The Visual Computer (2022) 38:2999–3025
https://doi.org/10.1007/s00371-021-02173-8

ORIGINAL ARTICLE

HyFiPAD: a hybrid approach for fingerprint presentation attack detection using local and adaptive image features

Deepika Sharma[1] · Arvind Selwal[1]

Accepted: 25 May 2021 / Published online: 14 June 2021

Abstract

With the pervasiveness of secured biometric authentication applications, the fingerprint-based identification system has fascinated much attention recently. However, the major detriment is their recognition sensors are vulnerable to presentation or spoofing attacks from fake fingerprint artifacts. To resolve these issues, a viable anti-deception countermeasure known as presentation attack detection (PAD) mechanism is developed. As handcrafted feature-based classification techniques exhibit encouraging results in computer vision, they are widely employed in fingerprint spoof detection. Notably, the single-feature-based techniques do not perform uniformly over different spoofing and sensing technologies. In this research work, we expound a new hybrid fingerprint presentation attack detection approach (HyFiPAD) that discriminates live and fake fingerprints using majority voting ensemble build on three local and adaptive textural image features. We propose a new descriptor (i.e., a variant of LBP) which is termed as Local Adaptive Binary Pattern (LABP). Thus, the notion of proposed LABP is used to extract more detailed micro-textural features from the fingerprint images. Our LABP features are combined with an existing Complete Local Binary Pattern (CLBP) descriptor to learn two respective SVM classifiers and additionally a sequential model is trained with the manually extracted Binary Statistical Image Features (BSIF). The experiments are performed on benchmark anti-spoofing datasets namely; LivDet 2009, LivDet 2011, LivDet 2013, and LivDet 2015, where an average classification error rate (ACER) of 4.11, 3.19, 2.88, and 2.97% is, respectively, achieved. The overall experimental analysis of the HyFiPAD demonstrates superiority against majority of the state-of-the-art methods. In addition, the proposed technique yields a promising performance on cross-database and cross-sensor liveness detection tests, claiming good generalization capability.

Keywords Fingerprint biometrics · Spoof attacks · Presentation attack detection · Image features · Sequential model

1 Introduction

With an expeditious growth in the demand for secured authentication, the conventional methods are being replaced with contemporary biometrical approaches as they cannot be stolen, misplaced, transferred or forged [1, 2]. Dominantly, the fingerprint biometric is widely deployed for this task due its higher reliability and low sensing cost with user's convenience. Thus, the fingerprint-based recognition systems are utilized in several applications including law enforcement, border control, cell-phone authentication, physical and logical access, citizen identification, healthcare and subsidies, commercial applications and public security [3]. However, these systems endure a variety of attacks threatening the security that has hindered their use in various computing applications. In a biometric system, out of eight vulnerable attack points, the presentation or spoof attacks is regarded as most commonly attempted as it does not require any internal information [4]. Generally, the presentation attack in fingerprint biometrics is an attempt by an imposter to capture the biometrical subsystem by presenting a fake fingerprint artifacts usually created from glue, wax, and play-doh. Few samples of fingerprint artifacts created through cooperation and non-cooperation of users and thereby acquired with a variety of sensing devices are shown in Fig. 1.

To alleviate presentation attacks, a fingerprint PAD (FiPAD) a.k.a liveness detection module is integrated with the biometric recognition system which serves as a security

✉ Deepika Sharma
sharmadeepika749@gmail.com

[1] Department of Computer Science and Information Technology, Central University of Jammu, Samba, Jammu and Kashmir 181143, India

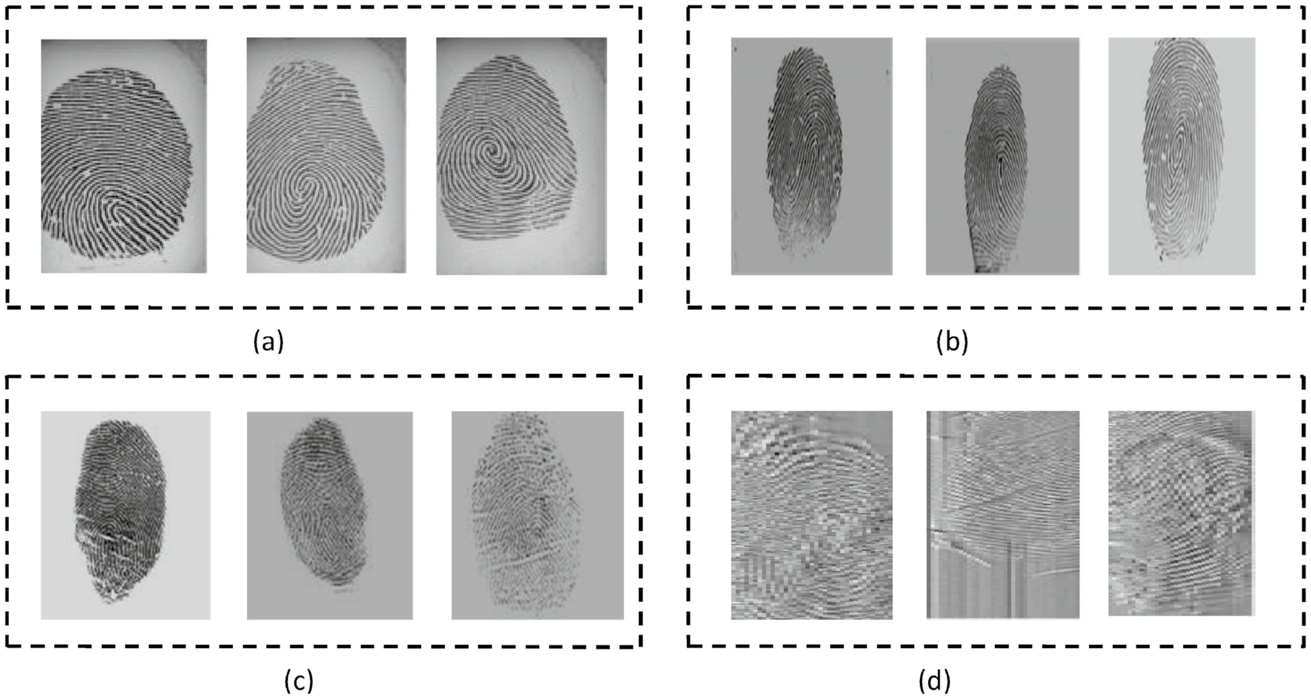


Fig. 1 Fake fingerprint images acquired from: **a** Biometrika sensor using non-cooperative method, **b** ItalData sensor using Non-cooperative method, **c** CrossMatch sensor by using with cooperative method, **d** Swipe sensor by using with cooperative method (adopted from LivDet 2013 DB [5])

check. Broadly, the PAD mechanism may be treated as a binary classification problem that is accomplished through computing the difference of micro-textural or image quality characteristics of live or fake fingerprint traits. Contemporarily, the development of FiPAD mechanisms has become an active field of research that has witnessed numerous contributions to offer effective liveness detection solutions. While the use of single or multiple image features-based PAD techniques has resulted in accurate detection systems, performance and generalization capability to unknown attacks (anomaly detection) [6–10] has been limited. Additionally, one of the hefty tasks of traditional FiPAD techniques is to decide upon the appropriate number and type of image features for classifying the given image as real or fake. The field of deep learning-based PAD is emerging as a potential alternate compared to traditional methods in current time. There are multiple reasons behind it, like automatic deep feature extraction and their improved accuracy. However, the limitations associated with these techniques include an added overhead and larger training dataset requirement. Furthermore, deep learning-based PAD mechanisms necessitate particular hardware infrastructure. Whereas, the traditional classifier-based approaches may also perform well with smaller training datasets with higher discrimination capability of the feature set. The texture-based approaches that utilize single image features such as [8, 10–17] demonstrate limitations to execute uniformly over different fingerprint materials and sensing devices. To counter these problems few multiple feature-based approaches have been reported in past such as WLD + LPQ [14], PHOG + SURF [18], and PHOG + SURF + Gabor [18] which results in comparatively higher ACER (%) of 7.87, 7.32 and 6.90 respectively. Thus, to further improve the performance, our HyFiPAD model relies on extracting three micro-textural textural features from a single fingerprint.

