptual hashing algorithm is robust for some special attacks, such as luminance, adjusting contrast ratio, and hue. – The Application of this algorithm is to prevent some conventional attacks, such as adding Gaussian blurring and white Gaussian noises.	– Attacks like scaling and rotation robustness and complexity of the algorithm are not good
Li, Yuanman, and Jiantao Zhou., 2018	Hierarchical feature point matching method [218]	Six datasets (FAU, GRIP, MICC-F220, MICC-F600, CMH and COVERAGE)	– The proposed keypoint-based algorithm has accurate localization and enhances the crucial matching points. – AVERAGE NUMBER OF KEY POINTS, MATCHES = 41150 CPU time = 48.6 sec	– Compared to another algorithm which is not robust our proposed algorithm achieves very good accuracy and time complexity.	– Hierarchical feature point matching could lead to slower processing times, especially for large images or real-time applications.

Table 3 continued

Author & Year	Technique Used	Input	Performance	Strength	Weakness
Wang, Chengyou, et al., 2019	SURF, PCET and mathematical morphology with iterative strategy [112]	Image Lenna (size: 512*512)	– This method is robust (F scores .96, .96, .91 for blurring, JPEG compression, and noise addition tests). – This method has low computational complexity and high accuracy.	– In this proposed method we combine the advantages of keypoint-based and block-based image CMFD methods.	– It is difficult to generalize the parameters in various conditions like morphology operations and hierarchical clusters.
Kuznetsov, A, 2019	VGG-16 convolutional neural network. [217]	Image size 224*224*3	– Output of this method (Precision = 95.0%, Recall = 98.1% and Accuracy = 96.4%)	– This method works under conditions of repeated compression of distorted images by the JPEG algorithm in a narrow range.	– Sometimes training this type of network is computationally very complex and needs a large amount of data.
Agarwal, Ritu and Verma, Om Prakash, 2020	Deep Learning [205]	Input Image size:(224*224*3)	– Precision = 98.026 and Recall = 89.583 with accuracy = 95– In this proposed technique of accuracy of the algorithm is enhanced due to the visibility of the forged region at the end of the process. – This method has low computational complexity and high accuracy.	– Weakness of this method is, that the patch-matching procedure gets confused while matching multiple tampered patches in an image.	
Zhu, Ye, et al., 2020	AR-Net [166]	CASIAII, COVERAGE and CoMoFoD dataset.	– Comparison with state of arts on different data sets. – For CASIAII (Precision = 58.32, Recall = 37.33, F1 = 45.52), COVERAGE (AUC = 0.8488) and COMOFOD (Precision = 54.21, Recall = 46.55, F1 = 50.09).	– The AR-Net can detect tampered regions at a pixel level and has high robustness in post-processing operations, such as recompression, noise, JPEG, and blur.	– The Disadvantage of AR-Net is that it is single-stream without utilizing the information of multiple modes.

Table 3 continued

Author & Year	Technique Used	Input	Performance	Strength	Weakness
Chen, Xu, et al., 2020	Deep Neural Network [206]	Input Image Size 500 x 500.	– In this experiment C19-Res Net and C19-ST-Res Net are 5.7% and 7.4%, which is higher than accuracy of C19-SVM	– This Method based on a convolution neural network has better performance than that based on artificial features of SVM.	– We will improve the demerit of this method in the future by analyzing how things work in the black box and enhancing the robustness of the method.
Wang, Yilan, Xiaobing Kang, and Yajun Chen., 2020	PCET-SVD and histogram [167]	CoMoFoD and CASIA image database with image sizes 512*512 and 3000*2000.	– Following are Precision rates for different attacks: – Rotation (20) = 95.92, Scaling (95 %) = 97.92, Gaussian noise (0.02) = 97.36 and Gaussian blurring (w = 5,	– For different types of geometric rotation and scaling attacks, this copy move method performs very well with the low computational complexity and adaptability of the similarity threshold.	– The proposed method did not perform well for smooth regions under combinations of multiple geometric attacks.
Islam, Ashraful, et al., 2020	Attentive Generative Adversarial Network [216]	USC-ISI dataset and 80,000 pristine images (size 320Ã-320)	– For USC-ISI dataset Precision = 96.83, Recall = 96.14 and F1 = 96.48 – The Accuracy of the proposed method is better compared to state of art method.	– compared to the previous state-of-the-art DOA-GAN has provisionally appeared to produce more accurate copy-move masks and better distinguish copy-move target regions from source regions.	– The proposed method is not capable of solving problems like vision tasks co-saliency detection, localization, and identifying image-level forgery in satellite images.

total error function as in Equation 73:

$$\sum_i \left[d\left(\mathbf{x}_i, \hat{\mathbf{x}}_i\right)^2 + d\left(\mathbf{x}'_i, \hat{\mathbf{x}}'_i\right)^2 \right] \text{ subject to } \hat{\mathbf{x}}'_i = \mathrm{H}\hat{\mathbf{x}}_i \forall i \tag{73}$$

- In the article [38], authors use DCT and SIFT methods to identify the various post-processing techniques. For Gaussian noise, DCT and JPEG compression are robust. Hence, This method detects forgery even if the image has gone through different post-processing operations because of the robust nature of DCT towards these operations.

Yuan Rao proposed a method using CNN for copy move and splicing detection. In this paper, a convolutional neural network (CNN) is applied to automatically learn hierarchical representations from the input RGB color images. In this method, two steps are involved, i.e., feature learning and feature extraction. In the first step, we pre-train a CNN model based on the labeled patch samples from the training images. CNN architecture in this process contains several cascaded convolutional layers. Suppose $F^n(X)$ the feature map in layer n of the convolution with the kernel (filter) and bias defined by W^n and B^n, respectively, we have:

$$F^n(X) = \text{pooling}\left(f^n\left(F^{n-1}(X) * W^n + B^n\right)\right) \tag{74}$$

where $F^0(X) = X$ is the input data, $f^n(\cdot)$ is a non-linear activation function that applies to each element of its input, and pooling $(\cdot)$ represents the pooling operation that performs downsampling along spatial dimension using MAX or MEAN operation (max or mean-pooling). In this proposed method CNN consists of 8 convolutional layers, 2 pooling layers, and a fully-connected layer with a 2-way softmax classifier. Further in the model initialization weights of the first convolutional layer of our CNN model with 30 basic high-pass filters are used in the calculation of the residual maps in SRM. Let W^c_{SRM} and W^c_{CNN} be the filter kernels of size $m \times n$ in SRM and the weight matrix of size 5×5 used in the first convolutional layer, respectively, we have:

$$W^c_{CNN}(x, y) = \begin{cases} W^c_{SFM}(x, y), & \text{if } x \le m, y \le n \\ 0, & \text{otherwise} \end{cases} \tag{75}$$

In feature fusion, the pre-trained CNN learns a feature mapping f that transforms an input patch $X \in \mathbb{R}^{p \times p}$ (patch of size $p \times p$) to a much-condensed representation $Y = f(X) \in \mathbb{R}^K$ (features of K-Dimension) and serves as the local descriptor for the input patch. Y_i for each sampled patch X_i, and concatenate all the Y_i together to obtain the new image representation:

$$\boldsymbol{Y} = \left[Y_1 \cdots Y_T \right] \tag{76}$$

where $T = (\lceil (m - w)/s \rceil + 1) \times (\lceil (n - w)/s \rceil + 1)$, and $\lceil x \rceil$ is the ceiling function. For the obtained image representation $\boldsymbol{Y}$ with T local descriptors Y_i, pooling operation (max or mean pooling) is applied on each dimension of Y_i over T sampled patches, i.e.,

$$\hat{Y}[k] = \text{ Mean or Max } \{Y_1[k] \cdots Y_T[k]\} \tag{77}$$

Comparison and future scope We also compare the performance of the proposed method with some existing methods with respect to different datasets. Our proposed model detection accuracy is better than paper He in [169], Muhammad in [177], and Zhao in [178] on 3 public datasets. In the future, we can perform forgery detection for different dataset

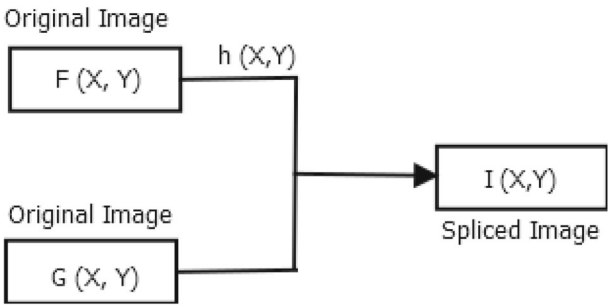

Fig. 37 Steps of Image Splicing

5.2 Splicing attack

In today's internet-dominating age manipulation of an image becomes a very easy task. There are many techniques of image forgery. One of the common techniques is image splicing. In image splicing, one segment of the image is cropped and pasted on the different or same image. So, it becomes very tedious to decide whether an image is authentic or not. In this modern age, due to easy access to editing software, splicing images is not a big deal. Splicing of images can violate the property of integrity and authenticity of images. It is a mechanism of creating a new image by cutting some part of an image and pasting it into a different image. Splicing detection of an object is difficult compared to Copy Move Forgery Detection. The Copy Move Forgery Detection technique is easy because it contains a similar contour with the same texture property. In splicing detection, different texture objects increase the feature complexity of the image. Hence, making it difficult to detect. Many researchers have found that, due to the sharp transition of edges and corners, it is difficult to detect the splicing of an image. Figure 37 is showing the important steps involved in image splicing.

5.3 Image splicing detection

As we know, Image splicing detection is more difficult compared to Copy Move forgery detection. Convolutional Neural Network (CNN) is one of the traditional mechanisms for detecting the splicing of images. In the method, attribute of different regions of an image is retrieved and compared, to differentiate regions of the image. Another modern method is the Ringed Residual U-Net (RRU-Net) for the detection of image splicing forgery. In this method, one well-known Neural network namely CNN is used. RRU-Net is also based on CNN, The Concept of CNN mimics the logic of the human brain working [39]. A ringed residual U-Net(RRU-Net) is proposed to overcome the drawbacks of traditional feature extraction-based methods,

RRU-Net robustness is dependent on the learning of the CNN network. This can be enhanced by strengthening the learning process. The u-Net method is developed by Olaf Ronneberger et al. [40] in 2015. In U-Net, the extraction of information is done with the help of successive layers. To minimize the information loss and locate exact point feature sampling, a combination of features with high resolution is propagated through an expanding path that is symmetric. There are some methods based on U-Net image segmentation [41, 42].In the paper [41] author shows that with the help of pre-trained weights performance of U-Net can be easily improved. Generally, a U-Net architecture consists of a contracting

path to capture context and a symmetrically expanding path that enables precise localization. U-Net is capable of learning from a relatively small training set. U-Net is capable of learning from a relatively small training set. Following are the expressions for discrete objects, where y_i is a label of the corresponding pixel i and $\hat{y}_i$ is predicted probability for the pixel.

So, we mention the image segmentation task as a pixel classification problem, The loss function for binary classification tasks - binary cross entropy that is defined as:

$$H = -\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log \hat{y}_i + (1 - y_i)\log\left(1 - \hat{y}_i\right)\right) \tag{78}$$

After combining these expressions, we get the generalized loss function, as follows,

$$L = H - \log J \tag{79}$$

we maximize probabilities for the right pixels to be predicted and maximize the intersection, to minimize this loss function.

Figure 26 describes an example, presenting the splicing of an image. In this example, a forged image was formed by combining the faces of two different persons.

A new method for image splicing detection has been developed in the paper [43]. It is based on 2D array statistical features with wavelet and Markov features. In this method, image segmentation has been done into multiple blocks of DCT to obtain appropriate accuracy. The advantage of this natural image model having the following two combinations: the first one is the combination of features derived from the image pixel 2-D array and those derived from the MBDCT coefficient 2-D arrays and the second one is, the combination of moments of characteristic functions based features and Markov process based features. The proposed model has the following steps:

- This depends on image pixel 2-D array as well as multi-size block discrete cosine transform (MBDCT) coefficient 2-D arrays. Denoting an $n \times n$ image block by $f(x, y)$, $x, y = 0, 1, \cdots, n-1$, the 2-D block DCT coefficients are given by

$$F(s,t) = \frac{2}{n}\sum_{x=0}^{n-1}\sum_{y=0}^{n-1}\Lambda(x)\Lambda(y)\cos\frac{\pi s(2x+1)}{2n}\cos\frac{\pi t(2y+1)}{2n}f(x,y), \tag{80}$$

  where

$$\Lambda(x) = \begin{cases} \frac{1}{\sqrt{2}}, x = 0 \\ 1, \text{ otherwise} \end{cases},$$

  and $s, t \in \{0, 1, \cdots, n-1\}$.
- For every 2-D array, its prediction-error 2-D array, and statistical moments of the corresponding 1-D and 2-D characteristic functions are extracted as moment features and all of their wavelet subbands. Prediction-error 2-D array can be expressed by:

$$\Delta x = x - \bar{x} = x - \text{sign}(x) \cdot \{|a| + |b| - |c|\}. \tag{81}$$

- For each of these 2-D arrays, we round each element to integer, take absolute value (for an image pixel 2-D array, these operations result in the image pixel 2-D array itself), and then form different 2-D arrays. F
- These two feature parts form the feature vector for a given image.
- Support vector machine (SVM) classifier is used with radial basis function (RBF) kernel.
- The classifier is trained with a dataset of sufficiently large number of images before being used as a classifier for splicing detection.

