Image splicing detection with principal component analysis generated low-dimensional homogeneous feature set based on local binary pattern and support vector machine

Debjit Das[1] · Ruchira Naskar[1] · Rajat Subhra Chakraborty[2]

Abstract

Due to the wide availability of various ready-to-use image tampering software tools, image manipulation has become widespread nowadays. Tampered images are often used intentionally for unlawful and malicious purposes. One of the most common forms of image manipulation today includes the *Image Splicing* attack. Image splicing can be defined as generating a composite image by combining parts of different images from different sources, forming one artificially generated composite image. Given its prevalence, image splicing detection has a fundamental importance in digital forensics, and therefore, several research works have been carried out in the recent past to address this problem. In this paper, we propose an image forensic technique using a homogeneous image feature set of optimal dimensions for the successful detection of image splicing attacks. The *Columbia Image Splicing Detection Evaluation Dataset* has been used in our experiments, and different feature sets extracted from the available data have been adopted and experimented with. We have succeeded in finding a feature set of dimension as low as 31 comprising of *Local Binary Pattern* features for image splicing detection, obtaining state-of-the-art spliced image classification result using the *Support Vector Machine* classifier.

Keywords Classification · Digital forensics · Feature dimension · Image splicing · Local binary pattern · Principal component analysis

✉ Debjit Das
  2020itp002.debjit@students.iiests.ac.in

  Ruchira Naskar
  ruchira@it.iiests.ac.in

  Rajat Subhra Chakraborty
  rschakraborty@cse.iitkgp.ac.in

[1] Department of Information Technology, Indian Institute of Engineering Science and Technology, Shibpur, 711103, India

[2] Department of Computer Science and Engineering, Indian Institute of Technology, Kharagpur, 721302, India

1 Introduction

With the easy availability of various free and user-friendly image tampering software, image manipulation has become widespread nowadays. Image forgery can be intentionally performed to harm the social reputation of any individual or an organization as a whole; fraudsters can carry it out to entrap innocent users that may result in financial and emotional distress, more so given the massive growth of social networks popularity over the recent years [7, 24]. Such crimes also constitute enormous threats to national security, mainly brought about through producing forged identity verification documents or false identity proofs. Such image manipulation attacks have grown so sophisticated over time that even the most potent forensic measures often fail to detect their existence in a given image (or a scanned document), which otherwise looks completely natural. In digital forensics, the detection of image forgery has fundamental importance. Among the different classes of digital image forgery that have gained attention in the domain of digital forensic researches in recent years, the two most noteworthy probably are: *Image Splicing* [14, 29] and *Copy Move Forgery* [22, 33]. *Image Splicing* is the act of artificially generating a composite image by merging portions of multiple images from varied sources to form a single forged image. In case of *Copy Move Forgery*, one or more regions(s) of an image are duplicated within the same image, hence repeating or obscuring significant image objects. Hence, the major difference between these two classes of forgery lies in the fact that in the case of a spliced image, image textures and features undergo abrupt changes from region to region, whereas, in copy-move forgery, image statistical properties are preserved in the forged regions. This makes it even more challenging to perform forensic analyses on the spliced images. An example of image splicing is presented in Fig. 1.

Image splicing has grown significantly widespread today, but it is challenging to be detected since parts of the composite (spliced) image are taken from different image sources. Several research works have been reported on image splicing detection in digital images. The image splicing detection schemes proposed in state-of-the-art researches, were based on techniques such as *Hybrid feature set* [14], *Multi-size Block Discrete Cosine Transform (MBDCT)* [35], *motion blur* [4], *Hilbert-Huang Transform (HHT)* [6], *Camera Response Function(CRF)* [11], *Markov features* [10, 27], *Discrete Cosine Transform (DCT)* [13, 21], *Deep Matching and Validation Network (DMVN)* [39], *Gaussian blur* [5] etc. Such approaches posed splicing detection as a classification problem and majorly used Support Vector Machine (SVM) classifier for the classification purpose.

Most of the existing feature-based researches include multiple combined techniques and have used high-dimensional feature sets. In many approaches, high dimensional hybrid

Fig. 1 Splicing of image: (*left*) authentic image 1, (*middle*) authentic image 2, (*right*) spliced image made of authentic image 1 and 2

feature sets had also been used, combining different types of features. So, to reduce the complexity of detection schemes and to minimize the feature vector set dimension, in this work, we aim to find an effective image splicing detection technique based on a homogeneous feature set with an optimal feature dimension. In this paper, an image splicing detection technique has been proposed, which is based on *Local Binary Pattern (LBP)* features of images. We have succeeded in solving the image splicing detection problem with the help of a suitable homogeneous feature set whose usage leads to a classifier that can achieve high detection accuracy having significantly reduced dimension. Experimental results prove that our proposed model can achieve state-of-the-art accuracy for image splicing detection, despite being based on a low dimension homogeneous feature set.

1.1 Our motivation

The primary motivation of this work can be summarized as follows:

- To optimize the feature dimension used in state-of-the-art feature engineering-based splicing detection methods.
- To optimize the complexity of the splicing detection scheme by minimizing the feature set dimension and using only a homogeneous feature set.
- To develop a detection scheme solely based on one homogeneous feature set, unlike other state-of-the-art researches which combine multiple high dimensional feature sets for splicing detection in digital images. Most of the existing literature used hybrid feature sets, whereas we aim to generate an efficient homogeneous feature vector for image splicing detection.

The rest of this paper is organized as follows: Section 2 presents a literature survey of recent researches related to image splicing detection and our contribution in this paper. In Section 3, we present and discuss the proposed method in detail, along with a brief description of the features set selection and dimensionality reduction. In Section 4, we present and analyze the results of our experiments for different combinations of feature sets. Finally, in Section 5 we conclude the paper with a discussion on the possible future extension of the work.

2 Previous work and our contribution

In the present state-of-the-art, we majorly find researchers exploring diverse feature sets and classifiers, posing image splicing detection as a classification task in the digital forensics domain. This Section provides a brief overview of recent notable research advancements in this domain.