A plentiful of handcrafted feature descriptors is available for the task of image classification problems like LBP, LPQ, LCP, SIFT, SURF, HoG, PHOG, and WLD. The original LBP feature set encodes the information based only on basic relationship among the central pixel with its neighbors. One of the variant of LBP (CeLBP) utilizes global threshold for computing binary code that may result in lower discrimination capability as it extract comparatively lesser significant information from local region. To address this issue, we propose a new variant of local features termed as LABP which calculates the binary codes by making use of dynamically adapting a new threshold. The newly proposed LABP features are combined with existing Complete Local Binary Pattern (CLBP) and BSIF to enhance the overall performance of the HyFiPAD model. Moreover, CLBP features are yet to be explored for FiPAD task that may be a constructive descriptor to minimize the effect of rotational variance in the images. One of the problems in fingerprint images involves unusual characteristics due to several inconsistencies encountered during capturing process. To conquer these problems, the BSIF descriptor is popularly useful for encoding the texture information into a feature set by utilizing filters that are learned from the natural images unlike other descriptors such as LPQ, LCP, SIFT, HoG and many others. The learning using BSIF offers a flexible way to adjust the length of descriptor and also helps to adapt in anti-spoofing task involving unusual image properties like blurring, imperfect alignment, and rotation. In the light of aforementioned challenges, and to further improve the performance, the obvious choice for our novel hybrid PAD model is to exploit the key characteristics of the proposed LABP together with CLBP and BSIF descriptors. Indeed, our experiments that are performed on LivDet 2009 to LivDet 2015 datasets manifest that our selected individual feature descriptors do not perform well to separate fingerprint images into live and fake classes. Hence, all the chosen feature set complements each other to yield excellent classification accuracy.

To the best of our knowledge, the literature affirmed a few hybrid techniques for FiPAD mechanism that are based on ensemble learning (i.e., Random Forest, AdaBoost) using multiple features [18, 14]. With this motivation, our work focus to design a novel hybrid approach that makes use of enhanced image textural features to build an ensemble with traditional SVM classifier and a basic sequential model. A viable solution is to build a majority voting-based ensemble using three classifiers trained on selected feature set.

Therefore, this work broadly aims to develop a generalized hybrid technique that is not impacted by the type of fabrication material used for spoof creation. To attain this, we train and test our model on different dataset (cross-database) and with different sensing technologies (cross-sensor). In summary, the major contributions of this work are listed as follows:

i. The proposed LABP descriptor used in the HyFiPAD technique produces robust feature set with higher discrimination potential.
ii. The complementary combinations of the proposed LABP together with CLBP and BSIF offer superior performance.
iii. An ensemble based on majority voting created with SVM and sequential model yields additional improvements in performance.
iv. The HyFiPAD approach exhibits promising generalization capabilities across unknown attack scenarios evaluated on two publically available anti-spoofing datasets.
v. The experimental results demonstrate the effectiveness of proposed approach in terms of performance as compared to existing state-of-the-art methods.

The remainder of the paper is organized as follows: Sect. 2 presents a review on recent advancements in FiPAD techniques. Section 3 illustrates the framework and algorithms of the proposed HyFiPAD approach. In Sect. 4, the experimental benchmark datasets along with performance protocol and a detailed experimental analysis are systematically discussed. In last, the conclusions as well as the future scope of this work are briefly presented in Sect. 5.

2 Related work

Due to the importance of the counter mechanisms and recent advancements in fingerprint-based recognition systems, it becomes requisite to detect the liveness of the presented trait as intruders can easily counterfeit the authentication system by using various presentation instruments (PAIs). Consequently, the imperative research challenge is to discriminate the live fingerprint from forged one, which has been inspirited by the ISO-Standard IEC 30107-3 E. During the past decades, the fingerprint anti-spoofing techniques have witnessed different trends based on a variety of key concepts as shown in Fig. 2. Usually, FiPAD approaches are separated into two main categories based on the type of liveness indicator, i.e., (a) hardware-based analysis (b) software-based analysis. The former involves an additional sensing device along with the fingerprint recognition system to measure the vitality characteristics such as temperature, pulse-oximetry [19], blood pressure [20], odor [21], skin-impedance [22], and etc. for classifying the live and fake fingerprint traits. In contrary to former, the software module processes the single (static approach) or multiple (dynamic approach) fingerprint images for exploring their image characteristics to detect liveness properties [23, 24]. The handcrafted feature-based methods (static or dynamic) work by extracting the textural features [11, 12, 18, 25–30] or pore-based [31, 32] information from the fingerprint images, whereas dynamic approach utilizes the skin deformation [33, 34] or sweat secretion [35–37] at fingertip as an indicator for FiPAD. Due to its remarkable results in PAD mechanisms, software-based approaches have received greater attention these days. Particularly, texture-based approaches can discover features or patterns from images that could be targeted to examine the difference between two classes of fingerprint traits. Our proposed approach is based on extracted handcrafted micro-textural features from the fingerprint image; therefore, this work purely comes under software-based FiPAD approach. Thus, the brief literature review in this section is limited to pioneer contributions related to software-based mechanisms.