Comparison and future scope This experiment has demonstrated that by using the proposed splicing detection scheme outperforms the state-of-the-art by a significant margin when we implement with the above-mentioned dataset, indicating that the proposed approach possesses very good capability in splicing detection.

In this paper [44], a new feature dependent on Hilbert-Huang Transform (HHT) and moments of characteristics functions along with wavelet decomposition is proposed. This algorithm considers high non-linearity and non-stationarity of images.

A novel technique dependent on Principal Component Analysis (PCA) [45] is developed for the detection of the image splicing technique. This technique, referred to as cPCA++, utilizes the fact that the interesting features of a "target" dataset may be obscured by high variance components during traditional PCA. The advantage of this proposed cPCA++ method that, traditional PCA was not able to identify important dataset-specific patterns but cPCA++ wide variety of settings. so, the proposed cPCA++ method is more efficient than cPCA, because it does not require the parameter sweep in the latter approach. PCA" (cPCA). We derive the cPCA++ method from the traditional PCA method. For a data matrix $\boldsymbol{Z} \in \mathbb{R}^{M \times N}$, where M denotes the original feature dimension of the data and N denotes the number of instances included in $\boldsymbol{Z}$. PCA first computes the empirical covariance matrix of $\boldsymbol{Z}$:

$$\widehat{\Sigma} \triangleq \frac{1}{N} Z Z^{\top} \tag{82}$$

where we assumed that $\boldsymbol{Z}$ has zero-mean. Next, PCA would compute the subspace spanned by the K top or leading eigenvectors of $\widehat{\boldsymbol{\Sigma}}$ (i.e., those corresponding to the largest K eigenvalues). The basis for this space would constitute the filters used to process the input data:

$$\boldsymbol{F}_{\mathrm{PCA}} \triangleq \mathrm{evecs}_K(\widehat{\boldsymbol{\Sigma}}) \tag{83}$$

where K denotes the number of principal eigenvectors to return. Now, a low-dimensional version of the input data Z can be obtained as:

$$\boldsymbol{Y}_{\mathrm{PCA}} = \boldsymbol{F}_{\mathrm{PCA}}^{\top} \boldsymbol{Z} \tag{84}$$

So, we proposed cPCA++, which is a novel method for discovering discriminative features in high-dimensional data.

Comparison and future scope The recently proposed algorithm (contrastive PCA (cPCA)) compared with the cPCA++ approach which achieves similar discriminative performance in a wide variety of settings because this eliminates the need for the hyperparameter sweep required by cPCA. If "cPCA++" offers better performance in capturing essential features with fewer dimensions, it could find applications in various fields dealing with high-dimensional data, such as natural language processing, computer vision, and bioinformatics.

After the failure of traditional PCA to detect the location of spliced image, a new dimensionality reduction technique is introduced called "contrastive PCA" (cPCA) [46]. Based on previous research on image-splicing detection techniques, two categories of image-splicing techniques can be found. The first technique is splicing region localization and another technique is image splicing detection. In the first category, they only detect whether an image has gone through splicing detection or not. In the second technique identification of spliced image region has been done.

In the article [47], Johnson and Farid proposed a camera response function (CRF) based algorithm for the detection of spliced image region by detecting the inconsistent light region of an image. To create image forgery we combine two different images. In this process, matching the lightning effect from different direction play a vital role. With the help of the

following assumptions we can follow the standard approaches for estimating light source direction:

- The surface of interest is Lambertian (the surface reflects light isotropically);
- The surface has a constant reflectance value and is illuminated by a point light source infinitely far away
- The angle between the surface normal and the light direction 1 is in the range 0° to 90°.

The image intensity can be expressed as:

$$I(x, y) = R(\overrightarrow{N}(x.y).\overrightarrow{L}) + A \tag{85}$$

where $\overrightarrow{L}$ is a 3-vector pointing in the direction of the light source, R is the constant reflectance value, $\overrightarrow{N}$ (x, y) is a 3-vector representing the surface normal at the point (x, y), and A is a constant ambient light. For multiple light sources: Light has magnificent linear properties. As such, a scene illuminated with, two infinite light sources takes the form:

$$\begin{aligned} I(x, y) &= R((\overrightarrow{N}(x, y)\overrightarrow{L1}) + (\overrightarrow{N}(x, y)\overrightarrow{L2})) + A \\ &= R(\overrightarrow{N}(x, y)(\overrightarrow{L1} + \overrightarrow{L2})) + A \\ &= R(\overrightarrow{N}(x, y)\overrightarrow{L} +) + A, \end{aligned} \tag{86}$$

In this method, the author described a technique for calculating the direction (within one degree of freedom) of an illuminating light source. The author currently investigating how surfaces of known geometry in the image (cylinder, sphere, plane, etc.) can be used to estimate the third component of the light source direction, N_z. We expect that the technique described here, in coincidence with other forensic tools [119, 120, 154, 180] will make it increasingly very tough to create convincing digital forgeries.

Here are a few papers that describe the CRF-based methods for the detection of spliced image regions [48, 51].

Some of the splicing localization methods have been discussed in the articles [48, 49, 66, 83–88] (Table 4). Table 4 summaries some recent image splicing detection techniques with different classifiers.

In this article [83], authors Liu and Pun explained the Fully Convolutional Network (FCN) and Conditional Random Field (CRF) based technique to identify the forged region of synthesized images from various resources. In this proposed algorithm mainly two-step process involves:

In the first step, three different fully convolutional networks were initialized with the help of the VGG-16 network, after that training of a handmade forged dataset was done with these networks. The trained FCNs-CRF classifier then gives pixel-to-pixel prediction given a test image. In Multi-scale Fully Convolutional Networks (MFCN), Convolutional neural networks (CNN) work as classifiers. For example, a trained convolutional neural network f classification output is $y = f(I)$. For forgery detection, y is a binary output: $y\epsilon(0, 1)$, denoting whether the image I is forged or not. Further, a conditional random field 120 is an undirected graph model conditioned on observations. Let $G = (V, E)$ be an undirected graph where V represents a set of nodes for each pixel in the given image, and E represents edges connecting neighboring pixels.

Comparison and future scope After comparing output with [148, 225] we conclude that There are two factors affecting their performance: First is the variance of added Gaussian noise to the spliced objects and the second one is the number of spliced objects in the pictures. The

Table 4 Some Recent Image Splicing Detection Methods

Technique Proposed by	Classifier Used	Extracted Feature	Accuracy	Strength	Weakness
Suthiwan [69]	Support Vector Machine	Edge feature and rake transform-based features	99.0%	In this article a natural image model which combines the power of edge statistics features of reconstructed images from the Cr channel and rake transform-based features from the Y channel are represented	In this method some forged images as well as forged images generated by using advanced image composition techniques trained classifiers fail in testing.
Chennamma, H. Rand, and Lalitha Rangarajan. [73]	Line-based calibration	In this method Edge feature is extracted and the Boosting Feature Selection (BFS) algorithm is used for optimal feature selection	86.0%	In this method We have used a lens radial distortion camera parameter (novel passive technique (with no signature or watermark) for detecting copy-paste forgery.	If two images having similar types of distortion are spliced together and the position of the copied and pasted portion is the same, In this scenario the proposed approach may fail to detect the spliced image.
He, Zhongwei, et al. [169]	Markov Machine + DCT+ DWT	This method extracted Markov features	89.76%	In this proposed method we handle a large number of developed features by using two methods SVM-RFE is utilized. Markov features derived from the transition probability matrices, named the Markov features in the DWT domain and expanded Markov features in the DCT domain.	The proposed result outperforms only a few states of arts. we need more improvement in this proposed method.
Alahmadi, Amani A., et al. [72]	SVM	This article uses edges and texture feature	97.0%	In this proposed method, with the help of LBP and DCT a novel splicing image forgery detection method has been proposed.	For JPEG compression DCT is robust but aggressive compression or multiple recompressions degrade its effectiveness.

Table 4 continued

Technique Proposed by	Classifier Used	Extracted Feature	Accuracy	Strength	Weakness
Amerini, Irene, et al. [170]	SVM + DCT	Differentiate and locate a single and a double JPEG compression in portions of an image is used as a feature	99.18%	The proposed method shows effectiveness with respect to diverse quality forgery dimensions, compressions, and multiple forgeries.	It is not able to deal with flat and heavily compressed image regions
El-Alfy et. al [75]	SVM + PCA	This technique extracts Markov features and combines them with spatial and Discrete Cosine Transform domains to detect the tampering operation.	98.22%	In this method to improve the accuracy we use a support vector machine (SVM) with RBF kernel	Disadvantages of DCT-based Markov features should provide a set of discriminative features with low correlation to each other and also affect the computational time .
Zhang, Qingbo, Wei Lu, and Jian Weng. [168]	Markov model + DCT	Markove feature is extracted	96.69%	Combination of SVM-RFE with SVM is used for many features. Experiment results show that the proposed method is extensible and effective for detecting gray-to-color image splice detection.	Performance of the proposed method may decrease for other datasets.
Rao, Yuan and Ni, Jiangqun [213]	SVM + CNN	With the help of pre-trained CNN we extract dense features from the test images	98.04%	This method is proposed for both copy move and splicing detection by using a convolutional neural network (CNN) which automatically learn hierarchical representations.	The proposed CNN model is not able to localize splicing region properly, With the help of fully connected CRF efficiency can be improved.
Alahmadi, Amani, et al. [74]	Support Vector Machine, LBP and DCT	Each DCT coefficient of all blocks are computed and used as features in this paper.	97.50%	On this method Chroma channel is used which perform better than other channel, use of this increase the accuracy.	The results of this method may vary after tuning the parameters using meta-heuristics methods.

Table 4 continued

Technique Proposed by	Classifier Used	Extracted Feature	Accuracy	Strength	Weakness
Chen, Can and McCloskey, Scott and Yu, Jingyi [212]	Intensity Gradient Histogram (IGH) and CNN	Edge feature is extracted	97.0%	In this method, non-linear CRFs is used. Which affects edges in terms of the intensity-gradient bivariate histograms. We also use a deep-learning framework to detect and localize forged edges.	Due to the presence of blocking artifacts and image quality, Many forensics are known to be sensitive.
Shah, Atif, and El-Sayed El-Alfy. [71]	LBP + DCT	DCT coefficients and Multi-Scale LBP is used as texture feature extractor in this method	97.30%	In this method, two technique DCT coefficients and Multi-Scale LBP is used for feature extraction to detect image splicing.	The proposed method struggles to detect other types of forgeries, object removal, or more advanced manipulations that do not involve texture duplication.
Liu, Yaqi, et al. [171]	Logistic Regression	Gray Level Co-occurrence Matrix(GLCM)	99.5%	DMAC network is based on a deep-learning solution for CISDL, and it's gained real-time performance for a large group of investigated images.	The disadvantage of this method is that it is difficult to detect regions that are under radical changes. Especially, both DMAC and DMVN are sensitive to scale change.
El-Latif, Abd and Eman, I and Taha, Ahmed, and Zayed, Hala H [214]	Convolution Neural Network (CNN), Haar Wavelet Transform (HWT) + SVM	In this particular algorithm, CNN is used to extract features, and then HWT is applied	96.36%	Deep learning approach CNN is utilized to extract features automatically from the color image. In this algorithm, HWT with CNN is used to generate the final features.	The proposed method is not able to localize the forged region in spliced images.

Table 4 continued

Technique Proposed by	Classifier Used	Extracted Feature	Accuracy	Strength	Weakness
Jaiswal & Srivastava [70]	Logistic Regression	Image feature like HoG, LTE and DWT and LBP is extracted	99.5%	Relevant set of feature is used to the classification of the spliced image correctly	We can improve the detection accuracy by using machine learning or deep learning technique for localization of spliced objects in an image
Hussien, Nadheer Younus and Mahmoud, Rasha O and Zayed, Hala Helmi [211]	Deep belief network-based classifier	Extracting the feature of images based on the analysis of color filter array (CFA)	95.05%	In this method, extracting the feature of images based on the analysis of color filter array (CFA) an automated image splicing forgery detection scheme is presented.	CFA artifacts create unique patterns that can be misinterpreted as evidence of manipulation even though they are actually genuine artifacts from the camera's sensor. This leads to false alarms and damages the credibility of the detection method.
Rao, Yuan and Ni, Jiangqun and Zhao, Huimin [213]	Deep convolutional neural network (CNN) + SVM	With the help of CNN extraction of the diverse and expressive residual features take place	97.50%	Using two branch CNN (extract the diverse and expressive residual features) leads to better detection accuracy.	When we deal with challenging viewpoints, lighting, or environmental changes Deep learning local descriptors might not generalize well to various image conditions and could lead to poor performance.