In one of the considerable recent research works on this problem [27], Pham et al. proposed a Markov feature-based method for the detection of image splicing. In this method, firstly, two types of Markov features, i.e., block-wise Markov features and coefficient-wise Markov features in the DCT domain, were extracted. Afterward, by combining these two types of Markov features, a feature vector was created and fed into the SVM classifier.

He et al. [10] proposed a Markov feature-based scheme in Discrete Wavelet Transform (DWT) and DCT domains for detection of image splicing. First of all, Markov features were generated in the DCT domain, and then in the DWT domain. Then SVM-RFE was applied, and finally, SVM was used for classification.

Zhao et al. [43] proposed a passive splicing detection scheme using a 2-D noncasual Markov model. In this method, an image was treated as a 2-D noncasual signal. This model was applied in the block DCT domain and the discrete Meyer wavelet transform domain, and finally, the cross-domain features were used for splicing detection.

Liu et al. [21] developed a scheme based on the correlation of nearby DCT coefficients. Firstly, they converted all the test images into JPEG images with the highest quality factor. Then, the neighboring joint density features of DCT coefficients and total features were extracted and input to a trained SVM classifier to detect splicing of images successfully.

Based on the natural image model, a passive splicing detection scheme had been given by Shi et al. [35]. Each test image is considered a 2-D array based on its spatial representation. By applying MBDCT to the test images, MBDCT 2-D arrays were generated. Finally, Markov transition probabilities and statistical features were extracted from the MBDCT 2-D array to form the feature vector for splicing detection.

Islam et al. [13] proposed an image manipulation detection technique based on DCT and LBP. This method can detect both image splicing as well as copy-move forgery.

In [14], Jaiswal et al. developed a novel scheme for the detection of image splicing using a hybrid feature set which is a mixture of texture and shape-based features. Different shape and texture-based features like DWT, Laws Texture Energy (LTE), Histogram of Oriented Gradients (HOG), and LBP features were extracted from images. Then selected features of each kind were used in particular combinations to form the hybrid feature set, which was then fed to a logistic regression classifier for the above purpose.

In [3], Chen et al. proposed a decision fusion-based method to detect and identify multiple image manipulating operations performed on the input image in image operator chains. They used similarity coefficient to find out the conflict between different forensic methods' obtained information and assigned credibility accordingly.

Liao et al.[19] developed an effective two-stream Convolutional Neural Network (CNN)-based architecture for detecting the image operator chain successfully. When an input image has undergone multiple consecutive image tampering operations, this scheme can catch all of the tampering operations performed and also the ordering of those operations. This method is capable of detecting the image operator chain even when the input image is JPEG compressed and without prior information about operating parameters.

In [18], Liao et al. developed a framework for parameter estimation of different operations in different image operator chains. They classified the degree of correlation of the image operator chains into two categories - coupled and uncoupled, based on the inherent correlation among different tampering operations. They extracted well-directed features for estimating the parameters for each operator chain, effectively even for the JPEG compressed images.

In [9], Han et al. developed an efficient splicing detection method of color images based on Markov features. Additionally, they proposed a threshold expansion algorithm to reduce the loss of information. They also proposed an even-odd state decomposition algorithm.

In [37], Wang et al. proposed an image splicing detection method based on Markov features and quaternion components in Quaternion DCT (QDCT) and Quaternion Wavelet Transform (QWT) domain. Finally, the extracted features were fed to an ensemble classifier.

In [42], Zhang et al. proposed a splicing detection method based on block DWT and Markov features. First of all, block DWT was applied on the input image and then markov features were extracted. Then, suitable features were selected using SVM-RFE. Finally, SVM classifier was used for classification.

In [17], Li et al. developed a model for image splicing detection of color images based on Markov features in QDCT transform. Here, a color input image was sub-divided into non-overlapping blocks of size 8×8 and then Red (R), Green (G), and Blue (B) components of each block were used to construct the quaternion matrix with QDCT transform. They calculated the intra-block difference and inter-block difference 2-D arrays for all four directions. Finally, the extended Markov features were fed to SVM classifier.

In [31], Rao et al. proposed a detection and localization method of image splicing based on local feature descriptor, using a deep CNN. At the first convolution layer of the CNN, the authors applied a constrained learning strategy. They incorporated the "block pooling" technique to generate discriminatory features for detecting image splicing.

In [2], Ahmed et al. developed a blind image tampering detection method, based on a deep learning neural network architecture, named *ResNet-conv*. The generated feature map had been used to train the supervised Mask-RCNN to generate masks for segments, which were spliced. For this purpose, two Residual Network (ResNet) architectures, viz. ResNet-50 and ResNet-101 were considered. A spliced image dataset based on the COCO dataset [20] had been used for training and evaluation purpose.

Another novel method based on CNN and *illuminant maps* had been proposed by Pomari et al. [29] for detecting spliced images. Here, automatically extracted deep splicing features were used for detecting image tampering. The SVM classifier is used to replace the top layer of ResNet-50 in this approach.

Stanton et al. [36] proposed a splicing detection method based on color phenomenology, which is known as *White Point Illuminant Consistency* or WPIC algorithm. In this approach, the segmented areas of an image were converted into chromaticity coordinates. Then they were compared with white points of the camera's EXIF file. Compared to the white point of EXIF, the chromaticity coordinates would have shifted illuminant color in case of a tampered image. By applying CNN they detected image tampering.

Wu et al. [38] proposed a novel method to detect spliced images, where an input color image was divided into multiple overlapping blocks. The illuminant color was estimated for every block, and after that, they calculated the difference between the reference illuminant color and the estimated color. If that difference exceeded a prefixed threshold value, that corresponding block was labeled as a spliced block.

In the method of Wu et al. [39], two images were taken as input images, one as a potential donor image and another one as a query image. Their model, called as DMVN, was capable of detecting image splicing and locating the spliced region. The authors developed a Visual Consistency Validator module to detect the spliced images efficiently.

Chen et al. [4] developed a method where they extracted features from a test image set by exploiting phase and magnitude information.