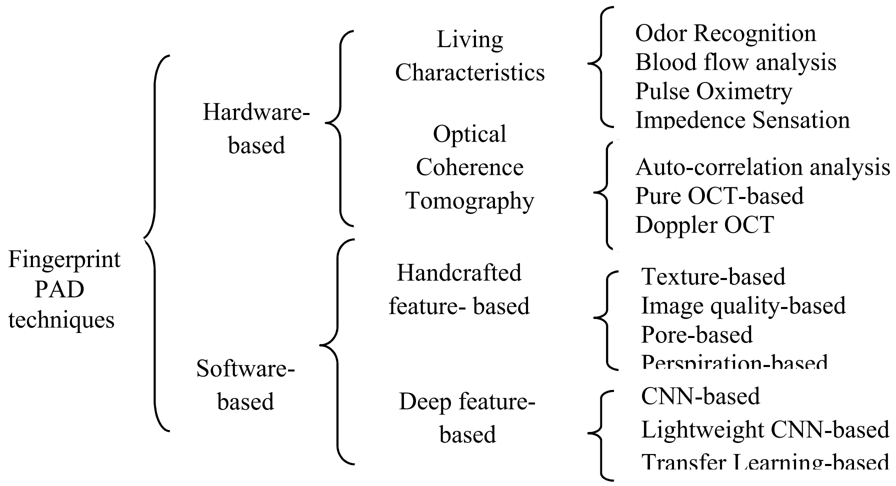


Fig. 2 A classification of fingerprint liveness detection techniques

2.1 Handcrafted feature-based approaches

Most of the conventional handcrafted (manually extracted) feature-based PAD approaches are investigated in the early years of 2000s, where a variety of image descriptors were employed for the task of fingerprint PAD. The handcrafted feature-based descriptors extract texture, color, fingerprint pores, perspiration, and edge information from a given fingerprint images to discriminate between live and fake traits. Early in 2005, the first proposal of texture-based FiPAD is established by Moon et al. [11] and their concept is centered around fingerprint surface analysis by using wavelet feature. The Wavelet analysis facilitates to explore the input signal from images with various scales. Therefore, it is used to minimize the outcome of valley or ridge pattern while surface coarseness estimation is the main basis of fingerprint liveness detection in this method. Nikam et al. [12] introduced wavelet energy features along with LBP histograms. The LBP is used to capture textural information whereas wavelet energy features describe the ridge orientation and frequency information. Zhang et al. [13] also suggested wavelet analysis along with LBP for detecting the liveness of the fingerprint. They developed a scheme where the wavelet analysis is used for obtaining the residual noise and de-noised image. The images are then divided into blocks with equal size and LBP features are extracted from the histograms for FiPAD. Although, LBP is a powerful image descriptor but there is a limitation of small support area for LBP. Therefore, Jia et al. [38] address the limitation of original LBP by employing the multi scale LBP (MSLBP) for fingerprint liveness detection. The MSLBP can be implemented in two ways; the first is to accomplish it by increasing the operator radius and other is to apply filters and then LBP operator for the fixed radius. Ghiani et al. [30] proposed a PAD method that is based on Local Phase Quantization (LPQ). The LPQ is in-variant to rotation and is insensitive to the effects of blurring; hence, it is used in fingerprint liveness detection as it represents all the spectrum characteristics of an image in a compressed feature representation. In another work, Ghiani et al. [39] designed a PAD technique based on BSIF algorithm. BSIF is a local image descriptor which is created by using binary operation on the responses obtained from the linear filters. The filters as oppose to previous binary descriptors are learnt from the natural images by making use of independent component analysis (ICA). The features so obtained are used for fingerprint liveness detection. The authors extend their work in [7] by thoroughly exploring the BSIF features for extracting textural information. Likewise, Li and Chan [40] presented a technique by revising the original BSIF descriptor where 2-Dimensional Gaussian distribution function is applied on the BSIF image. Gragnaniello et al. [14] proposed a method using weber local descriptor (WLD) which is an efficient image descriptor that is inspired by the Weber's law. It contains two main modules, namely differential excitation and orientation. Then, joint histograms generated from both the components are used to build the discriminating FVs. Though the original WLD is a robust local descriptor but two imperfections were observed in it. First, the orientation calculation only considers the pixels in horizontal and vertical directions, which does not adequately reflect the gradient orientations. Next, the differential excitation component of original WLD accumulates the sum of the differences between the center pixel and its neighbors. The positive and negative differences will counteract each other, which degrades the effectiveness of descriptor. Aforementioned issues are addressed with the development of a novel Weber local binary descriptor (WLBD) by Xia et al. [10]. To design an efficient descriptor, the original LBP is combined with Weber's law for constructing the local binary differential excitation (LBDE). Here, a function used to obtain the uniform LBD that describes the change in amplitude with respect to neighborhood pixels. The local binary gradient orientation (LBGO) is defined to characterize the orientation related information. Hence, WLBD is an amalgamation of

LBDE and LBGO to generate discriminating features from images. Dubey et al. [18] introduced a hybrid technique that use low level gradient features from SURF, PHOG as well as texture features are extracted from Gabor wavelet, which are then coalesced by using dynamic score level integration. The framework is divided into three main blocks such as image pre-processing, feature extraction, and a classification stage. In image pre- processing stage, the quality of images is enhanced by first applying the crop operation and then increasing the perception information with histogram equalization technique. In the next step, the images with different scales and rotations are captured from multiple subjects and with various scanners. The SURF is invariant to illumination, scale and rotation, on the other hand, PHOG is used to extract shape information for achieving extra discriminating features. Xia et al. [6] extracted features from image gradients and generate a co-occurrence array. They applied quantization operation on each image and calculate their horizontal and vertical gradients. Similarly, second and third order array is computed by truncating the gradients. The extracted features are then used for image classification. Kim et al. [15] came out with a new PAD model that is based on Local Coherence Pattern (LCP), where the difference of the dispersion in the image gradient field between the live and fake fingerprints is focused. The coherence is computed from the fingerprint image. Then LCP histogram is used to create a FV for training the PAD model training and classification. In analogues to prior research, Jiang and Liu [8] make use of Uniform LBP (ULBP) in spatial pyramid and these features are used to train an SVM classifier to discriminate live and fake fingerprint traits. In a recent work, Agarwal et al. [41] employed the notion of ensemble learning by adopting stacking and bagging method. The dataset is portioned into various subsets and LBP features are then extracted from each subset to train an SVM classifier. Recently, Gonzalez-Soler et al. [42] proposed an integrated approach in which local dense-SIFT descriptor is combined with three feature encoding techniques, namely Fisher Vector (FV), Vector of Locally Aggregated Descriptors (VLAD) and Bag of Words (BoW). The encoded features are used to train an SVM classifier. After training, the experiments are performed on LivDet 2011 to LivDet 2019, where the FV showed the best performance in realistic and more complex scenarios.

2.2 Deep learning-based approaches

With the emanation of deep convolution neural networks (DCNNs) and their outspread applications in pattern recognition, deep learning-based PAD techniques have attained a momentous growth in recent years [43]. With this succession in deep learning, several attempts have been made by research community to apply deep feature-based extraction for fingerprint liveness detection.

Initially, in 2014 Nogueira and De Alencar [44] applied DCNN model for the first time in fingerprint PAD mechanisms. They achieved the tasks of feature extraction and classification in two different stages, which further makes the system not to be optimized simultaneously. Then, Wang et al. [45] removed this limitation by introducing a DCNN-based model named Finger-Net in which Softmax layer is applied as classifier which optimizes the feature extraction and classifier training task simultaneously by using back-propagation pass. Nogueira et al. [46] elongated their model by incorporating the transfer learning concept, where they employed a pre-trained DCNN model named AlexNet [47]. The network is trained by VGG model as it increased the classification accuracy further by 2%. The authors compared four distinctive models in which (two are CNNs pre-trained on natural and fine-tuned with fingerprint images, a CNN model with random weights, and a conventional LBP approach) pre-trained CNNs can yield the state-of-the-art results. Park et al. [48] and Toosi et al. [49] designed two different techniques where fingerprint images are first segmented to perform data augmentation and then voting strategy is applied to determine the liveness of fingerprint images. Lazimul and Binoy [50] trained a ConvNet architecture, and employed this model for fake and live class prediction. The discriminating features are extracted from pre-trained network, i.e., conception of transfer learning is adopted. The authors also applied image enhancement techniques in order to augment the accuracy of the overall system. Jung and Heo [51] proposed CNN architecture that makes use of ‘squared regression error (SRE)’ instead of fully connected layer. The SRE layer sets a certain threshold value which restraint the suitable level of false positive or negative rates. Souza et al. [52] proposed a DBM model to extract deep features from the fingerprint image, and these extracted features are fed to SVM classifier for discriminating live and fake fingerprint images. Zhang et al. [53] developed a new lightweight framework known as SlimResCNN. The network is called SlimResCNN as the adaptation of neural network structure on smaller datasets broadened the convolutional filters to twice and compressed the depths to an extent in contrast to original network structure. Their network consists of stacked residual blocks where the local patches are segmented from foreground region of fingerprint image using ‘center of gravity’ technique. After extraction the patches are fed to the model to overcome the problem of information loss due to up or down sampling of image size. The proposed model outperforms the state-of-the-art techniques and is also a winner of LivDet2017 competition. Yuan et al. [54] designed an improved DCNN model which is based on ‘Learning Rate Adaptive Adjustment’ process. The main goal of the proposed technique is to model a structure that is not limited to the fingerprint image scale; therefore a new layer known as Image Scale Equalization is added in the middle of the fully concatenation layer and the last convolution layer. Uliyan et al. [55] proposed a novel DCNN model for fingerprint liveness detection on the basis of Deep Restricted Boltzmann Machine (DRBM) and Deep Boltzmann Machine (DBM). This multilayer architecture deals with complex textural patterns in an efficient manner. In the proposed method, after training a DBM, such structure has employed to extract deep level features from the grayscale fingerprint images. The KNN classifier is trained with the feature vectors, extracted by the DBM, to examine spoof forgeries. Another lightweight and efficient CNN architecture, named FLDNet, is given by Zhang et al. [56] which overcomes the limitations of Global Average Pooling (GAP) in fingerprint liveness detection. The FLDNet consists of altered dense block structure where residual path is incorporated and this network effectively upgrades the detection accuracy. In another work, Zhang et al. [57] fused the task of fingerprint matching and spoof detection. The authors calculated the similarity between two fingerprints on the basis of Octantal Nearest-Neighborhood Structure (ONNS), where the minutia point closest to the central minutia is taken from each sector of an octant. Then, the liveness detection score from the fingerprint image is attained by employing modified Slim-ResCNN model. Finally, a score-level fusion is implemented on fingerprint matching and liveness detection results by generating score feature vector. Then, logistic regression is applied on the obtained score feature vector for discriminating live and fake fingerprint images.