Table 4 continued

Technique Proposed by	Classifier Used	Extracted Feature	Accuracy	Strength	Weakness
Ahmed, Belal and Gulliver, T Aaron and alZahir, Saif [215]	ResNet-50 and ResNet-101 architecture	A Mask-RCNN model is used with the ResNet model to extract the initial feature map	96.70%	In this method to create the initial feature map for train the Mask-RCNN. A new backbone architecture is designed. Which is named ResNet-conv.	This method can not be used for forgery like copy-move forgery or image retouching.
Meena, Kunj Bihari, and Tyagi, Vipin [210]	SVM with RBF kernel	ResNet-50 network is used as a feature extractor	97.24%	This method is used deep learning based technique name 'Noiseprint'. Which is used to get the noise residual by extinguishing the image content with the help of the ResNet-50 network as a feature extractor.	This method failed to locate the exact spliced region in the forged image.

proposed method shows 82.6% TPR while FPR is only at 7.2%. The TPRs of the others are below 70% when FPR is at 7.2% 290 and the FPRs are over 20% when TPR is at 82.6%. So, the proposed method's performance is good in realistic scenarios. There may be some improvement in the proposed method. Optimizing the existing convolutional neural networks to fit splicing detection can be future work. another problem that may be caused is the over-fitting problem due to a lack of sufficient tampered images for training.

The drawback of the previously proposed techniques is removed in the paper [279]. Region proposal network and FCN together improve the learning of images. In this proposed method three FCNs (FCN8, FCN16, and FCN32) with different upsampling layers have been used and all the three FCNs are initialized from the VGG-16 network. Experiments were performed on public dataset DVMM dataset, CASIA v1.0 dataset, and CASIA v2.0 dataset. In this method to improve the performance we introduce a region proposal network (RPN) and condition random field (CRF). The use of CRF changes the whole network into an end-to-end learning system, that learns the parameters for the FCN and the CRF in a single unified deep network. After the convolutional layers and the deconvolutional layers, the obtained feature maps are fed into CRF and RPN separately to achieve two losses ($Loss_{CRF}$ and $Loss_{RPN}$). We get the final loss (87) with the help of these two losses.

$$Loss = \alpha Loss_{RPN} + (1 - \alpha) Loss_{CRF} \tag{87}$$

where α= weight
the probability y of this pixel with two weights β_1 and β_2 is given by

$$y = \beta_1 y_{32} + \beta_2 y_{16} + (1 - \beta_1 \beta_2) y_8 \tag{88}$$

Then, the final locating map m for each pixel is achieved by

$$m = \begin{cases} 1 & \text{y} \geq 0.5 \\ 0 & \text{else} \end{cases} \tag{89}$$

The algorithm has the following steps:

- We use an FCN32 network with five layers with a kernel size of 3Ã-3 of each convolutional layer. The numbers of kernels in five blocks are 64, 128, 256, 512, and 512 respectively.
- After the convolution of blocks, the obtained feature maps are used as RPN and CRF input separately.
- We calculate total loss with the help of two losses $Loss_{CRF}$ and $Loss_{RPN}$.
- The above steps first and second are for FCN32. FCN16 and FCN8 are similar to FCN32.
- with the help of (88) and (89) we combine the locating maps to get the final locating map.

Another method based on a characteristic of noise distribution in an image for splicing detection has been proposed in article [49]. According to the Columbia Image Splicing Detection Evaluation Data set, spliced location identification is one of the most difficult tasks [50], lot of researchers have developed many algorithms regarding splicing detection. The papers [48, 51] show CRF-based techniques for the detection of the spliced region of an image. Authors in this article [52] describe the detection of the spliced region of a different area of an image with the help of camera property by checking consistency evaluation. The following steps are the detection of the splicing region.

- Establishment data sets
- Calculating cluster centers
- FCM clustering results

5.4 Image retouching attack

Image retouching is also a type of image tampering method. In this process, manipulation of an image has been done by changing several properties of an image. In image processing, many researchers have devised various algorithms for the detection of image retouching. Image feature changes like color, blurred background, change, contrast, etc. are known as image retouching.

In this modern day of the internet, the creation of retouched images is very easy with the help of editing software. Platforms like movies, social media, magazines, etc. It uses a retouching method to enhance the image's appearance to look more attractive. With the help of retouching, enhancement in the image feature can be made to make the image more attractive. The manipulations which can be done in retouching are not always possible in forging. In the article, retouching is included because it includes the forging of the primary images. This method has been utilized to enhance a particular feature of the image for a certain purpose. Image retouching is not legal ethically [53].

5.4.1 Retouching tools

There are many well-known tools for retouching purposes. Some of them are mobile based and some are computer based. One mobile-based application is Beauty Plus. It contains many features like color enhancer, skin toner, and face slimming [54]. Adobe PhotoShop is one of the most used applications with different specific features such as Color Replacement, Patch, Healing Brush, Clone Stamp, and Pattern Stamp. This application is available in both paid and unpaid forms.

The image shown in Fig. 28 was released by the Iranian army to hyperbolize their army power by showing a retouched image with four missiles instead of three missiles.

5.5 Image retouching detection

Retouching detection of an image is a process of extracting the different feature blurring, color change, etc. of the forged image.

In the article [58], authors designed a classifier, which compares the change in the retouched and original image. This classifier worked efficiently for more number of operations. A new supervised deep learning algorithm has been developed to detect synthetically altered images by using CNN [58]. In this method, a CNN architecture is to extract nonoverlapping patches (size (64,64,3) or (128,128,3)) (only for retouching detection) from the image. Mathematically, the residual connection for wider networks can be summarized as:

$$y = F_1(x, \{W_{1i}\}) + F_2(x, \{W_{2i}\}) \tag{90}$$

For the classification of patches CNN performs very well, here our focus is to detect differences in tampered patches. The prediction of the CNN pipeline is therefore post-processed using:

$$Output = \frac{Totalno.ofpatchespredictedastampered}{Totalno.ofpatches} \times 100 \tag{91}$$

classification based on SVM learns the decision boundary on the training samples and give better results compared to the thresholding-based approach.

Comparison and future scope The performance of Deep learning-based Models is good in detecting double-compressed JPG images [204, 236]. Introducing tampering/retouching in an already compressed JPG image followed by re-compression afterward leads to double compression. In this article, researchers find another possible avenue for research in the classification of testing photo realism as well as generated images.

In this paper classification of different images has been done with ResNet neural networks [59] on the basis of local and global features.

In the paper [64], an algorithm has been described for Image Forgery Detection which is dependent on a technique for detecting patterns of sensor noise. The complete image has been scanned with a Markov random field instead of an individual pixel scan. Application of this method is at a large scale while the performance of this algorithm is not very good. In PRNU estimation We calculate the maximum likelihood of the PRNU from M-given images is computed as follows:

$$k = \frac{\sum_{m=1}^{M} y_m r_m}{\sum_{m=1}^{M} y_m^2} \tag{92}$$

where the weighting terms y_m account for the fact that dark areas of the image present an attenuated PRNU and hence should contribute less to the overall estimate.

Comparison and future scope In this paper we improve upon the seminal PRNU-based forgery detection technique described in the article [21] by converting the problem into a Bayesian framework, and by modeling the decision variables as a Markov random field, thus accounting for their spatial dependencies. In spite of the present advances, there is much scope for improvement. We are working to design a better and more robust predictor. In future research, we will focus on improving spatial resolution, which allows to detection of smaller forgeries (Table 5). Table 5 thumbnail the various retouching forgery detection techniques. Table 6 summaries various passive forgery detection techniques with their pros and cons.

5.6 Deep learning based image forgery detection

In the last few years, many researchers have devoted major efforts to develop image forgery detectors. In modern days deep learning plays a very crucial role in the image processing area. Current image forgery areas have high interest in CNN-based detectors, because of their promising results. We have seen exponential growth in the field of digital image forgery. Today image manipulation become very easy with the advancement of software, such as Adobe Photoshop or Gimp. Machine learning is a field of computer science that is a subset of artificial intelligence that gives computers the ability to learn without being explicitly programmed. It is used in various computational tasks where designing and programming explicit algorithms with good performance is not easy. In digital forensics, the detection of image forgery detection is of significant importance.

In the modern research age, deep learning plays a vital role in the image processing area. Its image feature extraction property makes it very important. Deep learning is a subpart of machine learning which is based on artificial neural networks. The key feature of Deep Learning is the use of deep neural networks, In deep neural networks multiple layers are interconnected. Deep learning algorithms are capable of automatically learning hierarchical

Table 5 Retouching Forgery Detection Techniques with Different Classifier

Techniques	Publisher & Year	Extracted Feature	Classifier	Accuracy of Detection
Avcibas et al. [65]	IEEE, 2004	In this article distortion between two images has been measured. This measurement is used as a feature in this experiment	Linear regression classifier	79.00-80.00%
Stamm & Liu [66]	IEEE, 2010	Detecting addition of noise feature and contrast enhancement	Thresholding classifier	98.00-99.00%
Cao et al. [63]	Elsevier, 2010	Detect color contrast with the help of gamma correction	Threshold classifier	85.0 - 95.0 %
Kee, Eric, and Hany Farid. [60]	National Academy of Sciences (United States of America), 2011	Extraction of the geometric and photometric feature.	Support Vector Regression	48.8 - 49.0%
Mahalakshmi, S. Devi, K. Vijayalakshmi, and S. Priyadharsini. [175]	Elsevier, 2012	This method detects operations such as contrast enhancement, re-sampling (rotation, rescaling), and histogram equalization	Fast Fourier Transform (FFT)	96.3 - 99.3%
Lin et al. [176]	IEEE, 2013	Interchannel correlation	Thresholding classifier	89.0 - 90.0%
Cao et al. [67]	IEEE, 2014	Histogram peak/gap artifacts	Thresholding classifier	99.95-99.99%
Ding, Feng, et al. [179]	IEEE, 2015	Rotation invariant LBP	Support Vector Machine (SVM)	89.00-90.00%

Table 5 continued

Techniques	Publisher & Year	Extracted Feature	Classifier	Accuracy of Detection
Bharati, Aparna, et al. [61]	IEEE, 2016	For classification of image facial patches and supervised features are extracted	Unsupervised DBM	87.00-99.00%
Bharati, Aparna, et al. [54]	IEEE, 2017	Histogram of Gradients (HOG) and Local Binary Pattern (LBP) are used as features in this method.	Subclass Supervised Sparse Autoencoder	94.2 - 94.3%
Jain, Anubhav, Richa Singh, and Mayank Vatsa. [58]	IEEE, 2018	Detect digital alteration in an image, which is GANs-based alterations and retouching.	CNN with SVM	99.65-99.83%
Dang, L Minh and Hassan, Syed Ibrahim and Im, Suhyeon and Moon, Hyeonjoon. [219]	Elsevier, 2019	Learned feature and Raw pixels	AdaBoost and XGBoost classifier	94.5 and 95.1%
Marra, Francesco and Saltori, Cristiano and Boato, Giulia and Verdoliva, Luisa. [220]	IEEE, 2020	Deep Learning Features	CNN + Incremental Learning	99.3%
Dang, Hao and Liu, Feng and Stehouwer, Joel and Liu, Xiaoming and Jain, Anil K. [221]	IEEE, 2020	Deep Learning Features	CNN + Attention Mechanism	AUC = 99.4% and EER = 3.4%

Table 6 Types of Tampering Attacks with Different Features and Classifier

Tampering Attacks	Features	Classifier	Performance	Pros	Cons
Copy-Move [28–38]	– Keypoint-based and block-based – CNN and GAN – Color feature (CF), Bit feature (BF) – SIFT SURF – Similar texture characteristics in the same image	– SIFER, FQRHFMs – SVM – Halftoning-based Block Truncation Coding (HBTC) – BusterNet – CNN-based classifier – Efficient Subwindow Search algorithm (ESS)	– F1-image = 0.9630, F1-pixel = 0.9601, CPU Time = 67.41 [28] – Accuracy = 77.49% [29] – Accuracy (classification) = 73.63% [30] – F1 = 0.8835, Precision = 0.6963, Recall = 0.8042 [31] – Accuracy = 98%, FNP = 0.02 [32] – Accuracy maximum for Threshold value = 0.45 [34] – Precision = 93.5%, Recall = 82.7% [35] – TPR = 100%, FPR = 0.04% [36]	– Detection rate is high with high accuracy. – In keypoint-based CMFD technique to detect the duplicate region of an image, the extracted feature is matched with the local image feature. – The Keypoint approach is more robust than block-based. – SIFT feature descriptors are very stable, This method shows very good results (robust) for image processing (Rotation, noise, JPEG compression, scaling, etc.).	– Some proposed work is very difficult to implement in practical work. – HBTC is suitable for compression only and gives the best result for grey-level images. – Block-based approach takes more computational time compared to keypoint to detect the copy-move forgeries. – Image tampering detection is very popular in the researcher domain, it should be extended to audio and video as well. – In the copy move forgery detection most of the algorithms have a large amount of time, which causes high false positives. – Operations such as scaling, rotation, JPEG compression, and Gaussian noise block-based approach fail.