In [6], Fu et al. proposed a technique called as HHT, for extracting essential features from the test image set. They implemented a natural image model based on wavelet decomposition and moments of characteristic functions for the successful detection of image splicing.

Hsu et al. [11] described a scheme based on consistency checking, image segmentation, and geometry invariant CRF estimation for splicing detection. Here, a test image was partitioned into multiple arbitrary shaped parts, and then CRF was estimated from each part. Image features and CRF-based cross fitting were computed, and then those were fed to statistical classifiers.

In [25], Moghaddasi et al. proposed a low dimensional efficient method based on Singular Value Decomposition (SVD) features. Kernel Principal Component Analysis (PCA) was

also applied for improvement of the classification result. The SVM classifier was used for classification purpose.

Kakar et al. [15] developed an image splicing detection scheme by using discrepancies in motion blur. Firstly, the test image was subdivided into multiple overlapping blocks, and then for each block, motion blur was estimated. Then smoothing and up-sampling was performed. The image parts with inconsistent blur were marked as spliced regions in the original image.

In [5], Das et al. proposed a method where they initially evaluated the Gaussian blur of the test image and then computed the standard deviation of that blurred image and de-blur of the image was performed. In the spliced regions of the image, the ringing effect was more prominent and based on a prefix threshold value; those spliced regions could be separated from the input image.

2.1 Our contribution

Most of these existing image splicing detection techniques are a combination of multiple techniques. In the case of feature-based methods, most of the techniques consist of various features used to make a combined hybrid feature set. Besides, in most cases, the feature sets have relatively high dimensions (e.g., 14240 in [43]). The proposed model presented in this paper aims to overcome these limitations. Following are the significant contributions of this work:

- We have developed an image splicing detection scheme based on a homogeneous feature set instead of a hybrid feature set.
- The proposed scheme consists of a considerably low-dimensional feature set (31-dimensional) consisting of a single type of image feature.
- Our proposed approach helps to optimize the computational overhead of feature-based splicing detection due to adoption of one single feature type, of dimension as low as 31.
- Finally, the proposed scheme succeeds to obtain state-of-the-art accuracy in image splicing detection.

Comparative performance analysis has been performed and presented between the proposed approach and HOG, DWT, and LBP features of different dimension feature sets.
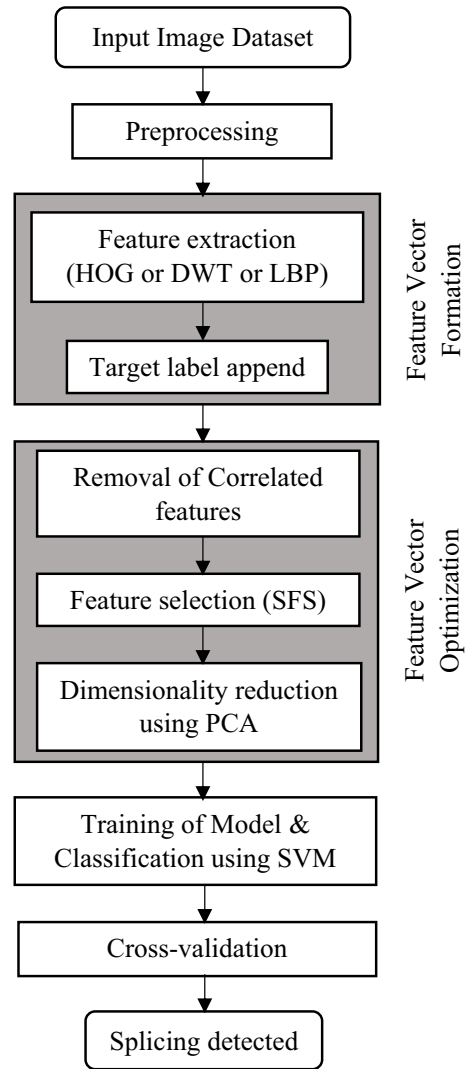
The following Section describes our proposed approach for splicing detection in images in detail.

3 Proposed scheme for image splicing detection with homogeneous low-dimensional feature set

Our work aims to detect image splicing based on a homogeneous feature set instead of a hybrid feature set. Besides, we also aim to optimize the dimension of the feature set to be used. We majorly work with three independent feature sets in this approach, viz. *HOG*, *DWT* and *LBP*. We find the best subset, which helps us achieve the above-stated aims while preserving state-of-the-art performance in splicing detection efficiency.

In Fig. 2, we present a block diagram to represent the operational flow of the proposed classification framework adopted in this work for image splicing detection. From the input dataset, each image first undergoes required preprocessing. Then, specified features are

Fig. 2 Operational flow of the proposed scheme

```
                    ┌─────────────────────────┐
                    │   Input Image Dataset   │
                    └─────────────────────────┘
                                 │
                    ┌─────────────────────────┐
                    │      Preprocessing      │
                    └─────────────────────────┘
                                 │
        ┌────────────────────────────────────────────┐
        │  ┌──────────────────────────────────────┐  │  Feature Vector
        │  │        Feature extraction            │  │  Formation
        │  │      (HOG or DWT or LBP)             │  │
        │  └──────────────────────────────────────┘  │
        │                    │                        │
        │  ┌──────────────────────────────────────┐  │
        │  │        Target label append           │  │
        │  └──────────────────────────────────────┘  │
        └────────────────────────────────────────────┘
                                 │
        ┌────────────────────────────────────────────┐
        │  ┌──────────────────────────────────────┐  │  Feature Vector
        │  │      Removal of Correlated           │  │  Optimization
        │  │           features                   │  │
        │  └──────────────────────────────────────┘  │
        │                    │                        │
        │  ┌──────────────────────────────────────┐  │
        │  │     Feature selection (SFS)          │  │
        │  └──────────────────────────────────────┘  │
        │                    │                        │
        │  ┌──────────────────────────────────────┐  │
        │  │    Dimensionality reduction          │  │
        │  │         using PCA                    │  │
        │  └──────────────────────────────────────┘  │
        └────────────────────────────────────────────┘
                                 │
                    ┌─────────────────────────┐
                    │    Training of Model &  │
                    │ Classification using SVM│
                    └─────────────────────────┘
                                 │
                    ┌─────────────────────────┐
                    │     Cross-validation    │
                    └─────────────────────────┘
                                 │
                    ┌─────────────────────────┐
                    │    Splicing detected    │
                    └─────────────────────────┘
```

extracted from each input image, along with each image's feature set, the target label value is also stored. After this, correlated features are identified and removed from this dataset. We take the help of *Sequential Forward Selection* (SFS) [23] method for choosing an optimal feature set in this work. After that *PCA* [1, 40] is applied for further dimensionality reduction. The modified feature set is fed into SVM classifier. 10-fold cross-validation has also been performed for validation checking.