Please refer ‘Appendix,’ where table illustrates the existing software-based FiPAD techniques with their key concept and performance. The summary points out that majority of the single descriptor-based FiPAD methods exhibits limited performance, while multiple feature-based methods offer improved results. Though, deep feature-based methods have been explored in state-of-the-art techniques, but there are several challenges that obstruct their effectiveness. Hence, this encourages us to develop a texture-based FiPAD models that yields enhanced performance with appropriate selection of robust discriminating features, learning approaches, and the integration of handcrafted features trained classifiers to create an ensemble.

3 The proposed approach

The discriminative micro-textural features are advantageous for capturing similarity among adjacent pixel values to safeguard against spoof attacks. To achieve superior detection results, it is essential to train an appropriate PAD model that is based on significant features extracted from a sufficient number of relevant images. To address the specific concerns related to existing state-of-the-art methods such as low discriminative power of single descriptor and use of single classifier, we follow to present a hybrid PAD technique that substantially improves the overall performance. To obtain detailed textural features form images, use of dynamic threshold for local regions is a better solution. The rotational variance in fingerprint images may be overcome by exploiting CLBP features. Apart from this, to address the issue of unusual characteristics introduced while capturing the images, there is a need to utilize a robust descriptor such as BSIF. In the following subsections, we describe the underlying idea of HyFiPAD framework, feature sets, algorithms, and learning algorithms together with the ensemble approach for fusing the outcome of multiple classifiers.

3.1 The HyFiPAD framework

The potential of local patterns combined with the modified adaptive features may result in constructing more robust FVs that may significantly enhance the accuracy. To this end, we present a hybrid framework for FiPAD to accomplish the task of the anti-spoofing sub-module as illustrated in Fig. 3. The key concept of the proposed framework is to function in two phases that comprise of training and testing procedures. The training phase broadly covers a series of activities that are applied to fingerprint dataset such as pre-processing, FV extraction, building a ensemble on base estimators, and tuning parameters of the model. The goal of the testing phase is to validate the HyFiPAD model on a randomly selected set of images covering different range of sensors and datasets for evaluating its accuracy. The detailed explanation of these stages is discussed in the following sections.

3.1.1 Image pre-processing

The motive of image pre-processing is to enrich the quality of fingerprint images. The acquired fingerprint images are usually of low quality as they are captured under different environmental conditions (i.e., dry or wet fingers, poor illumination effect, and moisture in the air) through a variety of sensing devices. To prepare and enhance these images for handcrafted features-based model, the dataset is subjected to a series of image processing operations.

To eliminate the unnecessary background information, firstly the region of interest (ROI) is segmented from the fingerprint images. Later on, we transform the colored images to gray scale for reducing the computational complexity. Indeed, in our approach, the color feature is not necessary to distinguish between the class of fake and live modalities as it contains more information that append unnecessary complexity and takes more space in memory. To attain uniformity, the next phase resizes the fingerprint images to a dimension of 312 × 372. Further, to improve quality, we

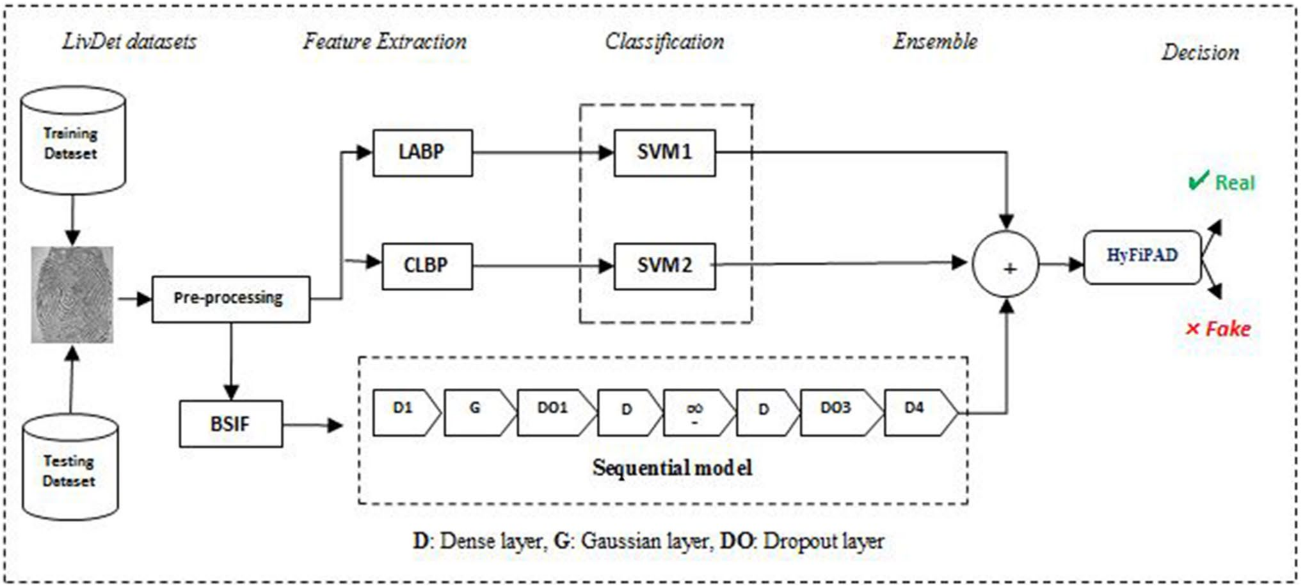


Fig. 3 Framework of the proposed HyFiPAD technique

apply the Histogram Equalization (HE) technique, which is used to stretch the intensity of input images to create normal distribution so that the active range of the image is fully normalized.

3.1.2 Feature extraction

We present a brief description of image features that are used for designing the HyFiPAD. The PAD architectures that are built on the LBP feature descriptor have achieved substantial performance, particularly for fingerprint anti-spoofing models. In the beginning, Ojala et al. [58] introduced the LBP descriptor to extract texture features as a binary string from a local region preferably in the $N \times N$ neighborhood by comparing the intensity value of the central pixel with its neighbors. The LBP code of a pixel is computed by using Eqs. (1) and (2). The feature vector (FV) is constructed by consolidating all the patterns for a given image.

$$\mathrm{LBP}_{N,R} = \sum_{h=0}^{h-1} f(p_c - p_h)2^h \tag{1}$$

$$f(d) = \begin{cases} 1, & \text{if} d \geq 0 \\ 0, & \text{otherwise if} d < 0 \end{cases} \tag{2}$$

where h indicates the number of the neighborhood of a central pixel c and N represent the total number of pixels in the given region, i.e., 8, or 24. The function f computes the difference between the intensity levels of both pixels.

It is clear that the original LBP fails to discriminate the regions within the smaller or larger variations in the intensity levels which usually results in the erroneous computation of LBP codes. Moreover, the small change in the intensity values due to the presence of noise may lead to inaccurate computation of these codes. In another variant LBP as CLBP, the threshold for computing the binary code is calculated from the intensity levels of the whole image which result in capturing less significant information from a local image region. Therefore, to lessen these effects, we put forward a new image descriptor based on adaptive threshold and it is then combined with CLBP and as well as the BSIF feature descriptors. The fundamental approach which is adopted to extract micro-textural image features along with their pseudo-codes is described in following subsections.

3.1.2.1 Proposed local adaptive binary pattern (LABP) To overcome the issue of intensity variations in existing versions of LBP, we design a modified descriptor (LABP) that utilize an adaptive or dynamic threshold for each central pixel during the process of computation of the binary codes. The adaptive threshold ensures to gather refined information of the surrounding pixels to crop the micro-level details during feature extraction. The LABP code for a given image may be computed by using Eqs. (3) and (4).

$$\mathrm{LABP}_{N,R} = \sum_{h=0}^{n-1} f(p_c - p_h)2^h \tag{3}$$

$$f(d) = \begin{cases} 1, & \text{if} d \geq DT \\ 0, & \text{otherwise if} d < 0 \end{cases} \tag{4}$$

where DT is the dynamic threshold that is computed by averaging the intensity level of all the neighboring pixels of central pixel p_c. Therefore, LABP code for a given image is more discriminative and free from local variations in a specific region of the image. The dynamic threshold is calculated for each separate region by using the expression as given in Eq. (5).

$$DT = \frac{\sum_{h=0}^{n-1} Ph}{n} \tag{5}$$

The adaptive threshold for computing the LABP code extracts more specific characteristics from a particular region of an image. It generates a feature descriptor that holds added discrimination power as compared to the original LBP or CLBP. Hence, our proposed LABP features make use of adaptive threshold throughout the whole process in comparison with CLBP, where a single threshold value is used. Figure 4 displays the process of computing the LABP code for a given image. The algorithmic steps for extracting the LABP features from a fingerprint image I(x, y) are shown in Fig. 5.