Table 6 continued

Tampering Attacks	Features	Classifier	Performance	Pros	Cons
Splicing [39–48, 51, 52, 66, 85–88]	– Moments of characteristic functions of wavelet subbands – Markov transition – Hilbert-Huang transform (HHT) – Geometry invariants – Color sharpness – Statistical moments of run-length and edge detection – Chroma components from the grey level co-occurrence matrix	– RRU-Net – Logistic Regression – Support Vector Machine, LBP and DCT – SVM + PCA – Markov model + DCT – Line-based calibration – U-Net + Fine-tuning – Support vector machine (SVM) – Radial basis function (RBF) – Contrastive PCA (cPCA++) – CFA artifacts – Demosaicing regularity.	– Accuracy = 76% [39] – IoU Value = 0.687 [41] – IoU = 0.863 [42] – Accuracy = 89.14% [43] – Accuracy = 80.15% [44] – Accuracy 90.74% [51] – Accuracy 90.00% [52] – Precision = 80%, Recall = 70% [48] – Accuracy = 99.1% [87]	– In the method cPCA++ with the help of stochastic gradient descent and back-propagation up gradation of weights and classifier is not required. – Splicing attack detection algorithm computationally economical compared to copy move detection. – Perspective-constraint-based algorithms give very good output whether the images to be tested have been down-sampled or compressed with a low-quality factor.	– Less number of features is identified for tampered and un-tampered regions of an image. – Overfitting is a big problem in network training. This problem can be reduced with the help of a pre-trained network or encoder. – Most of the forgery detection algorithms detect only one type of tampering, so in the future we need to develop an algorithm for more than one tampering detection simultaneously. – Like most of the algorithms Perspective-constraint method is also proposed for one type of forgery detection.

Table 6 continued

Tampering Attacks	Features	Classifier	Performance	Pros	Cons
Retouching [54, 58–61, 63–65, 67]	– Color contrast with the help of gamma correction – Noise feature and contrast enhancement – Histogram peak – Histogram of Gradients (HOG) and Local Binary Pattern (LBP) – Interchannel correlation	– Support Vector Regression – Linear regression classifier – Thresholding classifier – Fast Fourier Transform (FFT) – Support Vector Machine (SVM) – Unsupervised DBM	– Detection Accuracy 95.9% [54] – Accuracy = 99.73% [58] – Accuracy = 96.2% [61] – Overall CPU time = 1.79s [64] – Accuracy = 100% Scaling (10%), 100% Rotation (5deg), 99% Brightness, 97.5% Histogram Eq., 100% Mix. [65] – [67]	– This algorithm [54] can be generalized in the future with various demographic areas. – The Deep learning model gives very convincing results on retouched image detection. – Digitally retouched image forgery detection is more complicated than makeup retouched image forgery detection. – By using an appropriate classifier and new feature retouching algorithm can perform more accurately.	– Due to the advanced forgeries tools and software it becomes very easy to manipulate the image which can not be detected by simple detection techniques. – Still there are very few algorithms that can detect practical manipulation. Automatic detection of Generative Adversarial Networks (GANs) generated images is still a challenging task. – In retouching detection some algorithm's computation time is high. So we need to develop a novel method with the help of existing methods or by developing a new method.

Table 6 continued

Tampering Attacks	Features	Classifier	Performance	Pros	Cons
Resampling [181–183, 185–189]	– Long-Short Term Memory (LSTM) – Measuring degree of saturated pixels per row/column – To detect resampling in re-compressed images.	– CNN-LSTM (Deep learning-based approach) – Constrained convolutional neural network (CNN) – Extremely randomized trees (ET) classifier – Novel method based on a new spectral interpolation model is proposed.	– Accuracy = 94.80% [181] – AUC $\geq$ 0.998 [182] – Accuracy = 97.88% [185] – AUC = 85 [187] – TPR = 98.1% [188] – Accuracy = Recognition Accuracy = 95.5% [189]	– The article [181] is identifying the image manipulations at pixel level with high precision. – Performance of constrained CNN output is more precise to detect resampling in re-compressed images compared to state of art methods. – In this proposed method good quality images in terms of visual and objective measure are obtained.	– Some algorithm is limited to only a few passive tampering detections. – This method [185] is not working if image is upscaled re-compressed with 120% with QF = 70 and 50%. – CFA pattern is not performing well for more than one image

Table 6 continued

Tampering Attacks	Features	Classifier	Performance	Pros	Cons
JPEG Compression [190–193, 195–199, 204]	– Markov random process with second-order statistics – Markov angle features – Quantization matrix – Localizing double JPEG compression	– Support vector machine (SVM) – Data-driven CNN – Markov in quaternion discrete cosine transform (QDCT) – Enhanced random perturbation-based method – Error-based statistical feature extraction method – DCT coefficient's histograms combined with a CNN deep model	– Accuracy = 98.29% [190] – Accuracy = 94% [191] – Accuracy = 93.72% [192] – Accuracy = 97.95& [193] – Accuracy = 88.53% [195] – TPR = 100 [196] – Accuracy = 93% [198] – Accuracy = 79.4% [204]	– In the article [191] Thresholding strategy reduces time complexity of the algorithm. – Markov features decrease the computation time of the algorithm in real time along with the intra-block correlation between block QDCT coefficients. – The proposed method gives better results for double JPEG compression and also has the pretty capacity to detect triple JPEG compression. – For database UCID, NRCS, and SYSU with various quality factors. This method [199] is outperformed compared to a state-of-the-art method. – In this proposed method output is good for small blocks, for QF2 < QF1 automatic localization is better.	– The performance may be degraded a little bit when we apply shifted double (SD) JPEG compression. – NA-DJPEG compression may not perform effectively for large sizes of images. – In this method [195] Due to the variable ratio of the selected JPEG coefficients for different images. It needs to be trained. – CNN time complexity is high in some cases, so a trade-off generation between the localization accuracy capability and the computational effort is required.

features and patterns with raw image data as input, This feature is highly helpful for tasks such as object detection, image recognition, image forgery detection, and image generation. In image processing implementation deep learning of different types of neural networks like Long Short Term Memory Networks (LSTMs) [181], convolution neural network (CNN) [245], AlexNet [246], Generative Adversarial Networks (GANs) [247], Multilayer Perceptrons (MLPs) [248], Deep Belief Networks (DBNs) [209], Autoencoders, [244] etc. for image feature extraction. The following Fig. 38 shows the basic architecture of the CNN-LSTM model for image forgery detection.

In this proposed [209] method, a novel deep learning approach to learn features in order to detect forged images for various types of image formats. The first step in this proposed method is Basic Features Generation. Further in this, we converted the image into a YCrCb color space after that image is segmented into 32 by 32 patches. In two Stage Training Hierarchy for Tampering Detection, the First stage uses stacked Autoencoders for Complex Feature Learning. To perform stacked autoencoders by the encoding of each layer forwards by using the following equations:

$$ae^{(l)} = f(z^{(l)}) \tag{93}$$

$$z^{(l)} = W^{(l)} ae^{(l)} + b^{(l)} \tag{94}$$

In the second step for tampered region detection, Context Learning is used. There should exist a consistent feature pattern among patches within this neighborhood.

$$N(p) = [y_p^0, y_p^1, y_p^0, ...y_p^k] \tag{95}$$

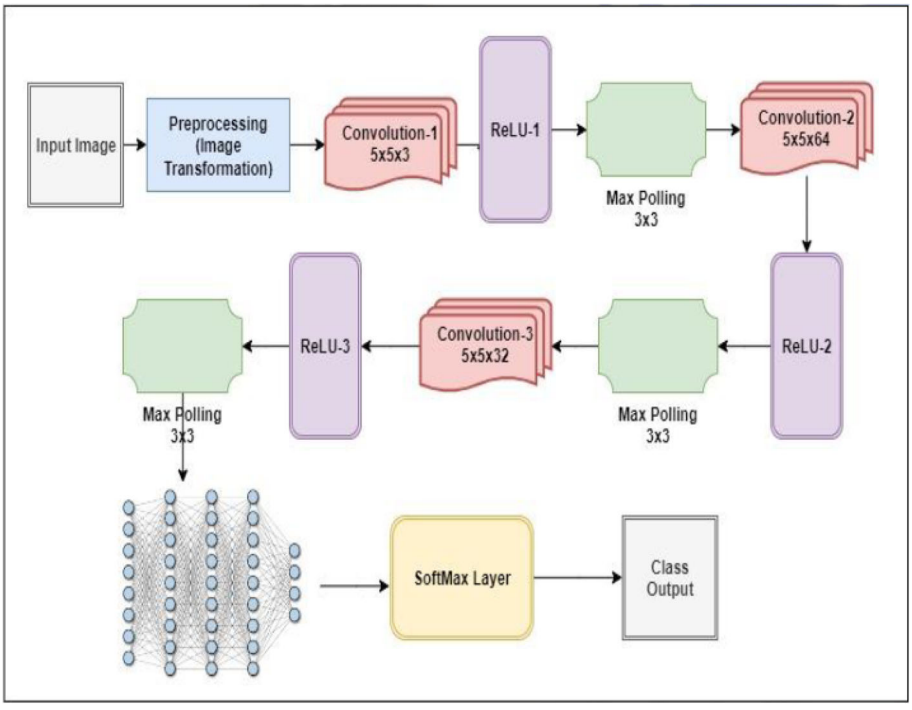

Fig. 38 CNN-LSTM Deep Learning Model for Image Forgery Detection

The performance of this proposed method has been compared with two state of art with a detection accuracy of 40.48% and 79.72%. The proposed method contains a detection accuracy of 87.51%.

Advantages and limitations This deep learning approach can detect tampering for region multi-format images. The limitation of this network is to not localize forged regions as our expectation for another file format. In the future, we can use other types of deep learning network models.

Another type of deep learning-based tampering detection is [240]. In this paper, an automatic resampling forgery using a deep learning method has been proposed. This method controls the false alarm rate using a-contrarian analysis with a two-step process. A-contrario methods come into the picture by Moisan, Desolneux, and Morel, with initially applied to identifying edges, and alignments. The a-contrario follows two-step inputs. The first step inputs one of the six heatmaps where each tested image block is mapped to a tampering confidence score between zero and one (high confidence). We input the original image I in the second input step. The a-contrario process thresholds the original mask M for binary 0-1 mask creation and then from the original image refines the mask using region proposals derived. When we Compared with the results from [207], the ROC curve increased by .08 using the a-contrario method using the same feature set with a different image scoring method.

Advantages and limitations Compared to previous work a-contrario procedure can increase the AUC measurement and we can apply a-contario procedure to all heatmaps bounded between zero and one. This method did not attempt to optimize the threshold (heatmaps value of 0.75) and through parameter tuning, there is still a chance of improvement in the presented a-contrario procedure.

Next deep learning-based method is presented by [241]. In this new deep learning-based forgery detection method for feature extraction an existing trained model AlexNet is used for image overlapped subblocks. This method has three steps:

- Feature extraction based on deep learning
- Matching of feature
- Post-processing

In the first step, the overlapped sub-square blocks are obtained from the image with a size of 16. After that block feature extraction takes place with the help of CNN architecture. We pre-trained the AlexNet model with 8 layers because of their less time for training. In feature matching, we have to calculate distances of vectors and this is compared with a predetermined threshold ∂, presented by following equation 96 to get the candidate matching vectors. (lets $\partial = 1.5$ empirically).

$$f^i = (f_1^i, f_2^i, f_3^i, ...f_{10}^i), \sqrt{\sum_{k=1}^{10}(f_k^i - f_k^j)^2} \leq \partial \tag{96}$$

To eliminate false matching we perform post-processing. We calculate upper left coordinates of the suspicious pairs the shift vectors $|x_i - x_j|, |y_i - y_j|$. To check image forgery we examine that the number of blocks that have same shift vector exceeds a predetermined threshold value ($\gamma = 32$ experimentally) We compare the proposed method with [184, 242], and find that our detects forged regions more truly than the others [184, 242].

Advantages and limitations The proposed AlexNet-based method has detection accuracy more than the referenced work even if the test image is similar or it has smooth regions.