3.1 Feature sets explored and their dimensionality

This Section provides a brief overview of three image feature sets explored in this work. These include the HOG, DWT, and LBP feature sets. Along with this, a basic idea about the dimensionality reduction technique is also presented here.

3.1.1 Histogram of Oriented Gradients (HOG)

In Image Processing and Computer Vision, HOG is widely used as a feature descriptor of images and for object detection purposes. To compute the HOG feature, the input image after necessary preprocessing is broken into many small connected squared regions, and then gradients, magnitude and orientation are calculated for each of those small regions. If G_x and G_y represent the small changes in X and Y directions respectively, (which are the gradients along X and Y axes), then for each pixel, magnitude and direction (orientation/value) of the angle can be found using the following:

$$Magnitude = \sqrt{G_x{}^2 + G_y{}^2} \tag{1}$$

$$Orientation\ or\ angle\ value\ \theta = \tan^{-1}\frac{G_y}{G_x} \tag{2}$$

Consequently, a separate histogram is generated for each of these small regions with the help of gradient and orientation information for the pixels in the region. Finally, the computed histograms are normalized and then concatenated to generate the HOG feature descriptor of the entire image.

3.1.2 Discrete Wavelet Transform (DWT)

DWT [14, 32] is widely used in image processing for denoising an image. In DWT, at discrete intervals, the wavelets are sampled. Common forms of wavelets in DWT are - Haar wavelet [30, 34], Daubechies wavelet [34] and Dual-tree Complex Wavelet Transform (DCWT) [8].

DWT can be applied to classify texture features, as described below. After preprocessing an input image, it is decomposed into small wavelet coefficients. Each decomposition level consists of two low-pass filters and two high-pass filters. Approximate and detailed information about the image are extracted by the low-pass filters and the high-pass filters, respectively. A 2D-DWT decomposes an image into four frequency sub-bands which are Low-Low, i.e., low frequency in both horizontal and vertical directions (also known as approximation coefficients or LL), Low-High, i.e., low frequency in horizontal and high frequency in the vertical direction (LH), High-Low, i.e., High frequency in horizontal and low frequency in the vertical direction (HL), and High-High, i.e., high frequency in both horizontal and vertical directions (HH). We can achieve further decomposition up to the desired level by decomposing Low-Low or LL sub-band successively. Usually, an image's 4 to 5 level decomposition is required to obtain the required information. Figure 3 presents the frequency representation of an image's fourth level two-dimensional DWT or 2D-DWT by decomposing the LL sub-band successively up to level 4.

3.1.3 Local Binary Pattern (LBP)

LBP [14, 28] is a widely used feature descriptor [12, 28, 41], adopted in image processing and pattern recognition applications. It is a texture-based feature used to extract information

Fig. 3 Frequency representation of 2D-DWT upto level 4

about local texture patterns in an image. The LBP feature generates a partially invariant vector to translation, scaling, and rotation from an input image. LBP labels the input image pixels, and finally, the result is produced as a binary number. First of all, the image is divided into multiple connected small square blocks or cells generally consisting of nine pixels. Then for each center pixel in the cell, eight neighboring pixel values are computed, and these values are considered in a clockwise or anticlockwise direction together. For each pixel in the cell, its LBP is computed as follows:

$$LBP_{p,r} = \sum_{p=0}^{p-1} s \cdot (g_p - g_c) \cdot 2^p \qquad (3)$$

Here, g_c indicates the intensity of the center pixel, and g_p indicates the intensity of a neighboring pixel on the circle with radius r. $LBP_{p,r}$ is the LBP value computed for p neighbors with radius r. The value of function $s(x)$ is 1, if $x \geq 0$, and 0 if $x < 0$. If the intensity of the center pixel is higher than or equal to a neighbor pixel, then a value of '1' is stored; otherwise, a '0' is stored. In this manner, an 8-bit binary number is generated finally. A histogram of a complete cell is computed and normalized. Finally, all the normalized histograms of all cells are merged, and hence LBP feature vector of the entire image is formed.

3.1.4 Dimensionality reduction

To get rid of the *curse of dimensionality* [16], we need to optimize the dimensionality of our feature set without losing relevant information. For this, *PCA* [1, 40], an unsupervised linear dimensionality reduction technique has been used in our approach. First of all, the target label column is ignored from a given dataset, and the rest is taken as a new dataset. Then, from the new dataset mean of each column or dimension is computed. Next, the covariance matrix of the entire dataset is computed, followed by eigenvectors and corresponding eigenvalues. The eigenvectors are sorted based on eigenvalues. Next, eigenvectors with the biggest eigenvalues are selected. Then the samples are transformed into a new subspace.

3.2 Proposed model for feature based splicing attack detection in images

Here, we present our proposed approach for spliced image detection based on the above-discussed feature sets. In this approach, firstly, each input image is converted into a grayscale image with the help of the following weighing method:

$$I = 0.299 \times R + 0.587 \times G + 0.114 \times B \tag{4}$$

where R, G and B represent *Red*, *Green* and *Blue* color components of an RGB image, respectively, and the gray scale intensity of the converted gray image is represented by I.

Specifically, in this work, we find that the features described in Section 3.1, viz. HOG, DWT, and LBP are among the most effective ones in image splicing detection and hence utilize those in our proposed model.