For an input image I(x, y), a distinct adaptive threshold is computed for a local given region that is used to compute the LABP code for all the pixels. The adaptive or dynamic threshold of a central pixel is the average of intensity levels of all its neighboring pixels. Finally, the normalized histogram 'h' of this LABP image symbolizes the feature descriptor of a given image I(x, y). In this manner, the micro-textural LABP features are extracted from the fingerprint image. The features obtained for a sample fingerprint image using LABP descriptor are illustrated in Fig. 6.

3.1.2.2 Complete Local Binary Pattern (CLBP) To cover specific textural properties of an image, Guo et al. (2010) [59] proposed an extended version of the original LBP. The descriptor compute three different LBP codes from a given region of the image; first by considering the sign bits of the

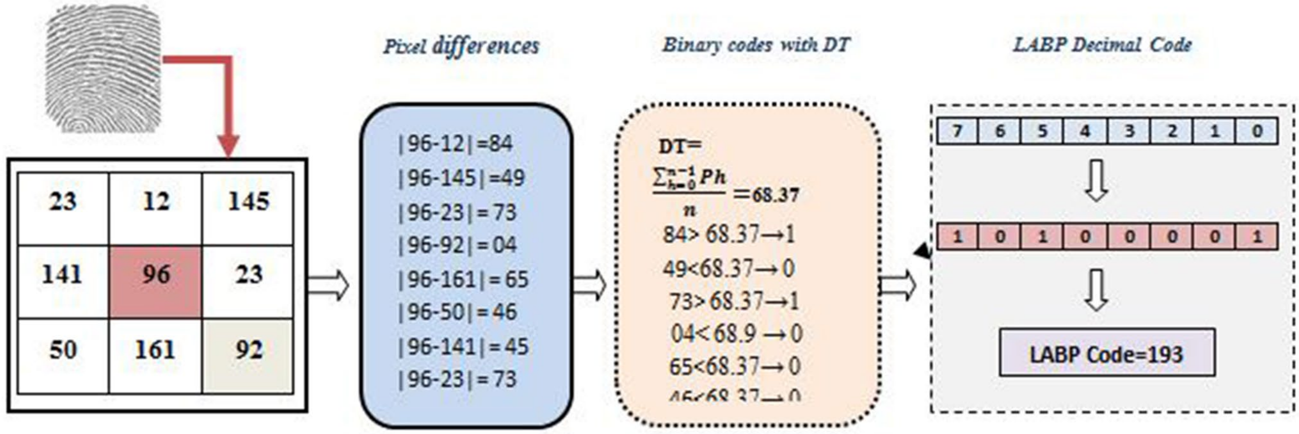


Fig. 4 An example of LABP code computation for a given image

Fig. 5 An algorithm to extract LABP features from an image *I*(*x*, *y*)

Algorithm 1: **LABP feature extraction from fingerprint images.**

```
Input :  Image I(x, y)
Output: LABP feature descriptor
Begin
      Let M,N=size(I(x, y)
  for i=1 to M
      for j=1 to N
          $x_c, y_c$=I(i, j)
          LABP(i, j) $_{N, R} = \sum_{□=0}^{n-1} f(p_c - p_h)\ 2^h$
          Compute DT=$\frac{\sum_{□=0}^{n-1} P□}{n}$
          $f(d) = \begin{cases} 1, & if\ d \geq DT \\ 0, & ot□erwise\ if\ d < 0 \end{cases}$
      end
  end
      h = Hist(LABP(i, j))
      h = normalise(h)
   Return h
end
```

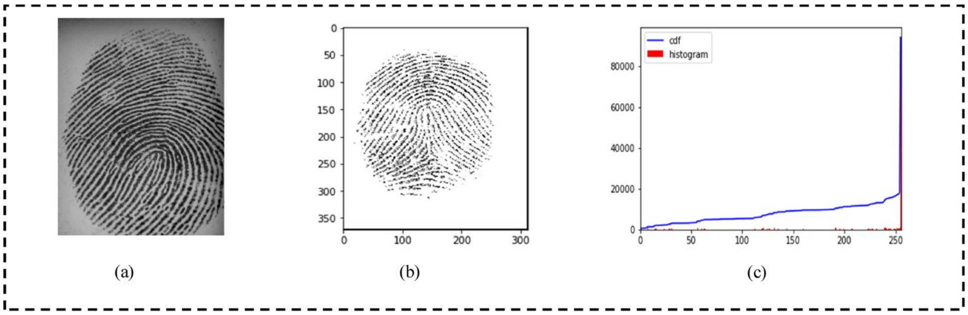


Fig. 6 An instance of image feature with LABP descriptor **a** original fingerprint image, **b** LABP image, **c** LABP histogram

difference of central pixel and neighboring pixels; second, the magnitude of the difference is compared with a predecided threshold, and third by comparing the central pixel with the average of intensity level of all the pixels of an image. The sign LBP codes are computed by using Eqs. (6) and (7), and these are similar to the original version of LBP.

$$\mathrm{SLBP}_{N,R}\left(x_c, y_c\right) = \sum_{h=0}^{h-1} f(p_c - p_h)2^h \tag{6}$$

$$f(d) = \begin{cases} 1, & \text{if} d \text{ is} + \text{ve} \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

The magnitude LBP is computed by using Eqs. (8) and (9),

$$\mathrm{MLBP}_{N,R}\left(x_c, y_c\right) = \sum_{h=0}^{h-1} f(p_c - p_h)2^h \tag{8}$$

$$f(d) = \begin{cases} 1, & \text{if} d \geq T \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

where T denotes the threshold which is computed by taking the average of the difference between central and all the pixels of the image, i.e., $T = \text{mean}|p_c - p_i|$ where $i = 1, 2, 3, \ldots, N^2$ with an image of size $N \times N$. Finally, the central LBP code is computed by using Eqs. (10) and (11).

$$\mathrm{CELBP}_{N,R}\left(x_c, y_c\right) = \sum_{h=0}^{h-1} f(p_c - p_h)2^h \tag{10}$$

$$f(p_c), f(d) = \begin{cases} 1, & \text{if} d \geq T \\ 0, & \text{otherwise} \end{cases} \tag{11}$$

where T is the average intensity of the whole image with size as $N \times N$.

The aforementioned steps to compute CLBP image features are summarized in an algorithm shown in Fig. 7. Given an image $I(x, y)$, of size $M \times N$, we first compute SLBP, MLBP, and CeLBP codes for each pixel (i, j). The CLBP of the image is obtained by consolidating these variants of LBP. Finally, the descriptor is constructed by using a normalized histogram of the CLBP image.

The visual and quantitative analysis of the CLBP features for a sample fingerprint image is depicted in Fig. 8. The corresponding discriminative information of the resultant CLBP features is clearly visible in Fig. 8d.