Shallow Architecture of AlexNet compare to ResNet, VGG, or DenseNet. Limited Receptive Field and Overfitting may lead to less accuracy in forgery detection.

Success of deep learning in image forgery detection

- Because of the large amount of data handling property of Deep learning models it became a very important part of the image processing area.
- In image processing Deep learning models can automatically learn intricate features from raw image data. This reduces the time complexity of the algorithm.
- Due to the high detection accuracy rate of the CNN model, deep learning is very successful in copy move, splicing, retouching, and other forgery detection techniques.
- Deep learning models have a very successful rate in restoring damaged images, removing noise, and inpainting missing parts.
- To reduce the need for manual feature engineering and create more efficient pipelines we use deep learning models.

Challenges for deep learning in image forgery detection

- Training and deploying deep learning models sometimes need substantial computational resources, which can be a drawback of deep learning.
- Image forgery detection requires annotated data which can be scarce, emerging forgery techniques faces problem sometimes.
- Sometimes it becomes very difficult to understand why a model classified an image as forged, this is because of the complexity of deep learning models.
- It becomes very difficult to balance between false negatives (missing actual forgeries) and minimizing false positives (flagging non-forged images as forgeries) for an effective forgery detection system.
- Data privacy and security can also become a challenge in deep learning.

6 State of art research directions

This paper provides in-depth knowledge of image forgery attacks, which can be seen in real-life-based problems. The work done in the paper will be helpful for academicians, researchers, and project handlers in the future. In modern-day image forgery detection has become very important due to its demand in different fields. Many researchers have done tremendous work in this field. After this survey, some research gaps have been found as follows:

- We have reviewed many survey paper-based image feature extraction techniques and image forensic techniques, and the contribution of these papers is very significant in the field of image processing and signal processing. In this paper, we provide a detailed survey of different latest image feature extraction techniques and the application of these techniques in image forensics. Particularly, we explain the color, noise, texture, shape, corner, and keypoint feature extraction techniques.
- In this survey, different types of attack has been analyzed. In passive forensic techniques, some important forgery detection techniques (Copy-move, Splicing, Re-sampling, Retouching, etc.) have been discussed in detail. We have found a few papers based on human splicing face detection, we have seen some important examples of human face splicing detection on social media such as Facebook, WhatsApp, Instagram, etc., in future research can be explored in the area of human splicing detection.

- In copy-move forgery detection for similar texture, descriptors like (SHIFT [34, 112, 174], SURF [125, 129], ORB, etc.) are not performing well. Improvement in the performance of this descriptor with copy-move can be improved in the near future.
- Most of the proposed forgery detection techniques are effective for a limited type of image forgeries. We need to work on the technique that can fulfill the maximum requirement (An algorithm that can detect copy-move, splicing, and resampling forgeries at the same instant) simultaneously so, there is a need to develop a sophisticated, robust forgery detection technique that could reduce aforesaid limitations.
- In this survey paper, we have reviewed the different watermarking techniques [18–22, 27, 76–80]. Capacity, imperceptibility, security, and robustness are key parts of any watermarking technique. In the future, the researcher can develop a new watermarking technique that fulfills the above requirements simultaneously.
- Literature survey shows that in active detection technique quality of watermarked images is measured in terms of MSE (Mean Square Error) and PSNR (Peak Signal to Noise Ratio). the watermark may be of visible or invisible type and each method has its own strengths and weaknesses. In the ideal case, the value of PSNR & MSE should be infinite and zero respectively. But practically this is not possible we can only try to maximize PSNR and minimize MSE value to achieve better results.
- As we know many future works in the computer science field depend on AI, Machine Learning, and Deep learning [239, 254]. From the last decade, we can see that Deep Learning plays a major role in the field of image processing and computer vision, which has piqued academics' curiosity. Many researchers have carried out excellent work in this area by modifying convolutional neural network (CNN/ConvNet) designs to achieve the best result. Modification in loss functions, architectural innovations, optimization, application-specific modifications in the architecture, and Changes in activation functions can propose various learning algorithms in CNN.

The aim of this survey is to provide the literature survey for the academics and researchers under one umbrella, working in the domain of image processing using image feature extraction technique, different types of attacks on an image, and their detection techniques. In this paper assorted existing methods on active [14, 16, 18–22, 26, 27, 76, 78–82] and passive image forgery detection [28–48, 51, 52, 54, 58–60, 66, 85–88, 181–183, 185, 190–193, 195–198] are reviewed. A broad classification of image forgery detection techniques is also given with some latest image feature extraction methods. The focus of this review is mainly on a comprehensive overview of four main types of forgery detection techniques such as image splicing, copy-move, resampling, and retouching detection. Various existing methods have been reviewed in each category and observed that existing techniques suffer from many problems such as being Vulnerable to various attacks such as scaling, rotation, blurring, JPEG compression, and brightness adjustment. It is also observed that high Computational time and low accuracy. Most of these techniques are applied to images only, In the future, we can extend the research to audio and videos.

7 Conclusion

The image features are the backbone to implement any project in the field of image processing. This paper provides a detailed description of several image features with their extraction techniques in the field of digital image processing. Further, we have also presented different types of possible attacks and their detection techniques. Feature extraction is a popular and

useful step in the field of image forensics such as copy-move forgery, splicing detection, retouching, etc. The different types of image features such as edge, texture, corner, color, shape, and keypoints have been utilized in the field of image forensics. Among these features, keypoints play a major role in the copy move forgery detection along with good performance. The different types of forgery detection techniques available in the literature survey along with their comparative performance such as precision, recall, FPR, TPR, etc. have also been presented in the paper. The detailed survey performed in the paper also explores the state of art issues and research directions in the domain. This paper can be very useful for those who are working in the field of image forensics and image processing.

Acknowledgements We hereby declare that the information stated in the article is true to the best of my knowledge.

Data availability statement Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

Compliance with ethical standards

Conflicts of interests/competing interests The authors whose names are listed immediately below certify that they have no conflicts of interest to disclose. All authors declare that they have no conflicts of interest.

References

1. Zhang D, Lu G (2004) Review of shape representation and description techniques. Pattern Recognit 37(1):1–19
2. Deole PA, Longadge R (2014) Content based image retrieval using color feature extraction with KNN classification. IJCSMC 3(5):1274–1280
3. Kamboj P, Versha R (2013) A brief study of various noise model and filtering techniques. J Glob Res Comput Sci 4(4):166–171
4. Y. Tao et al (2003) A texture extraction technique using 2D-DFT and Hamming distance. In: Proceedings 5th international conference on computational intelligence and multimedia applications (ICCIMA). IEEE
5. Zhang G et al (2008) Shape feature extraction using Fourier descriptors with brightness in content-based medical image retrieval. In: 2008 International conference on intelligent information hiding and multimedia signal processing. IEEE
6. Mueggler E, Bartolozzi C, Scaramuzza D (2017) Fast event-based corner detection, pp 1–8
7. Duval-Poo MA, Odone F, De Vito E (2015) Edges and corners with shearlets. IEEE Trans Image Process 24(11):3768–3780
8. Bagri N, Johari PK (2015) A comparative study on feature extraction using texture and shape for content based image retrieval. Int J Adv Sci Technol 80(4):41–52
9. Sandhu A, Kochhar A (2012) Content based image retrieval using texture, color and shape for image analysis. Int J Comput Technol 3(1c):149–152
10. Datta R et al (2008) Image retrieval: ideas, influences, and trends of the new age. ACM Comput Surv (Csur) 40(2):1–60
11. Amaricai A, Gavriliu CE, Boncalo O (2014) An FPGA sliding window-based architecture harris corner detector. In: 2014 24th International conference on field programmable logic and applications (FPL). IEEE
12. Rashedi E, Nezamabadi-Pour H, Saryazdi S (2013) A simultaneous feature adaptation and feature selection method for content-based image retrieval systems. Knowl Based Syst 39:85–94
13. Canny J (1986) A computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell 6:679–698
14. Samcovic A, Turan J (2008) Attacks on digital wavelet image watermarks. J Electr Eng Bratisl 59(3):131
15. Semma A (2021) Writer identification using deep learning with fast keypoints and harris corner detector. Expert Syst Appl 184:115473
16. Saini LK, Shrivastava V (2014) Analysis of attacks on hybrid DWT-DCT algorithm for digital image watermarking with MATLAB. arXiv:1407.4738

17. Agarwal S, Pal Priyanka U (2015) Different types of attack in image watermarking including 2D, 3D Images. Int J Sci Eng Res 6(1)
18. Kumar S, Dutta A (2016) A study on robustness of block entropy based digital image watermarking techniques with respect to various attacks. In: 2016 IEEE International conference on recent trends in electronics, information & communication technology (RTEICT). IEEE
19. Singh P, Aayush A, Jyoti G (2013) Image watermark attacks: classification & implementation. Int J Electron Commun Technol 4(2)
20. Hartung FH, Su JK, Girod B (1999) Spread spectrum watermarking: malicious attacks and counterattacks. In: Security and Watermarking of Multimedia Contents, vol 3657. International Society for Optics and Photonics
21. Moulin P, O'Sullivan JA (2003) Information-theoretic analysis of information hiding. IEEE Trans Inf Theory 49(3):563–593
22. Su JK, Eggers JJ, Girod B (2001) Analysis of digital watermarks subjected to optimum linear filtering and additive noise. Signal Process 81(6):1141–1175
23. Kumar C, Singh AK, Kumar P (2018) A recent survey on image watermarking techniques and its application in e-governance. Multimed Tools Appl 77(3):3597–3622
24. Mohanty SP et al (2017) Everything you want to know about watermarking: from paper marks to hardware protection: from paper marks to hardware protection. IEEE Consum Electron Mag 6(3):83–91
25. Singh AK, Dave M, Mohan A (2014) Wavelet based image watermarking: futuristic concepts in information security. Proc Natl Acad Sci, India, Sect A 84:345–359
26. Sen J, Sen AM, Hemachandran K (2012) An algorithm for digital watermarking of still images for copyright protection. Indian J Comput Sci Eng 3(1):46–52
27. Voloshynovskiy S et al (2001) Attacks on digital watermarks: classification, estimation based attacks, and benchmarks. IEEE Commun Mag 39(8):118–126
28. Wang XY et al (2019) Copy-move forgery detection based on compact color content descriptor and Delaunay triangle matching. Multimed Tools Appl 78(2):2311–2344
29. Wu Y, Abd-Almageed W, Natarajan P (2018) BusterNet: detecting copy-move image forgery with source/target localization. In: Proceedings of the European conference on computer vision (ECCV)
30. Fan S et al (2017) Image visual realism: from human perception to machine computation. IEEE Trans Pattern Anal Mach Intell 40(9):2180–2193
31. Abdalla Y, Iqbal MT, Shehata M (2019) Copy-move forgery detection and localization using a generative adversarial network and convolutional neural-network. Information 10(9):286
32. Harjito B, Prasetyo H (2017) Passive copy-move forgery detection using halftoning-based block truncation coding feature. J Phys Conf Ser 855
33. Warbhe AD, Dharaskar RV, Thakare VM (2016) A survey on keypoint based copy-paste forgery detection techniques. Procedia Comput Sci 78:61–67
34. Huang H, Guo W, Zhang Y (2008) Detection of copy-move forgery in digital images using SIFT algorithm. In: 2008 IEEE pacific-asia workshop on computational intelligence and industrial application, vol 2. IEEE
35. Zhang C, Guo X, Cao X (2010) Duplication localization and segmentation. In: Pacific-rim conference on multimedia. Springer, Berlin
36. Amerini I et al (2011) A sift-based forensic method for copy-move attack detection and transformation recovery. IEEE Trans Inf Forensics Secur 6(3):1099–1110
37. Amerini I Irene et al (2010) Geometric tampering estimation by means of a SIFT-based forensic analysis. In: 2010 IEEE international conference on acoustics, speech and signal processing. IEEE
38. Kaur A, Sharma R (2013) Copy-move forgery detection using DCT and SIFT. Int J Comput Appl 70(7)
39. Bi X et al (2019) RRU-Net: The ringed residual U-net for image splicing forgery detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops
40. Ronneberger O, Fischer P, Brox T (2015) U-Net: convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer, Cham
41. Iglovikov V, Shvets A (2018) TernausNet: U-net with VGG11 encoder pre-trained on imageNet for image segmentation. Preprint at https://arxiv.org/abs/1801.05746
42. Çiçek Ö et al (2016) 3D U-Net: learning dense volumetric segmentation from sparse annotation. In: International conference on medical image computing and computer-assisted intervention. Springer, Cham
43. Shi YQ, Chen C, Chen W (2007) A natural image model approach to splicing detection. In: Proceedings of the 9th workshop on multimedia & security