3.2.1 Using HOG features

In this approach, the resultant grayscale image is down-sampled so that the width to height ratio in its resolution scales to 1 : 2. After this preprocessing, while extracting the HOG features, the orientation value is set to 8, pixels per cell is set to (16, 16), and cells per block is set to (2, 2).

The HOG feature vectors, along with each image sample's corresponding target label, are stored to generate the training feature vector set.

3.2.2 Using DWT features

After converting all images into grayscale, DWT features are extracted. While extracting the DWT features, we have performed the simplest Haar wavelet transform [30] as it is discontinuous, simple, and resembles a one-step function. Haar is same as daubechies 1 or db1 wavelet transform. We have performed 5th level DWT feature extraction for each dataset image.

The DWT features, too, are stored along with corresponding target values for model training and subsequent validation.

3.2.3 Using LBP features

From each converted grayscale image, a uniform LBP feature vector is extracted. LBP encodes local texture information. We have not applied normalization to LBP features. The LBP features and corresponding target labels are stored to form the required feature matrix.

3.3 Optimizing the feature space and classification

In each of the above three approaches of using a homogeneous feature set, to reduce the feature space, the correlated features are found out and removed by setting a threshold value, i.e., if the correlation between any two features is equal or more than the prefixed threshold value then only one of those two features will be stored. Additionally, we adopt the *SFS* technique for feature selection over the reduced feature space; after that, to reduce the dimension of the feature set further, PCA is applied . Finally, this feature vector set is fed into SVM classifier. After shuffling, the feature set is broken into two subsets; 80 percent of it is used for training and 20 percent for testing purposes. We have also performed 10-fold cross-validation for accurate performance estimation of the proposed model.

3.4 Key parameters selection in the proposed model

Our experiment explored HOG, DWT, and LBP features to find the best homogeneous feature set. Then, we removed correlated features, applied SFS and PCA to select only the required features, and made the dimension of the feature vector as low as possible. Then we fed the feature vector to the SVM classifier. The key parameters of the proposed model are listed below:

- Parameters for HOG: during HOG feature extraction, the "orientation" value is set to 8 and number of pixels per cell is taken as (16,16), and we have also selected number of cells per block as (2,2).
- Parameters for DWT: we have selected the Haar Wavelet Transform or db1 to perform the single level 2-D DWT. We have performed DWT feature extraction up to level 5.
- Parameter for LBP: we have not applied normalization to LBP features and used default parameters for LBP feature extraction.
- Correlated features removal: to remove correlated features, the correlation threshold value is chosen empirically. We have used different threshold values in the range of 0.7 to 0.9, and 0.8 provides the best result for HOG while for DWT and LBP we received best accuracy with correlation threshold value 0.9. Table 1 presents the number of dropped correlated features for different correlation threshold values for HOG, DWT and LBP feature extraction.
- Parameters for Feature Selection: for feature selection we have used SFS along with KNN classifier. We set "forward" as True and "floating" as False, "scoring" as accuracy, and "CV" as 0, as we do not opt for cross-validation in this stage. The "k_features" value is different for different experiments; it specifies the number of features to be selected for feature selection purposes.
- Parameters for dimensionality reduction: for dimensionality reduction, we have adopted PCA. Here, the "n_components" value is taken as the required number of feature dimensions we wish to have after applying PCA. It varies for different experiments. For LBP-based feature extraction, we have finally concluded 31 to be the dimension of the most effective feature set.
- SVM classifier: for the Support Vector Classifier (SVC), the *Radial Basis Function (RBF)* kernel has been used.

4 Experiments, results and discussion

In this Section, we present our experimental details and the result of each experiment. Specifically, we have performed multiple experiments to determine the best individual type feature set for efficient image splicing detection.

4.1 Dataset and implementation

In our experiment, we have used the Columbia Image Splicing Detection Evaluation Dataset [26], which consists of 1845 numbers of 128×128 grayscale image blocks. Out of these 1845 images, 933 image blocks are authentic, and the remaining 912 are of spliced category. Authentic and spliced image blocks are further sub-divided into five sub-categories based on textured region, smooth region, and position of an object boundary between two regions.

Table 1 Correlated features removal for different correlation threshold values

Feature Set	Correlation Threshold	No. of correlated features dropped
HOG	0.70	417
	0.75	416
	0.80	410
	0.85	344
	0.90	102
DWT	0.70	63
	0.75	63
	0.80	63
	0.85	63
	0.90	21
LBP	0.70	46
	0.75	36
	0.80	30
	0.85	23
	0.90	08

In our experiment, we have used this entire dataset consisting of total 1845 images without any further modification.

We have implemented the proposed model with the help of Python 3.7.6, Jupyter notebook IDE and scikit-learn python library. This experiment is executed on a workstation with a 4th generation Intel i3-4005U CPU with a processor base frequency of 1.70 GHz.

4.2 Performance evaluation metrics

In our experiment, we have used SVC of SVM along with RBF kernel for classification of spliced and authentic images, based on the feature sets presented in Section 3. We have adopted a *10-fold cross validation* test method for an accurate evaluation of the proposed model; the average result of this is termed as the Cross-Validation Score. SVM classifier produces a confusion matrix, i.e., a 2×2 matrix that describes the performance of a model on classifier for a test dataset whose actual target label values are available. The confusion matrix consists of 4 values, which are True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). True Positive refers to the case where a test sample is actually positive and is also predicted rightly as positive. True Negative implies that a sample is negative and is also predicted correctly as negative. False Positive refers to a sample originally negative but wrongly predicted as positive, and False Negative suggests a sample is actually positive, but it is improperly predicted as negative.

Different performance metrics such as Accuracy, Precision, Recall (Sensitivity), and F1-score are computed from the confusion matrix. In the following, we define the performance evaluation metrics used in our work, one by one.

Accuracy provides a measure of the overall correctness of the model, i.e., how often it is correct. It is computed as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

If the given dataset is not correctly balanced, Accuracy can not be treated as a good performance metric. In such cases, other performance metrics like Precision, Recall, and F1-Score act as essential performance metrics for analyzing the performance of a classification model.