Fig. 7 An algorithm to extract CLBP features from an image $I(x, y)$

Algorithm 2: **To extract CLBP features from a fingerprint image.**

```
Input :  Image I(x, y)
Output: CLBP feature descriptor
Begin
    Let M,N=size(I(x, y)

    for i=1 to M
        for j=1 to N
            x_c,y_c=I(i, j)
            SLBP_{N, R}(x_c, y_c)    // compute
            MLBP_{N, R}(x_c, y_c)    // compute
            CELBP_{N, R}(x_c, y_c)   // compute
            CLBP(i, j) = merge(SLBP_{N, R}(x_c, y_c), MLBP_{N, R}(x_c, y_c), CELBP_{N, R}(x_c, y_c) )
    end
        end
        h =  hist (CLBP (i, j))
        h = normalise(h)
        Return h
end
```

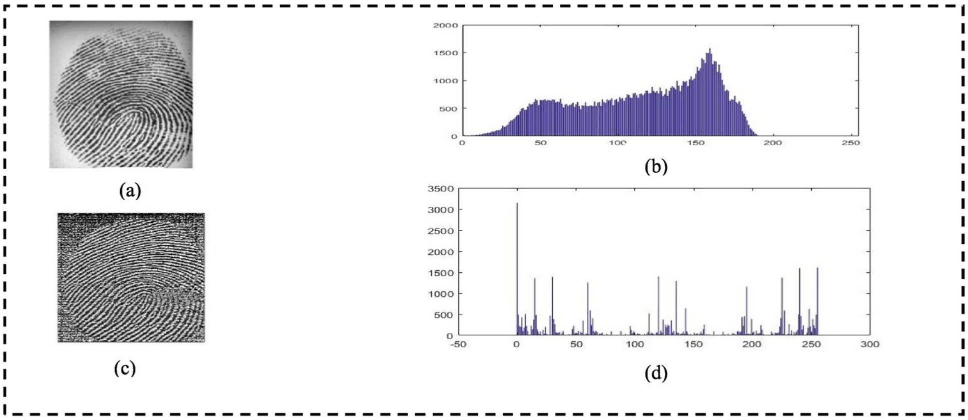


Fig. 8 An instance of image features with CLBP descriptor. **a** Original image, **b** histogram of original image, **c** CLBP image, **d** histogram of CLBP features

3.1.2.3 Binarized statistical image features (BSIF) Our approach make use of a third textural feature as introduced by Kannala et al. [60]. In this case, a binary string is computed for all the pixels of a given image. The code value of a pixel may be considered as a local descriptor of the image intensity patterns in the pixel's surroundings. Given an image segment $P(u, v)$ of size $q \times t$ pixels, and assume a linear filter X_i of the same size, the response of the filter r_i is computed by using Eq. (12).

$$r_i = \sum_{u,v} X_i(u,v)P(u,v) = x_i^T p \tag{12}$$

where the vectors x and p denotes the pixels of X_i and P. The binarized feature $\mathrm{b_i}$ is obtained by using Eq. (13).

$$b_i = \begin{cases} 1, & r_i > 0 \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

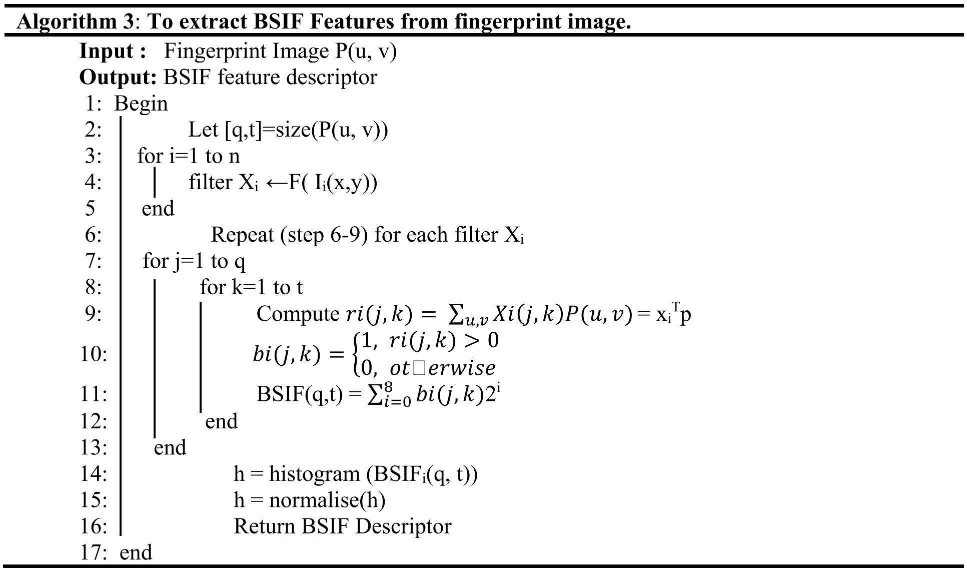
For n linear filters, different convolutions results in a set of features that may be stacked in W of size $n \times q \times t$. The set of filters are derived by using a natural set of images where statistical independence of r_i is maximized. The feature vectors generated in this manner are used to construct a set of histograms to represent the feature descriptors of the images. Let there are n number of natural images denoted by $I_i(x, y)$, where $i = 1,2,3,\ldots, 8$. Suppose that, X_i be the filter of size 11×11 pre-learned from ith image. The ith filter is convolved with the input image $P(u, v)$ to compute the response $r_i(u, v)$. Further $b_i(u, v)$ is the binary response of the ith filter at pixel (u, v). In this way, the binary response of all the filters is computed at the pixels of the whole fingerprint image. These responses are used to generate a code for each pixel to construct a feature image. In the last, the normalized histogram of the BSIF image provides the resultant feature descriptor (Fig. 9).

An example of features extracted from the algorithm as applied on a sample fingerprint image with a set of filters derived from natural images is exhibited in Fig. 10.

3.1.3 Classification stage

Once the image features are extracted by using the selected descriptors, the next task involves the training of a classifier or learner to discriminate an image as fake or live. A classifier is typically an algorithm that maps a given set of input features to predefined class labels. Several classifiers are available such as SVM, Decision Tree (DT), Naive Bayes, Random Forest, and etc. that works on the principle of supervised learning using labeled datasets. However, the modern techniques involve automatic feature extraction with multiple hidden layers followed by a fully connected classification layer, and these networks are known as DCNNs. Our approach employs two SVM classifiers and a generic sequential model as base estimator to accomplish the task of classification, that are all integrated through an ensemble. We briefly discuss the underlying principle of these classifiers in the following subsections.

3.1.3.1 Support vector machine (SVM) The HyFiPAD makes use of linear SVM to train two classifiers based on CLBP and LABP feature descriptors. Vapnik et al. [61] introduced the SVM as a part of his Doctoral thesis from the Moscow University. The SVM constructs a hyper-plane as a decision boundary in such a manner that the margin of separation between two class items is maximized. Typically, SVM algorithms construct learning model based on three functions, namely: polynomial, radial basis function (RBF), and a two-layer perceptron. The SVMs have good generalization performance and also called as the max-margin classifier. Usually, a linear classifier solves simple linearly separable problems of binary classification. For non-linearly separable problems the kernel trick is used for projecting the data into higher dimensional space to make it linearly separable. In our HyFiPAD, for training two different classifiers on CLBP and LABP descriptors, we use linear SVM with $c = 100$ and function = 'RBF.'



Algorithm 3: **To extract BSIF Features from fingerprint image.**

```
Input :  Fingerprint Image P(u, v)
Output: BSIF feature descriptor
 1: Begin
 2:         Let [q,t]=size(P(u, v))
 3:     for i=1 to n
 4:         filter $X_i$ ←F( $I_i$(x,y))
 5      end
 6:             Repeat (step 6-9) for each filter $X_i$
 7:     for j=1 to q
 8:         for k=1 to t
 9:             Compute $ri(j,k) = \sum_{u,v} Xi(j,k)P(u,v)$ = $x_i^T p$
10:             $bi(j,k) = \begin{cases}1, & ri(j,k) > 0\\ 0, & ot\square erwise\end{cases}$
11:             BSIF(q,t) = $\sum_{i=0}^{8} bi(j,k)2^i$
12:         end
13:     end
14:             h = histogram ($BSIF_i$(q, t))
15:             h = normalise(h)
16:             Return BSIF Descriptor
17: end
```

Fig. 9 An algorithm to extract BSIF features from an image *I*(*x*, *y*)