44. Fu D, Shi YQ, Su W (2006) Detection of image splicing based on Hilbert-Huang transform and moments of characteristic functions with wavelet decomposition. In: International workshop on digital watermarking. Springer, Berlin
45. Salloum R, Kuo CC (2019) Efficient image splicing localization via contrastive feature extraction. arXiv:1901.07172
46. Abid A et al (2018) Exploring patterns enriched in a dataset with contrastive principal component analysis. Nat Commun 9(1):1–7
47. Johnson MK, Farid H (2005) Exposing digital forgeries by detecting inconsistencies in lighting. In: Proceedings of the 7th workshop on multimedia and security
48. Hsu YF, Chang SF (2010) Camera response functions for image forensics: an automatic algorithm for splicing detection. IEEE Trans Inf Forensics Secur 5(4):816–825
49. Zhang D et al (2019) Image splicing localization using noise distribution characteristic. Multimed Tools Appl 78(16):22223–22247
50. Columbia DVMM (2004) Research lab: columbia image splicing detection evaluation dataset. columbia. http://www.ee.columbia.edu/ln/dvmm/downloads/AuthSplicedDataSet/AuthSplicedDataSet.html
51. Hsu YF, Chang SF (2006) Detecting image splicing using geometry invariants and camera characteristics consistency. In: 2006 IEEE international conference on multimedia and expo. IEEE
52. Fang Z, Wang S, Zhang X (2009) Image splicing detection using camera characteristic inconsistency. In: 2009 International conference on multimedia information networking and security, vol 1. IEEE
53. Dhir V (2017) A review on image forgery & its detection procedure. Int J Adv Res Comput Sci 8(4)
54. Bharati A et al (2017) Demography-based facial retouching detection using subclass supervised sparse autoencoder. In: 2017 IEEE international joint conference on biometrics (IJCB). IEEE
55. Bhattacharjee S, Kutter M (1998) Compression tolerant image authentication. In: Proceedings 1998 international conference on image processing (Cat. No. 98CB36269), vol 1, IEEE
56. Stinson DR (1995) Collision-free hash functions. In: Cryptography theory and practice, pp 234–236
57. Lu CS, Liao HY (2003) Structural digital signature for image authentication: an incidental distortion resistant scheme. IEEE Trans Multimed 5(2):161–173
58. Jain A, Singh R, Vatsa M (2018) On detecting GANs and retouching based synthetic alterations. In: 2018 IEEE 9th international conference on biometrics theory, applications and systems (BTAS). IEEE
59. Szegedy C et al (2017) Inception-v4, inception-ResNet and the impact of residual connections on learning. In: 31st AAAI Conf Artif Intell 2017
60. Kee E, Farid H (2011) A perceptual metric for photo retouching. Proc Natl Acad Sci 108(50):19907–19912
61. Bharati A et al (2016) Detecting facial retouching using supervised deep learning. IEEE Trans Inf Forensics Secur 11(9):1903–1913
62. Mushtaq S, Mir AH (2014) Digital image forgeries and passive image authentication techniques: a survey. Int J Adv Sci Technol 73:15–32
63. Cao G, Zhao Y, Ni R (2010) Forensic estimation of gamma correction in digital images. In: 2010 IEEE international conference on image processing. IEEE
64. Chierchia G et al (2014) A Bayesian-MRF approach for PRNU-based image forgery detection. IEEE Trans Inf Forensics Secur 9(4):554–567
65. Avcibas I et al (2004) A classifier design for detecting image manipulations. In: 2004 International conference on image processing (ICIP'04), vol 4. IEEE
66. Stamm MC, Liu KR (2010) Forensic estimation and reconstruction of a contrast enhancement mapping. In: 2010 IEEE international conference on acoustics, speech and signal processing. IEEE
67. Cao G et al (2014) Contrast enhancement-based forensics in digital images. IEEE Trans Inf Forensics Secur 9(3):515–525
68. Zhao J, Guo J (2013) Passive forensics for copy-move image forgery using a method based on DCT and SVD. Forensic Sci Int 233(1–3):158–166
69. Sutthiwan P et al (2010) Rake transform and edge statistics for image forgery detection. In: 2010 IEEE international conference on multimedia and expo. IEEE
70. Jaiswal AK, Srivastava R (2020) A technique for image splicing detection using hybrid feature set. Multimed Tools Appl 1–24
71. Shah A, El-Alfy ES (2018) Image splicing forgery detection using DCT coefficients with multi-scale LBP. In: 2018 International conference on computing sciences and engineering (ICCSE). IEEE
72. Alahmadi AA et al (2013) Splicing image forgery detection based on DCT and local binary pattern. In: 2013 IEEE global conference on signal and information processing. IEEE
73. Chennamma HR, Rangarajan L (2011) Image splicing detection using inherent lens radial distortion. arXiv:1105.4712

74. Alahmadi A et al (2017) Passive detection of image forgery using DCT and local binary pattern. Signal, Image Video Process 11(1):81–88
75. El-Alfy ES, Qureshi MA (2015) Combining spatial and DCT based Markov features for enhanced blind detection of image splicing. Pattern Anal Appl 18(3):713–723
76. Lin CY et al (2001) Rotation, scale, and translation resilient watermarking for images. IEEE Trans Image Process 10(5):767–782
77. Kang X, Huang J, Zeng W (2010) Efficient general print-scanning resilient data hiding based on uniform log-polar mapping. IEEE Trans Inf Forensics Secur 5(1):1–12
78. Zheng D, Zhao J, El Saddik A (2003) RST-invariant digital image watermarking based on log-polar mapping and phase correlation. IEEE Trans Circuits Syst Video Technol 13(8):753–765
79. Gao X et al (2010) Geometric distortion insensitive image watermarking in affine covariant regions. IEEE Trans Syst, Man, Cybern, Part C (Applications and Reviews) 40(3):278–286
80. Ohbuchi R, Masuda H, Aono M (1998) Watermarking three-dimensional polygonal models through geometric and topological modifications. IEEE J Sel Areas Commun 16(4):551–560
81. Craver S et al (1997) On the invertibility of invisible watermarking techniques. In: Proceedings of international conference on image processing, vol 1. IEEE
82. Kutter M, Voloshynovskiy SV, Herrigel A (2000) Watermark copy attack. In: Security and watermarking of multimedia contents II, vol 3971. International Society for Optics and Photonics
83. Liu B, Pun CM (2018) Locating splicing forgery by fully convolutional networks and conditional random field. Signal Process Image Commun 66:103–112
84. Bianchi T, Piva A (2012) Image forgery localization via block-grained analysis of JPEG artifacts. IEEE Trans Inf Forensics Secur 7(3):1003–1017
85. Ye S, Sun Q, Chang EC (2007) Detecting digital image forgeries by measuring inconsistencies of blocking artifact. In: 2007 IEEE international conference on multimedia and expo. IEEE
86. Ferrara P et al (2012) Image forgery localization via fine-grained analysis of CFA artifacts. IEEE Trans Inf Forensics Secur 7(5):1566–1577
87. Cao H, Kot AC (2009) Accurate detection of demosaicing regularity for digital image forensics. IEEE Trans Inf Forensics Secur 4(4):899–910
88. Yao H et al (2011) Detecting image forgery using perspective constraints. IEEE Signal Process Lett 19(3):123–126
89. Chaudhari R, Patil AM (2012) Content based image retrieval using color and shape features. Int J Adv Res Electr Electron Instrum Eng 1(5):386–392
90. Xu X et al (2008) A spine X-ray image retrieval system using partial shape matching. IEEE Trans Inf Technol Biomed 12(1):100–108
91. Jain A, Muthuganapathy R, Ramani K (2007) Content-based image retrieval using shape and depth from an engineering database. In: International symposium on visual computing. Springer, Berlin
92. Cui FY, Zou LJ, Song B (2008) Edge feature extraction based on digital image processing techniques. In: 2008 IEEE international conference on automation and logistics. IEEE
93. Indriani OR et al (2017) Tomatoes classification using K-NN based on GLCM and HSV color space. In: 2017 International conference on innovative and creative information technology (ICITech). IEEE
94. Jobin CMC, Parvathi RMS (2011) Segmentation of medical image using clustering and watershed algorithms. Am J Appl Sci 8(12):1349
95. Sural S, Qian G, Pramanik S (2002) Segmentation and histogram generation using the HSV color space for image retrieval. In: Proceedings international conference on image processing, vol 2. IEEE
96. Bora DJ, Gupta AK (2016) AERASCIS: an efficient and robust approach for satellite color image segmentation. In: 2016 International conference on electrical power and energy systems (ICEPES). IEEE
97. Bora DJ, Gupta AK (2016) A new efficient color image segmentation approach based on combination of histogram equalization with watershed algorithm.Int J Comput Sci Eng 4(6):156–167
98. Süsstrunk S, Buckley R, Swen S (1999) Standard RGB color spaces. In: Color and imaging conference, vol 1999. Society for Imaging Science and Technology
99. Haralick RM, Shanmugam K, Dinstein IH (1973) Textural features for image classification
100. Ojala T, Pietikäinen M, Harwood D (1996) A comparative study of texture measures with classification based on featured distributions. Pattern Recognit 29)(1):51–59
101. Verma M, Raman B (2016) Local tri-directional patterns: a new texture feature descriptor for image retrieval. Digit Signal Process 51:62–72
102. Zhang X et al (2017) A study for texture feature extraction of high-resolution satellite images based on a direction measure and gray level co-occurrence matrix fusion algorithm. Sensors 17(7):1474
103. Vogel J, Schiele B (2006) Performance evaluation and optimization for content-based image retrieval. Pattern Recognit 39(5):897–909

104. Hota RN, Venkoparao V, Rajagopal A (2007) Shape based object classification for automated video surveillance with feature selection. In: 10th International conference on information technology (ICIT 2007). IEEE
105. Tsai YT, Shih HC, Huang CL (2006) Multiple human objects tracking in crowded scenes. In: 18th International conference on pattern recognition (ICPR'06), vol 3. IEEE
106. Quddus A, Fahmy MM (1999) An improved wavelet-based corner detection technique. In: 1999 IEEE International conference on acoustics, speech, and signal processing (Cat. No. 99CH36258), vol 6. IEEE
107. Harris C, Stephens M (1988) A combined corner and edge detector. In: Alvey vision conference, vol 15
108. Peng W et al (2016) Harris scale invariant corner detection algorithm based on the significant region. Int J Signal Process Image Process Pattern Recognit 9(3):413–420
109. Lotfian S, Foroosh H (2021) Multi-scale keypoint matching. In: 2020 25th International conference on pattern recognition (ICPR). IEEE
110. Qin J et al (2019) An encrypted image retrieval method based on Harris corner optimization and LSH in cloud computing. IEEE Access 7:24626–24633
111. Wang H, Wang H (2018) Perceptual hashing-based image copy-move forgery detection. Secur Commun Netw 2018
112. Wang C et al (2019) An image copy-move forgery detection method based on SURF and PCET. IEEE Access 7:170032–170047
113. Talib, A et al (2013) Efficient, compact, and dominant color correlogram descriptors for content-based image retrieval. In: Proceedings of the 5th international conferences on advances in multimedia, Venice, Italy
114. Karkanis SA et al (2003) Computer-aided tumor detection in endoscopic video using color wavelet features. IEEE Trans Inf Technol Biomed 7(3):141–152
115. Utenpattanant A, Chitsobhuk O, Khawne A (2006) Color descriptor for image retrieval in wavelet domain. In: 2006 8th International conference advanced communication technology, vol 1. IEEE
116. Nazir A et al (2018) Content based image retrieval system by using HSV color histogram, discrete wavelet transform and edge histogram descriptor. In: 2018 International conference on computing, mathematics and engineering technologies (iCoMET). IEEE
117. Riaz F et al (2012) Invariant gabor texture descriptors for classification of gastroenterology images. IEEE Trans Biomed Eng 59(10):2893–2904
118. Yang Y, Newsam S (2008) Comparing SIFT descriptors and Gabor texture features for classification of remote sensed imagery. In: 2008 15th IEEE international conference on image processing. IEEE
119. Popescu AC, Farid H (2005) Exposing digital forgeries by detecting traces of resampling. IEEE Trans Signal Process 53(2):758–767
120. Popescu AC, Farid H (2005) Exposing digital forgeries in color filter array interpolated images. IEEE Trans Signal Process 53(10):3948–3959
121. Dettori L, Semler L (2007) A comparison of wavelet, ridgelet, and curvelet-based texture classification algorithms in computed tomography. Comput Biol Med 37(4):486–498
122. Gómez F, Romero E (2011) Rotation invariant texture characterization using a curvelet based descriptor. Pattern Recognit Lett 32(16):2178–2186
123. Bayram S, Sencar HT, Memon N (2009) An efficient and robust method for detecting copy-move forgery. In: 2009 IEEE international conference on acoustics, speech and signal processing. IEEE
124. Li L et al (2013) An efficient scheme for detecting copy-move forged images by local binary patterns. J Inf Hiding Multimed Signal Process 4(1):46–56
125. Pandey RC et al (2015) Passive copy move forgery detection using SURF, HOG and SIFT features. In: Proceedings of the 3rd international conference on frontiers of intelligent computing: theory and applications (FICTA) 2014. Springer, Cham
126. Yang B et al (2018) A copy-move forgery detection method based on CMFD-SIFT. Multimed Tools Appl 77(1):837–855
127. Zheng J et al (2016) Fusion of block and keypoints based approaches for effective copy-move image forgery detection. Multidimens Syst Signal Process 27(4):989–1005
128. Soni B, Das PK, Thounaojam DM (2017) Blur invariant block based copy-move forgery detection technique using FWHT features. In: Proceedings of the international conference on watermarking and image processing
129. Shivakumar BL, Baboo SS (2011) Detection of region duplication forgery in digital images using SURF. Int J Comput Sci Issues 8(4):199
130. Ardizzone E, Bruno A, Mazzola G (2015) Copy-move forgery detection by matching triangles of keypoints. IEEE Trans Inf Forensics Secur 10(10):2084–2094
131. Kekre HB et al (2010) Image Retrieval using Texture Features extracted from GLCM, LBG and KPE. Int J Comput Theory Eng 2(5):695