Precision tells when the model predicts positive how often it is actually positive. Model Precision is computed as:

$$Precision = \frac{TP}{TP + FP} \qquad (6)$$

Recall or Sensitivity is the ratio of true positives to all actual positives. It tells us how often the prediction is correct when the samples are actually positive. It is computed as:

$$Recall/Sensitivity = \frac{TP}{TP + FN} \qquad (7)$$

F1-Score is a versatile performance metric based on both Precision and Recall. When both Precision and Recall become high, F1-score will also be increased, and it becomes perfect 1 only when both Precision and Recall values become 1. Formulation of F1-Score is as follows:

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \qquad (8)$$

In the next Section, we present and analyze our experimental results, along with a comparative analysis between efficiencies of diverse feature sets in the problem of image splicing detection.

4.3 Experimental results and analysis

This Section presents our experimental results in detail and provides the performance analysis of the proposed model for image splicing detection. We have performed a separate experiment for each type of feature, and the comparative analysis of our results has been provided in Table 2.

We can observe from Table 2 that when we use a homogeneous feature set of dimension 672, made of HOG features without any feature optimization, achieve an accuracy of 0.67. After applying SFS on it, the feature dimension becomes 51, but accuracy drops to 0.66. After using PCA, the feature dimension is reduced to 41, but the accuracy is not improved. When only the DWT feature set is used in conjunction with a SVM classifier, the accuracy is 0.63. Then, applying SFS makes the detection accuracy as 0.64, and when PCA is applied on this, accuracy drops to 0.62. When a homogeneous feature set made of 59 LBP features is fed to the SVM, we obtain an accuracy of 0.76. When we have applied SFS on it, accuracy becomes 0.81, and when we apply PCA on the LBP feature set we receive an accuracy of 0.82. Finally, when a homogeneous LBP feature matrix after applying SFS and PCA, is input to the SVM classifier, we get the best result among all other feature sets, with a drastically reduced feature matrix dimension of 31. In this case, the proposed model achieved an Accuracy of 0.85, Precision of 0.82, Recall of 0.89, and F1-Score of 0.85. Hence, the performance of the splicing detection model is satisfactory in this case, despite using an individual homogeneous type feature set of a considerably low dimension feature vector, which is 31 only. Also, this assumes much low computational effort due to its simplicity of implementation requirements.

Next, we analyze the Receiver Operating Characteristic (ROC) of the proposed model. A ROC curve is a graphical plot that describes the performance and activity of a binary classification model at different classification threshold values. A ROC curve is plotted with two parameters: True Positive Rate (TPR) and False Positive Rate (FPR). TPR is plotted

Table 2 Comparative analysis of results obtained from different experiments

S. No.	Feature details	Feature dimension	Cross validation score	Accuracy	Precision	Recall	F1-Score
1	HOG	672	0.60	0.67	0.67	0.69	0.68
2	HOG with SFS	51	0.62	0.66	0.65	0.71	0.68
3	HOG with SFS and PCA	41	0.61	0.66	0.65	0.72	0.69
4	DWT	64	0.57	0.63	0.63	0.65	0.64
5	DWT with SFS	35	0.56	0.64	0.61	0.71	0.66
6	DWT with SFS and PCA	31	0.56	0.62	0.62	0.65	0.63
7	LBP	59	0.65	0.76	0.74	0.80	0.77
8	LBP with SFS	29	0.65	0.81	0.81	0.84	0.82
9	LBP with PCA	31	0.75	0.82	0.80	0.86	0.83
10	LBP with SFS and PCA	**31**	**0.75**	**0.85**	**0.82**	**0.89**	**0.85**

along the vertical axis, and FPR is plotted along the horizontal axis. The TPR and FPR components are computed as follows:

$$TPR = \frac{TP}{TP + FN} \tag{9}$$

$$FPR = \frac{FP}{FP + TN} \tag{10}$$

The ROC characteristics depicted by the proposed model in our experiments is presented in Fig. 4. Area Under the Curve (AUC) measures the total 2-Dimensional area under the ROC curve from (0,0) to (1,1). Here, we have used Support Vector Classifier (SVC). Our model received an AUC score of 0.85.

In Fig. 5, we represent the performance of our proposed scheme when we used only the LBP feature set after applying SFS and PCA on it. Performance metrics used are accuracy, precision, recall, F1-score and AUC score.
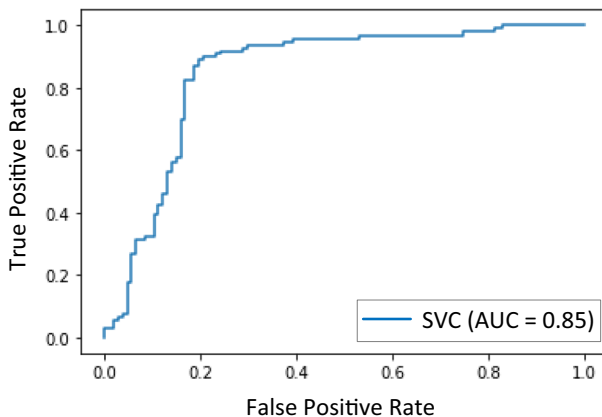
Fig. 4 ROC curve of our experiment based on LBP features with SFS and PCA
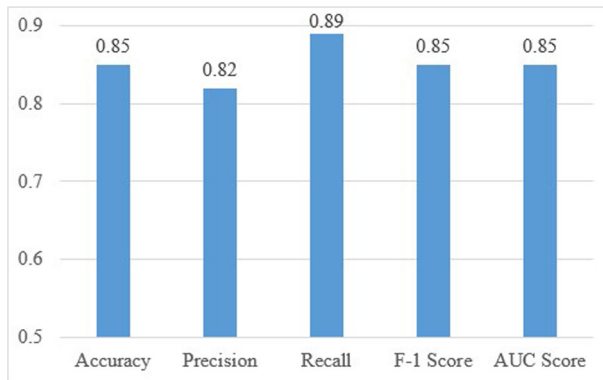
Fig. 5 Performance of the proposed model based on LBP with SFS and PCA
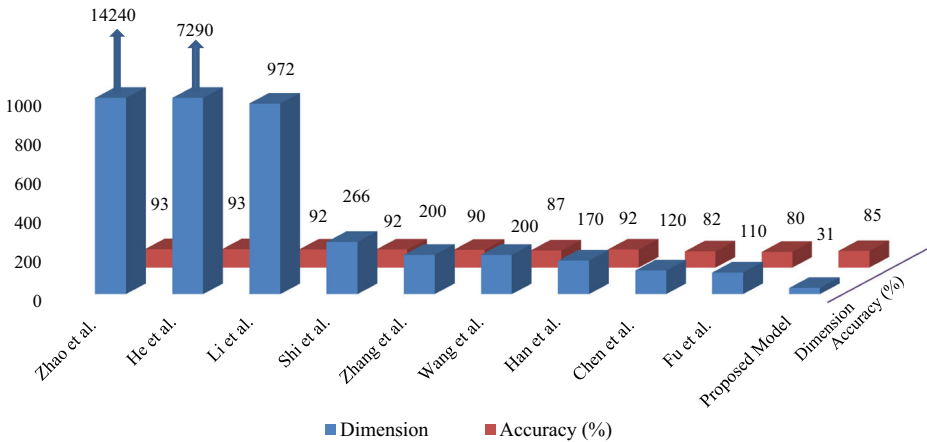
4.4 Comparison with state-of-the-art