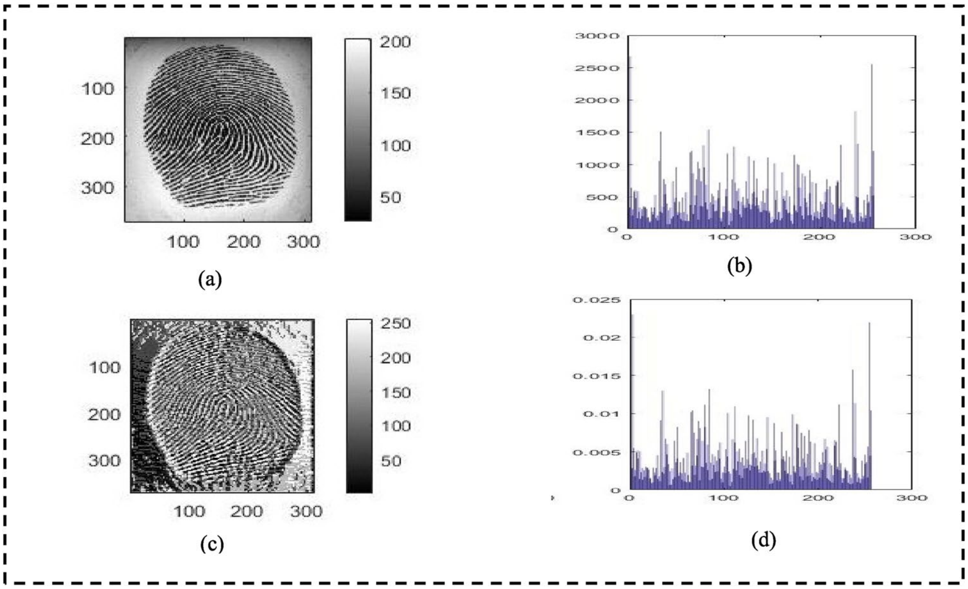


Fig. 10 An instance of image feature with BSIF descriptor **a** original image, **b** histogram of original image, **c** BSIF image, **d** BSIF histogram

3.1.3.2 Sequential model For the BSIF features in our method, we make use of the sequential model in Keras via sequential API. These are the simplest neural networks that allow building the model layer by layer. In these cases, each layer has a weight that corresponds to the next layer in the sequence. We use 'add ()' function to add new layers in the model. The basic sequential model used in our approach is summarized in Table 1. The dense layer is a regular layer of neurons in the neural network where each neuron gets inputs from all the neurons in the previous layer, thus densely connected. The purpose of the dense layer is to change the dimensionality of the data. We use the Rectified Linear Activation Unit (ReLu) in a dense layer for its activation. In this model, we use the Gaussian Noise layer for regulariza-

Table 1 A description of basic sequential model for learning BSIF in HyFiPAD

Sr. no.	Layer (Type)	Out shape	Arguments
1	Dense_1 (dense)	(None, 258)	activation = 'relu'
2	Gaussian_noise_1	(None, 258)	stddev = 3.5
3	Dropout_1 (dropout)	(None,258)	Rate = 0.3
4	Dense_2	(None,258)	activation = 'Relu'
5	Dropout_2 (dropout)	(None,258)	Rate = 0.3
6	Dense_3 (dense)	(None,258)	activation = 'Relu'
7	Dropout_3 (dropout)	(None, 258)	Rate = 0.35
8	Dense_4 (dense)	(None, 1)	activation = 'sigmoid'

tion purpose that is active during the training phase. The additive Gaussian layer in this case is used at stddev = 3.5 and the layer has the same output as input. The dropout layer is used to prevent the over fitting and it randomly sets the input to '0' with a frequency rate of training time. The input which is not set to 0 is incremented by 1/(1 − rate) so that overall inputs are unchanged. In our approach, we use the rate = 0.35 or 0.3 in the dropout layer.

3.1.4 Ensemble using majority voting

In general, a voting ensemble or a '*majority voting ensemble*' is a machine learning model that integrates the predictions from multiple classifications or regression learners. We employ a simple majority voting method in HyFiPAD to achieve fused outcomes of the three classifiers, i.e., SVM1, SVM2, and Sequential model. The voting scheme that predict class labels is called hard voting whereas soft voting predicts class membership probabilities. We use hard voting in our approach to predict the outcome as fake or live. The key lines of python code that are used in our approach are shown below.

```
models = list()
Models = [('model1',SVC( )),('model2',SVC( )), ('model3', Sequential())]
ensemble = VotingClassifier (estimators ='Models' , voting='hard')
```

3.2 The HyFiPAD learning and validation

The learning of our PAD approach by using a training algorithm is depicted in Fig. 11. Initially, a training set of fingerprint images of size '*t*' is chosen from the benchmark anti-spoofing dataset (D_t). For the better performance of the model, the training dataset should be of sufficient size and also cover samples from all the labeled classes and as well as sensors. Thereafter, the training images are pre-processed by applying few basic image processing operations to obtain a standardized set of images. Let $T_i(x, y)$ is the ith image of D_t obtained after applying pre-processing operations on corresponding input image $I_i(x, y)$. In the next step, we apply three image descriptors namely CLBP, LABP, and BSIF on $T_i(x, y)$ to extract respective micro-textural features as $FD_1{}^i$, $FD_2{}^i$, and $FD_3{}^i$ respectively. In this manner, three feature

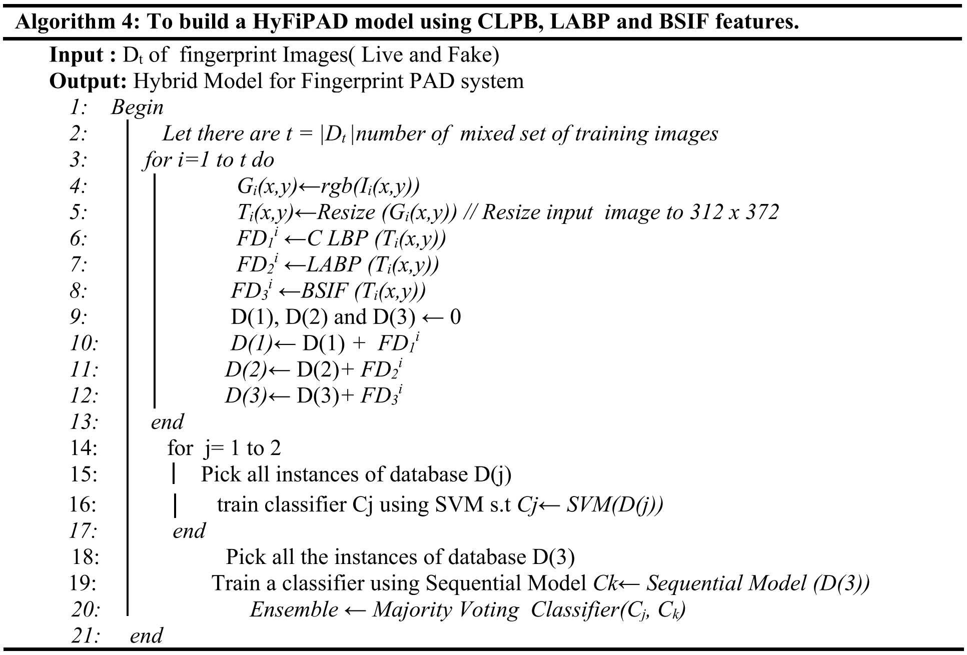


Fig. 11 A training algorithm for HyFiPAD

Algorithm 4: To build a HyFiPAD model using CLPB, LABP and BSIF features.

Input : D_t of fingerprint Images(Live and Fake)
Output: Hybrid Model for Fingerprint PAD system

```
Begin
    Let there are $t = |D_t|$ number of mixed set of training images
    for i=1 to t do
        $G_i(x,y) \leftarrow rgb(I_i(x,y))$
        $T_i(x,y) \leftarrow Resize\ (G_i(x,y))$ // Resize input image to 312 x 372
        $FD_1{}^i \leftarrow C\ LBP\ (T_i(x,y))$
        $FD_2{}^i \leftarrow LABP\ (T_i(x,y))$
        $FD_3{}^i \leftarrow BSIF\ (T_i(x,y))$
        D(1), D(2) and D(3) ← 0
        $D(1) \leftarrow D(1) + FD_1{}^i$
        $D(2) \leftarrow D(2) + FD_2{}^i$
        $D(3) \leftarrow D(3) + FD_3{}^i$
    end
    for j= 1 to 2
        Pick all instances of database D(j)
        train classifier Cj using SVM s.t $Cj \leftarrow SVM(D(j))$
    end
    Pick all the instances of database D(3)
    Train a classifier using Sequential Model $Ck \leftarrow Sequential\ Model\ (D(3))$
    $Ensemble \leftarrow Majority\ Voting\ \ Classifier(C_j, C_k)$
end
```

vectors, D_1, D_2, and D_3 are generated from D_t. The label of ith image is concatenated with the corresponding value of the FV to build a dataset (D_j). As a result, three feature sets are built, i.e., D_j ($j=1, 2, 3$) from D_t ($i=1, 2, 3,\ldots, t$).