132. Rampun A, Strange H, Zwiggelaar R (2013) Texture segmentation using different orientations of GLCM features. In: Proceedings of the 6th international conference on computer vision/computer graphics collaboration techniques and applications
133. Xie J et al (2010) Texture classification via patch-based sparse texton learning. In: 2010 IEEE international conference on image processing. IEEE
134. Tamura H, Mori S, Yamawaki T (1978) Textural features corresponding to visual perception. IEEE Trans Syst Man Cybern 8(6):460–473
135. Pu YF, Zhou JL, Yuan X (2009) Fractional differential mask: a fractional differential-based approach for multiscale texture enhancement. IEEE Trans Image Process 19(2):491–511
136. Rashid A (2016) Digital watermarking applications and techniques: a brief review. Int J Comput Appl Technol Res 5(3):147–150
137. Singh P, Chadha RS (2013) A survey of digital watermarking techniques, applications and attacks. Int J Eng Innov Technol 2(9):165–175
138. Milano D Harmonic incorporated (2012) Content control: digital watermarking and fingerprinting, White Paper. http://www.rhozet.com/whitepapers/Fingerprinting---Watermarking.pdf. Accessed 30 May
139. Agarwal N, Singh AK, Singh PK (2019) Survey of robust and imperceptible watermarking. Multimed Tools Appl 78(7):8603–8633
140. Qasim AF, Meziane F, Aspin R (2018) Digital watermarking: Applicability for developing trust in medical imaging workflows state of the art review. Comput Sci Rev 27:45–60
141. Allaf AH, Kbir MA (2018) A review of digital watermarking applications for medical image exchange security. In: The proceedings of the third international conference on smart city applications. Springer, Cham
142. Singh AK et al (eds) (2017) Medical image watermarking: techniques and applications. Springer
143. Amri H et al (2017) Medical image compression approach based on image resizing, digital watermarking and lossless compression. J Signal Process Syst 87(2):203–214
144. Zhou G, Lv D (2011) An overview of digital watermarking in image forensics. In: 2011 4th International joint conference on computational sciences and optimization. IEEE
145. Lan R, Zhou Y, Tang YY (2015) Quaternionic local ranking binary pattern: a local descriptor of color images. IEEE Trans Image Process 25(2):566–579
146. Fujieda S, Takayama K, Hachisuka T (2017) Wavelet convolutional neural networks for texture classification. arXiv:1707.07394
147. Ji H et al (2012) Wavelet domain multifractal analysis for static and dynamic texture classification. IEEE Trans Image Process 22(1):286–299
148. Lyu S, Pan X, Zhang X (2014) Exposing region splicing forgeries with blind local noise estimation. Int J Comput Vis 110:202–221
149. Huo LZ, Tang P (2011) Spectral and spatial classification of hyperspectral data using SVMs and Gabor textures. In: 2011 IEEE international geoscience and remote sensing symposium. IEEE
150. He L et al (2016) Discriminative low-rank Gabor filtering for spectral-spatial hyperspectral image classification. IEEE Trans Geosci Remote Sens 55(3):1381–1395
151. Rubel A et al (2016) Efficiency of texture image enhancement by DCT-based filtering. Neurocomputing 175:948–965
152. He C et al (2013) Texture classification of PolSAR data based on sparse coding of wavelet polarization textons. IEEE Trans Geosci Remote Sens 51(8):4576–4590
153. Julesz B, Bergen JR (1983) Human factors and behavioral science: textons, the fundamental elements in preattentive vision and perception of textures. Bell Syst Techn J 62(6):1619–1645
154. Lukas J, Fridrich J, Goljan M (2005) Determining digital image origin using sensor imperfections. In: Image and video communications and processing 2005, vol 5685. SPIE
155. Humeau-Heurtier A (2019) Texture feature extraction methods: a survey. IEEE Access 7:8975–9000
156. Hall-Beyer M (2017) Practical guidelines for choosing GLCM textures to use in landscape classification tasks over a range of moderate spatial scales. Int J Remote Sens 38(5):1312–1338
157. Meshkini K, Ghassemian H (2017) Texture classification using Shearlet transform and GLCM. In: 2017 Iranian conference on electrical engineering (ICEE). IEEE
158. Lloyd K et al (2017) Detecting violent and abnormal crowd activity using temporal analysis of grey level co-occurrence matrix (GLCM)-based texture measures. Mach Vis Appl 28(3–4):361–371
159. Deotale NT, Sarode TK (2019) Fabric defect detection adopting combined GLCM, Gabor wavelet features and random decision forest. 3D Res 10(1):5
160. Sumana IJ et al (2008) Content based image retrieval using curvelet transform. In: 2008 IEEE 10th workshop on multimedia signal processing. IEEE
161. Sumana IJ, Lu G, Zhang D (2012) Comparison of curvelet and wavelet texture features for content based image retrieval. In: 2012 IEEE international conference on multimedia and expo. IEEE

162. Monro DM, Rakshit S, Zhang D (2007) DCT-based iris recognition. IEEE Trans Pattern Anal Mach Intell 29(4):586–595
163. He XJ et al (2005) A new feature of uniformity of image texture directions coinciding with the human eyes perception. In: International conference on fuzzy systems and knowledge discovery. Springer, Berlin
164. Hemalatha S, Anouncia SM (2017) Unsupervised segmentation of remote sensing images using FD based texture analysis model and ISODATA. Int J Ambient Comput Intell (IJACI) 8(3):58–75
165. Chaudhuri BB, Sarkar N (1995) Texture segmentation using fractal dimension. IEEE Trans Pattern Anal Mach Intell 17(1):72–77
166. Zhu Y et al (2020) AR-Net: adaptive attention and residual refinement network for copy-move forgery detection. IEEE Trans Ind Inf 16(10):6714–6723
167. Wang Y, Kang X, Chen Y (2020) Robust and accurate detection of image copy-move forgery using PCET-SVD and histogram of block similarity measures. J Inf Secur Appl 54:102536
168. Zhang Q, Lu W, Weng J (2016) Joint image splicing detection in DCT and contourlet transform domain. J Vis Commun Image Represent 40:449–458
169. He Z et al (2012) Digital image splicing detection based on Markov features in DCT and DWT domain. Pattern Recognit 45(12):4292–4299
170. Amerini I et al (2014) Splicing forgeries localization through the use of first digit features. In: 2014 IEEE international workshop on information forensics and security (WIFS). IEEE
171. Liu Y et al (2019) Adversarial learning for constrained image splicing detection and localization based on atrous convolution. IEEE Trans Inf Forensics Secur 14(10):2551–2566
172. Jin G, Wan X (2017) An improved method for SIFT-based copy-move forgery detection using non-maximum value suppression and optimized J-Linkage. Signal Process Image Commun 57:113–125
173. Yu L, Han Q, Niu X (2016) Feature point-based copy-move forgery detection: covering the non-textured areas. Multimed Tools Appl 75(2):1159–1176
174. Warif NB et al (2017) SIFT-symmetry: a robust detection method for copy-move forgery with reflection attack. J Vis Commun Image Represent 46:219–232
175. Mahalakshmi SD, Vijayalakshmi K, Priyadharsini S (2012) Digital image forgery detection and estimation by exploring basic image manipulations. Digit Investig 8(3–4):215–225
176. Lin X, Li CT, Hu Y (2013) Exposing image forgery through the detection of contrast enhancement. In: 2013 IEEE international conference on image processing. IEEE
177. Muhammad G et al (2014) Image forgery detection using steerable pyramid transform and local binary pattern. Mach Vis Appl 25:985–995
178. Zhao X et al (2014) Passive image-splicing detection by a 2-D noncausal Markov model. IEEE Trans Circuits Syst Video Technol 25(2):185–199
179. Ding F et al (2014) Edge perpendicular binary coding for USM sharpening detection. IEEE Signal Process Lett 22(3):327–331
180. Fridrich J, Soukal D, Lukas J (2003) Detection of copy-move forgery in digital images. In: Proceedings of digital forensic research workshop, vol 3
181. Bappy JH et al (2019) Hybrid LSTM and encoder-decoder architecture for detection of image forgeries. IEEE Trans Image Process 28(7):3286–3300
182. Vásquez-Padín D, Comesana P, Pérez-González F (2015) An SVD approach to forensic image resampling detection. In: 2015 23rd European signal processing conference (EUSIPCO). IEEE
183. Birajdar GK, Mankar VH (2013) Digital image forgery detection using passive techniques: a survey. Digit Investig 10(3):226–245
184. Li J et al (2014) Segmentation-based image copy-move forgery detection scheme. IEEE Trans Inf Forensics Secur 10(3):507–518
185. Bayar B, Stamm MC (2017) On the robustness of constrained convolutional neural networks to jpeg post-compression for image resampling detection. In: 2017 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE
186. Hore A, Ziou D (2011) An edge-sensing generic demosaicing algorithm with application to image resampling. IEEE Trans Image Process 20(11):3136–3150
187. Feng X, Cox IJ, Doerr G (2012) Normalized energy density-based forensic detection of resampled images. IEEE Trans Multimed 14(3):536–545
188. Peng A et al (2015) Countering anti-forensics of image resampling. In: 2015 IEEE international conference on image processing (ICIP). IEEE
189. Su Y et al (2017) Hierarchical image resampling detection based on blind deconvolution. J Vis Commun Image Represent 48:480–490
190. Ding F et al (2020) METEOR: measurable energy map toward the estimation of resampling rate via a convolutional neural network. IEEE Trans Circuits Syst Video Technol 30(12):4715–4727

191. Chen C, Shi YQ, Su W (2008) A machine learning based scheme for double JPEG compression detection. In: 2008 19th International conference on pattern recognition. IEEE
192. Li B et al (2019) Detecting double JPEG compression and its related anti-forensic operations with CNN. Multimed Tools Appl 78(7):8577–8601
193. Wang J et al (2020) Non-aligned double JPEG compression detection based on refined Markov features in QDCT domain. J Real-Time Image Process 17(1):7–16
194. Yue G et al (2022) SMDAF: a novel keypoint based method for copy-move forgery detection. IET Image Process 16(13):3589–3602
195. Niu Y et al (2019) An enhanced approach for detecting double JPEG compression with the same quantization matrix. Signal Process Image Commun 76:89–96
196. Kumawat C, Pankajakshan V (2020) A robust JPEG compression detector for image forensics. Signal Process Image Commun 89:116008
197. Liu X et al (2019) Downscaling factor estimation on pre-JPEG compressed images. IEEE Trans Circuits Syst Video Technol 30(3):618–631
198. Valenzise G, Tagliasacchi M, Tubaro S (2012) Revealing the traces of JPEG compression anti-forensics. IEEE Trans Inf Forensics Secur 8(2):335–349
199. Yang J et al (2013) Detecting non-aligned double JPEG compression based on refined intensity difference and calibration. In: International workshop on digital watermarking. Springer, Berlin
200. Lu Z, Jiang X, Kot A (2017) A novel LBP-based color descriptor for face recognition. In: 2017 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE
201. Ashraf R et al (2018) Content based image retrieval by using color descriptor and discrete wavelet transform. J Med Syst 42(3):1–12
202. Patruno C et al (2019) People re-identification using skeleton standard posture and color descriptors from RGB-D data. Pattern Recognit 89:77–90
203. Xie G et al (2020) Combination of dominant color descriptor and Hu moments in consistent zone for content based image retrieval. IEEE Access 8:146284–146299
204. Wang Q (2016) Zhang R (2016) Double JPEG compression forensics based on a convolutional neural network. EURASIP J Inf Secur 1:1–12
205. Agarwal R, Verma OP (2020) An efficient copy move forgery detection using deep learning feature extraction and matching algorithm. Multimed Tools Appl 79(11):7355–7376
206. Chen X et al (2020) Automatic feature extraction in X-ray image based on deep learning approach for determination of bone age. Future Gener Comput Syst 110:795–801
207. Bunk, J et al (2017) Detection and localization of image forgeries using resampling features and deep learning. In: 2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW). IEEE
208. Sunitha K, Krishna AN (2020) Efficient keypoint based copy move forgery detection method using hybrid feature extraction. In: 2020 2nd international conference on innovative mechanisms for industry applications (ICIMIA). IEEE
209. Zhang Y et al (2016) Image region forgery detection: a deep learning approach. In: SG-CRC, pp 1–11
210. Meena KB, Tyagi V (2021) A deep learning based method for image splicing detection. J Phys Conf Ser 1714(1)
211. Hussien NY, Mahmoud RO, Zayed HH (2020) Deep learning on digital image splicing detection using CFA artifacts. Int J Sociotechnol Knowl Dev (IJSKD) 12(2):31–44
212. Chen C, McCloskey S, Yu J (2017) Image splicing detection via camera response function analysis. In: Proceedings of the IEEE conference on computer vision and pattern recognition
213. Rao Y, Ni J, Zhao H (2020) Deep learning local descriptor for image splicing detection and localization. IEEE Access 8:25611–25625
214. El-Latif EI et al (2019) A passive approach for detecting image splicing using deep learning and haar wavelet transform. Int J Comput Netw Inf Secur 11(5)
215. Ahmed B, Gulliver TA, alZahir S, (2020) Image splicing detection using mask-RCNN. Signal, Image Video Process 14(5):1035–1042
216. Islam A et al (2020) DOA-GAN: dual-order attentive generative adversarial network for image copy-move forgery detection and localization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition
217. Kuznetsov A (2019) Digital image forgery detection using deep learning approach. J Phys Conf Ser 1368(3)
218. Li Y, Zhou J (2018) Fast and effective image copy-move forgery detection via hierarchical feature point matching. IEEE Trans Inf Forensics Secur 14(5):1307–1322
219. Dang LM et al (2019) Face image manipulation detection based on a convolutional neural network. Expert Syst Appl 129:156–168