We have performed the comparative analysis of the proposed model with some recent existing methods mentioned in the current literature. The results have been presented in Table 3. Most of the splicing detection methods have been applied to the Columbia Image Splicing Detection Evaluation Dataset [26]. From the comparative analysis presented in Table 3, it is clear that although Zhao et al. [43] and He et al. [10] obtained an accuracy of 0.93, they have used huge dimension feature sets (dimensions of 14240 and 7290 respectively). Li et al. [17] obtained an accuracy of 0.92 with high feature vector dimension of 972. Shi et al. [35] and Han et al. [9] also obtained a pretty high accuracy of 0.92, but still, their feature set dimension was 266 and 170, respectively. Zhang et al. [42] achieved an accuracy of 0.90 with feature vector dimension 200. Wang et al. [37] also used a feature vector of dimension 200 and achieved an accuracy of 0.87. Fu et al. [6] and Chen et al. [4] reported accuracies of 0.80 and 0.82 respectively, while their feature set size was 110 and 120 respectively.

Compared to these existing works, the proposed scheme of homogeneous feature set made of LBP, after applying SFS and PCA achieved an accuracy of 0.85, with a feature set of dimensions as low as 31. Besides, the proposed scheme is based on a homogeneous feature

Table 3 Comparative analysis of proposed model with other splicing detection methods

Sl. No.	Scheme	Dimension of feature set	Accuracy
1	Zhao et al. [43]	14240	0.93
2	He et al. [10]	7290	0.93
3	Li et al. [17]	972	0.92
4	Shi et al. [35]	266	0.92
5	Zhang et al. [42]	200	0.90
6	Wang et al. [37]	200	0.87
7	Han et al. [9]	170	0.92
8	Chen et al. [4]	120	0.82
9	Fu et al. [6]	110	0.80
10	Proposed Model	31	0.85

Fig. 6 Comparison of result of proposed scheme with recent state-of-the-art works

set which is faster and computationally simpler compared to most of the other methods, where hybrid feature sets were used.

A comparative analysis of our proposed scheme with other existing state-of-the-art methods has been given in Fig. 6. It is based on Accuracy and the dimension of the corresponding feature set used in that method.

5 Conclusion and future work

In this paper, an effective image splicing detection method has been proposed that uses only a homogeneous type of feature set. We also keep the dimension of the feature set very low for simpler and faster model building and inference. Experimental results prove that among different feature-based approaches, the proposed LBP feature-based splicing detection model provides the best results for image splicing detection when it is applied along with SFS and PCA for dimensionality reduction of the LBP feature set. We have used the Columbia Image Splicing Detection Evaluation Dataset, a standard dataset used in most related works.

The future scope of research in this direction includes experiments on different datasets having different sized color images that may have undergone various forms of post-processing, such as sharpening, cropping, compression, blurring, noise reduction, etc. Also, the performance of the presented splicing detection model can be further improved with more fine-tuning so that the detection results are further enhanced while preserving usage of individual feature sets and keeping the feature dimension low.

Declarations

Conflict of Interests The authors declare that they have no potential conflict of interest.

References

1. Abdi H, Williams LJ (2010) Principal component analysis. Wiley Interdisciplinary Reviews: Computational Statistics 2(4):433–459
2. Ahmed B, Gulliver TA et al (2020) Image splicing detection using mask-rcnn. SIViP, pp 1–8
3. Chen J, Liao X, Qin Z (2021) Identifying tampering operations in image operator chains based on decision fusion. Signal Process Image Commun 95:116287
4. Chen W, Shi YQ, Su W (2007) Image splicing detection using 2d phase congruency and statistical moments of characteristic function. In: Security, Steganography, and watermarking of multimedia contents IX, vol 6505, pp 65050R. International Society for Optics and Photonics
5. Das A, Medhi A, Karsh RK, Laskar RH (2016) Image splicing detection using gaussian or defocus blur. In: 2016 International conference on communication and signal processing (ICCSP), pp 1237–1241. IEEE
6. 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, pp 177–187. Springer
7. Gill NK, Garg R, Doegar EA (2017) A review paper on digital image forgery detection techniques. In: 2017 8th International conference on computing, communication and networking technologies (ICCCNT), pp 1–7. IEEE
8. Hadi SJ, Tombul M (2018) Streamflow forecasting using four wavelet transformation combinations approaches with data-driven models: a comparative study. Water Resour Manag 32(14):4661–4679
9. Han JG, Park TH, Moon YH, Eom IK (2016) Efficient markov feature extraction method for image splicing detection using maximization and threshold expansion. J Electron Imaging 25(2):023–031
10. He Z, Lu W, Sun W, Huang J (2012) Digital image splicing detection based on markov features in dct and dwt domain. Pattern Recogn 45(12):4292–4299
11. 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
12. Huang D, Shan C, Ardabilian M, Wang Y, Chen L (2011) Local binary patterns and its application to facial image analysis: a survey. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 41(6):765–781
13. Islam MM, Karmakar G, Kamruzzaman J, Murshed M, Kahandawa G, Parvin N (2018) Detecting splicing and copy-move attacks in color images. In: 2018 Digital image computing: techniques and applications (DICTA), pp 1–7. IEEE
14. Jaiswal AK, Srivastava R (2020) A technique for image splicing detection using hybrid feature set. Multimed Tools Appl, pp 1–24
15. Kakar P, Sudha N, Ser W (2011) Exposing digital image forgeries by detecting discrepancies in motion blur. IEEE Trans Multimedia 13(3):443–452
16. Kuo FY, Sloan IH (2005) Lifting the curse of dimensionality. Notices of the AMS 52(11):1320–1328
17. Li C, Ma Q, Xiao L, Li M, Zhang A (2017) Image splicing detection based on markov features in qdct domain. Neurocomputing 228:29–36
18. Liao X, Huang Z, Peng L, Qiao T (2021) First step towards parameters estimation of image operator chain. Inform Sci 575:231–247
19. Liao X, Li K, Zhu X, Liu KR (2020) Robust detection of image operator chain with two-stream convolutional neural network. IEEE Journal of Selected Topics in Signal Processing 14(5):955–968
20. Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P., Zitnick CL (2014) Microsoft coco: common objects in context. In: European conference on computer vision, pp 740–755. Springer
21. Liu Q, Sung AH (2009) A new approach for jpeg resize and image splicing detection. In: Proceedings of the first ACM workshop on multimedia in forensics, pp 43–48
22. Mahmood T, Nawaz T, Irtaza A, Ashraf R, Shah M, Mahmood MT (2016) Copy-move forgery detection technique for forensic analysis in digital images. Mathematical Problems in Engineering 2016
23. Marcano-Cedeño A, Quintanilla-Domínguez J, Cortina-Januchs M, Andina D (2010) Feature selection using sequential forward selection and classification applying artificial metaplasticity neural network. In: IECON 2010-36th annual conference on IEEE industrial electronics society, pp 2845–2850. IEEE
24. Meena KB, Tyagi V (2019) Image forgery detection: survey and future directions. In: Data, engineering and applications, pp 163–194. Springer
25. Moghaddasi Z, Jalab HA, Noor RM (2017) Image splicing detection using singular value decomposition. In: Proceedings of the second international conference on internet of things, data and cloud computing, pp 1–5

26. Ng TT, Hsu J, Chang SF (2009) Columbia image splicing detection evaluation dataset. DVMM lab Columbia Univ CalPhotos Digit Libr
27. Pham NT, Lee JW, Kwon GR, Park CS (2019) Efficient image splicing detection algorithm based on markov features. Multimed Tools Appl 78(9):12405–12419
28. Pietikäinen M (2010) Local binary patterns. Scholarpedia 5(3):9775
29. Pomari T, Ruppert G, Rezende E, Rocha A, Carvalho T (2018) Image splicing detection through illumination inconsistencies and deep learning. In: 2018 25th IEEE International conference on image processing (ICIP), pp 3788–3792. IEEE
30. Porwik P, Lisowska A (2004) The haar-wavelet transform in digital image processing: its status and achievements. Machine Graphics and Vision 13(1/2):79–98
31. Rao Y, Ni J, Zhao H (2020) Deep learning local descriptor for image splicing detection and localization. IEEE Access 8:25611–25625
32. Rinky B, Mondal P, Manikantan K, Ramachandran S (2012) Dwt based feature extraction using edge tracked scale normalization for enhanced face recognition. Procedia Technol 6:344–353
33. Shahroudnejad A, Rahmati M (2016) Copy-move forgery detection in digital images using affine-sift. In: 2016 2nd International conference of signal processing and intelligent systems (ICSPIS), pp 1–5. IEEE
34. Sharif I, Khare S (2014) Comparative analysis of haar and daubechies wavelet for hyper spectral image classification. The International Archives of Photogrammetry. Remote Sens Spat Inf Sci 40(8):937
35. Shi YQ, Chen C, Chen W (2007) A natural image model approach to splicing detection. In: Proceedings of the 9th workshop on Multimedia & security, pp 51–62
36. Stanton J, Hirakawa K, McCloskey S (2019) Detecting image forgery based on color phenomenology. In: CVPR workshops, pp 138–145
37. Wang R, Lu W, Li J, Xiang S, Zhao X, Wang J (2020) Digital image splicing detection based on markov features in qdct and qwt domain. In: Digital forensics and forensic investigations: breakthroughs in research and practice, pp 61–79. IGI Global
38. Wu X, Fang Z (2011) Image splicing detection using illuminant color inconsistency. In: 2011 Third international conference on multimedia information networking and security, pp 600–603. IEEE
39. Wu Y, Abd-Almageed W, Natarajan P (2017) Deep matching and validation network: an end-to-end solution to constrained image splicing localization and detection. In: Proceedings of the 25th ACM international conference on Multimedia, pp 1480–1502
40. Xiao B (2010) Principal component analysis for feature extraction of image sequence. In: 2010 International conference on computer and communication technologies in agriculture engineering, vol 1, pp 250–253. IEEE
41. Xiao B, Wang K, Bi X, Li W, Han J (2018) 2d-lbp: an enhanced local binary feature for texture image classification. IEEE Trans Circuits Syst Video Technol 29(9):2796–2808
42. Zhang Q, Lu W, Wang R, Li G (2018) Digital image splicing detection based on markov features in block dwt domain. Multimed Tools Appl 77(23):31239–31260
43. Zhao X, Wang S, Li S, Li J (2014) Passive image-splicing detection by a 2-d noncausal markov model. IEEE Trans Circuits Syst Video Technol 25(2):185–199

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.