220. Marra F et al (2019) Incremental learning for the detection and classification of GAN-generated images. In: 2019 IEEE international workshop on information forensics and security (WIFS). IEEE
221. Dang H et al (2020) On the detection of digital face manipulation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition
222. Liu GH et al (2011) Image retrieval based on micro-structure descriptor. Pattern Recognit 44(9):2123–2133
223. Blum M et al (2012) A learned feature descriptor for object recognition in RGB-D data. In: 2012 IEEE international conference on robotics and automation. IEEE
224. Tian X et al (2014) Feature integration of EODH and Color-SIFT: application to image retrieval based on codebook. Signal Process Image Commun 29(4):530–545
225. Pun CM, Liu B, Yuan XC (2016) Multi-scale noise estimation for image splicing forgery detection. J Vis Commun Image Represent 38:195–206
226. Rocha A et al (2011) Vision of the unseen: current trends and challenges in digital image and video forensics. ACM Comput Surv (CSUR) 43(4):1–42
227. Gupta S, Mohan N, Kaushal P (2021) Passive image forensics using universal techniques: a review. Artif Intell Rev 1–51
228. Castillo Camacho I, Wang K (2021) A Comprehensive review of deep-learning-based methods for image forensics. J Imaging 7(4):69
229. Sharma M, Singh S (2001) Evaluation of texture methods for image analysis. In: The 7th Australian and new Zealand intelligent information systems conference. IEEE
230. Sharma M, Singh S (2002) A comparative study of Fourier descriptors for shape representation and retrieval. In: Proceeding 5th Asian conference on computer vision. Citeseer
231. Kauppinen H, Seppanen T, Pietikainen M (1995) An experimental comparison of autoregressive and Fourier-based descriptors in 2D shape classification. IEEE Trans Pattern Anal Mach Intell 17(2):201–207
232. Xiang Y et al (2017) An advanced rotation invariant descriptor for SAR image registration. Remote Sens 9(7):686
233. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60:91–110
234. Alcantarilla PF, Bartoli A, Davison AJ (2012) KAZE features. In: 12th European conference on computer vision, Florence, Italy
235. Rajawat M, Tomar DS (2015) A secure watermarking and tampering detection technique on RGB image using 2 level DWT. In: 2015 Fifth international conference on communication systems and network technologies. IEEE
236. Li B et al (2017) A multi-branch convolutional neural network for detecting double JPEG compression. arXiv:1710.05477
237. Khan MI, Rahman MM, Sarker MI (2013) Digital watermarking for image authenticationbased on combined DCT, DWT and SVD transformation. arXiv:1307.6328
238. Wang XY et al (2017) A new keypoint-based copy-move forgery detection for small smooth regions. Multimed Tools Appl 76:23353–23382
239. Ning X et al (2023) Hyper-sausage coverage function neuron model and learning algorithm for image classification. Pattern Recognit 136:109216
240. Flenner A et al (2018) Resampling forgery detection using deep learning and A-contrario analysis. arXiv:1803.01711
241. Muzaffer G, Ulutas G (2019) A new deep learning-based method to detection of copy-move forgery in digital images. In: 2019 Scientific meeting on electrical-electronics & biomedical engineering and computer science (EBBT). IEEE
242. Zandi M, Mahmoudi-Aznaveh A, Talebpour A (2016) Iterative copy-move forgery detection based on a new interest point detector. IEEE Trans Inf Forensics Secur 11(11):2499–2512
243. Liu C, Xu J, Wang F (2021) A review of keypoint's detection and feature description in image registration. Sci Program 2021:1–25
244. Bibi S et al (2021) Digital image forgery detection using deep autoencoder and CNN features. Hum Cent Comput Inf Sci 11:1–17
245. Marra F et al (2020) A full-image full-resolution end-to-end-trainable CNN framework for image forgery detection. IEEE Access 8:133488–133502
246. Samir S et al (2020) Optimization of a pre-trained AlexNet model for detecting and localizing image forgeries. Information 11(5):275
247. Sharma P, Kumar M, Sharma H (2023) Comprehensive analyses of image forgery detection methods from traditional to deep learning approaches: an evaluation. Multimed Tools Appl 82(12):18117–18150
248. Dansena P, Bag S, Pal R (2017) Differentiating pen inks in handwritten bank cheques using multi-layer perceptron. In: International conference on pattern recognition and machine intelligence. Springer, Cham

249. Routray S, Ray AK, Mishra C (2017) Analysis of various image feature extraction methods against noisy image: SIFT, SURF and HOG. In: 2017 Second international conference on electrical, computer and communication technologies (ICECCT). IEEE
250. Chandrasekhar V et al (2009) CHoG: compressed histogram of gradients a low bit-rate feature descriptor. In: 2009 IEEE Conference on computer vision and pattern recognition. IEEE
251. Albukhanajer WA, Briffa JA, Jin Y (2014) Evolutionary multiobjective image feature extraction in the presence of noise. IEEE Trans Cybern 45(9):1757–1768
252. Fan J, Cao H, Kot AC (2013) Estimating EXIF parameters based on noise features for image manipulation detection. IEEE Trans Inf Forensics Secur 8(4):608–618
253. Jacobson RE, Jenkin RB (2003) Modelling and application of contrast enhancement of visually indistinct colours using simple single band image capture techniques. System 10(10):5
254. Foote KD (2022) The history of machine learning and its convergent trajectory towards AI. Mach Learn City Appl Archit Urban Des 129–142
255. Bay H, Tuytelaars T, Van Gool L (2006) Surf: speeded up robust features. In: 9th European conference on computer vision, Graz. Springer, Berlin
256. Luo C et al (2019) Overview of image matching based on ORB algorithm. J Phys Conf Ser 1237(3)
257. Zhao Q et al (2009) Stone images retrieval based on color histogram. In: 2009 International conference on image analysis and signal processing. IEEE
258. Hafner J et al (1995) Efficient color histogram indexing for quadratic form distance functions. IEEE Trans Pattern Anal Mach Intell 17(7):729–736
259. Ganar AN, Gode CS, Jambhulkar SM (2014) Enhancement of image retrieval by using colour, texture and shape features. In: 2014 International conference on electronic systems, signal processing and computing technologies. IEEE
260. Swain MJ, Ballard DH (1991) Color indexing. Int J Comput Vis 7(1):11–32
261. Johnson GM et al (2010) Derivation of a color space for image color difference measurement. Color Res Appl 35(6):387–400
262. Shao H et al (2008) Image retrieval based on MPEG-7 dominant color descriptor. In: 2008 The 9th international conference for young computer scientists. IEEE
263. Gou H, Swaminathan A, Wu M (2009) Intrinsic sensor noise features for forensic analysis on scanners and scanned images. IEEE Trans Inf Forensics Secur 4(3):476–491
264. Mohanaiah P, Sathyanarayana P, GuruKumar L (2013) Image texture feature extraction using GLCM approach. Int J Sci Res Publ 3(5):1–5
265. Sobania A, Evans JPO (2005) Morphological corner detector using paired triangular structuring elements. Pattern Recognit 38(7):1087–1098
266. Koo KM, Cha EY (2017) Image recognition performance enhancements using image normalization. Human Cent Comput Inf Sci 7(1):1–11
267. Liu CL, Zhou XD (2006) Online Japanese character recognition using trajectory-based normalization and direction feature extraction. In: Tenth international workshop on frontiers in handwriting recognition. Suvisoft
268. Schadt EE et al (2001) Feature extraction and normalization algorithms for high-density oligonucleotide gene expression array data. J Cell Biochem 84(S37):120–125
269. Pei SC, Lin CN (1995) Image normalization for pattern recognition. Image Vis Comput 13(10):711–723
270. John J, Mini MG (2016) Multilevel thresholding based segmentation and feature extraction for pulmonary nodule detection. Procedia Technol 24:957–963
271. Han JG et al (2016) Efficient Markov feature extraction method for image splicing detection using maximization and threshold expansion. J Electron Imaging 25(2):023031–023031
272. Li W, Huang Q, Srivastava G (2021) Contour feature extraction of medical image based on multi-threshold optimization. Mobile Netw Appl 26:381–389
273. Dobreva ID et al (2021) Thresholding analysis and feature extraction from 3D ground penetrating radar data for noninvasive assessment of peanut yield. Remote Sens 13(10):1896
274. Chowdhary CL, Acharjya DP (2020) Segmentation and feature extraction in medical imaging: a systematic review. Procedia Comput Sci 167:26–36
275. Sezgin M, Sankur BL (2004) Survey over image thresholding techniques and quantitative performance evaluation. J Electron Imaging 13(1):146–168
276. Torres C, Gonzalez CI, Martinez GE (2022) Fuzzy edge-detection as a preprocessing layer in deep neural networks for guitar classification. Sensors 22(15):5892
277. Asghar K et al (2019) Edge-texture feature-based image forgery detection with cross-dataset evaluation. Mach Vis Appl 30(7–8):1243–1262
278. Xu Q et al (2014) A distributed canny edge detector: algorithm and FPGA implementation. IEEE Trans Image Process 23(7):2944–2960

279. Chen B et al (2018) An improved splicing localization method by fully convolutional networks. IEEE Access 6:69472–69480
280. Pandey RC, Singh SK, Shukla KK (2015) Passive copy-move forgery detection using speed-up robust features, histogram oriented gradients and scale invariant feature transform. Int J Syst Dyn Appl (IJSDA) 4(3):70–89
281. Pandey RC et al (2016) Image splicing detection using HMRF-GMM based segmentation technique and local noise variances. INROADS-An Int J Jaipur National University 5(1s):223–228

Dhirendra Kumar received M. Tech. degree in computer science and engineering from Motilal Nehru National Institute of Technology, Allahabad, India, in 2017 and He is pursuing PhD in computer science and Engineering from Dr. A.P.J. Abdul Kalam Technical University Lucknow U.P.

Ramesh Chand Pandey received M. Tech. degree in computer science and engineering from Thapar University Patiala in 2008 and a Ph.D. Degree in computer science and Engineering from IIT(BHU) in 2017. He is working as an assistant professor in the Department of Information Technology, at Rajkiya Engineering College Ambedkar Nagar. He has published nineteen various papers in reputed international Journals and Conferences with good citations.

Ashish Kumar Mishra received M.Tech degree in computer science and engineering from Motilal Nehru National Institute of Technology, Allahabad, India, in 2015. He is working as an assistant professor in the Department of Information Technology, at Rajkiya Engineering College Ambedkar Nagar. He received PhD Degree in computer science and Engineering from Motilal Nehru National Institute of Technology, Allahabad, India in 2020. His research interests include areas in Cloud Computing, Distributed Systems, Object Oriented Modeling, Formal Methods, Machine Learning, IoT. He has 23 International publications in various Conferences and Journals.