Thereafter, D_1 and D_2 feature sets are used to learn SVM to build two respective classifiers. Further, the D_3 that hold handcrafted BSIF features is used to train a basic sequential model comprising of eight layers. Afterward, a hard majority voting ensemble is created by using three base estimator classifiers. Finally, the trained model is fine-tuned to various parameters with proper experimentation at an appropriate search space.

The steps for testing algorithm of the HyFiPAD method are listed in Fig. 12. The trained HyFiPAD model is validated by presenting images from the testing dataset. A similar sequence of steps such as image pre-processing and feature extraction from three descriptors is applied on test images to generate the corresponding feature sets. Finally, the classification of the test samples is carried out by HyFiPAD by assign a class label as either live or fake.

4 Experimental results and discussions

In this section, we evaluate our technique for effectiveness of liveness detection mechanism. Firstly, we present a concise introduction of datasets and the evaluation protocols which are being used as a standard criterion for measuring the performance. Afterward, the HyFiPAD is fine-tuned and the resultant model is evaluated on the four publically available datasets namely; LivDet 2009, LivDet 2011, LivDet 2013, and LivDet 2015. For measuring the generalization capability of the technique, it is also tested with cross-data and cross-sensor scenarios. An ablation study is also carried out to evaluate the performance of the model under excisions of various components. Finally, the performance of the HyFiPAD is compared against the related state-of-the-art PAD methods.

4.1 Experimental datasets

For evaluation purposes four benchmark fingerprint anti-spoofing datasets, i.e., LivDet 2009 [62], LivDet 2011 [63], LivDet 2013 [5] and LivDet 2015 [64], are used. The detail of these datasets is summarized in Table 2. The LivDet 2009 is comprised of 11,000 images acquired from 464 subjects using three different optical sensors. On the other hand, the LivDet 2011 dataset contains 8000 images acquired from 712 subjects using four different optical sensing devices. The 'with cooperation' method of fake fingerprint creation is used for creating the majority of the spoof images as it results in spoof images of better quality. The LivDet 2013 consists of images captured from four different sensors namely Swipe, ItalData, CrossMatch and Biometrika. Besides, LivDet 2013 uses first time the non-cooperation method for creating spoof images from material such as latex, gelatine, body glue, ecoflex, wood glue, modasil and play-doh. Additionally, the LivDet 2015 uses Biometrika, Digital Persona, GreenBit and CrossMatch sensors for image acquisition. The whole database in all the datasets is divided into two components namely training and testing sets. The classifiers are trained by using the training dataset whereas the testing dataset is used for validating the resultant classifiers.

4.2 Performance protocols

For performance evaluation, we select the overall protocol related to metrics and appropriate dataset selection. We utilize the training images from all the datasets to learn the HyFiPAD and testing samples are selected across various domains to compute the performance. The technique is

Fig. 12 Testing algorithm for HyFiPAD

```
Algorithm 5: To validate the HyFiPAD model
Input: Test fingerprint image as Q(x,y)
Output: Class Label (Fake or Live)
Begin
    $G_i(x,y) \leftarrow rgb(Q_i(x,y))$
    $G_i(x,y) \leftarrow Resize$ // Resize input  image to 312 x 372
    $Q_{D1} \leftarrow CLBP\ (Q'(x, y))$
    $Q_{D2} \leftarrow LABP\ (Q'(x,y))$
    $Q_{D3} \leftarrow BSIF\ (Q'(x,y))$
    for  j= 1 to 2
        Test $Q_{D(j)}$ on $Cj$
        $O_{c(j)} \leftarrow Cj(Q_{D(j)})$
    end
    Test $Q_{D(3)}$ on $C_k$ s.t- $O_{c(k)} \leftarrow Ck(Q_{D(4)})$
    Apply major voting for  j= 1 to 2 $O_{C(j)}$ and $O_{Ck}$
    Class { Live, Fake} ← Voting($Q_{D(j)}$ and $Q_{D(4)}$)
end
```

Table 2 An outline of LivDet 2009 to LivDet 2015 datasets

Dataset	Technology	Type of sensor	Resolution (dpi)	Image size	Live	Spoof	Subjects
LivDet 2009	Optical	CrossMatch	500	480×640	2000	2000	254
	Optical	Biometrika	569	312×372	2000	2000	50
	Optical	Identix	686	720×720	1500	1500	160
LivDet 2011	Optical	Biometrika	500	312×372	1000	1000	200
	Optical	Digital Persona	500	355×391	1000	1000	200
	Optical	ItalData	500	640×480	1000	1000	200
	Optical	Sagem	500	352×384	1000	1000	112
LivDet 2013	Optical	CrossMatch	500	800×750	2000	2000	225
	Optical	Biometrika	569	315×352	2000	2000	100
	Optical	ItalData	500	640×480	2000	2000	100
	Thermal	Swipe	96	208×1500	2000	2000	225
LivDet 2015	Optical	Green Bit	500	500×500	1000	1500	100
	Optical	Digital Persona	500	252×324	1000	1500	100
	Optical	Biometrika	1000	1000×1000	1000	1500	100
	Optical	CrossMatch	500	640×480	1500	1448	100

evaluated in terms of three standard performance metrics, namely average classification error rate (ACER), average classification accuracy (ACA) and receiver operating characteristic (ROC).

4.2.1 ACER

The average error is resulted due to misclassification while testing a PAD method by presenting a set of sample images is expressed as ACER. The ACER performance metric is computed by using Eq. (14).

$$\mathrm{ACER} = \frac{\mathrm{FPR} + \mathrm{FNR}}{2} \tag{14}$$

where FPR stands for 'false positive rate' and it measures the percentage of live fingerprints which are actually misclassified. Whereas, FNR stands for 'false negative rate' and it measures the percentage of misclassified fake fingerprints.

4.2.2 ACA

It represents average number of instances correctly classified by the PAD algorithm when presented with the set of fingerprint samples. The ACA is computed by using Eq. (15).

$$\mathrm{ACA} = \left(\frac{1}{N_{\mathrm{f}}} \sum_{i=1}^{N_{\mathrm{f}}} N_{\mathrm{c}} \right) \times 100 \tag{15}$$

where N_{f} represents the total number of live and fake fingerprint images and N_{c} represents the total number of images which are correctly classified as live or fake.

4.2.3 ROC

An ROC curve represents the variation of true positive rate with false positive rate to evaluate the output quality of the classifier. Particularly, ROC curves are useful for the binary classification problems to study the outcome of the classifiers. Further, the area under curve (AUC) denotes the region bounded between the plot of false positive rate versus true positive rate and a line emerging from origin. The bigger AUC represents a better machine learning model for discriminating the specified classes.

4.3 Performance evaluation

To evaluate and validate the robustness of the proposed HyFiPAD technique to a variety of fingerprint PAs or spoof attacks, rigorous experiments are performed. These experiments aim to test the performance of the proposed algorithm on selected datasets that won various liveness detection competitions conducted by Clarkson University during the year 2009, 2011, 2013, and 2015.

4.3.1 Performance tuning

The hyper-parameter settings may have a significant impact on the detection accuracy of the anti-spoofing model. The optimal hyperparameters are tuned for each dataset by using a Meta-process. For HyFiPAD, we choose the hyperparameters such as kernel function and C-value (regularization) in SVM, Stddev for Gaussian layer, rate in dropout layer, activation in dense layer and type of voting in majority voting classifier. The hyper-parameter tuning results of our proposed model are listed in Table 3.

Table 3 Performance evaluation of HyFiPAD at different parameter settings

Hyper-parameter	Search space	Parameter setting	Model performance (%)	Selected Value
Kernel (SVM)	[Polynomial, RBF, linear]	Polynomial	89.90	RBF
		RBF	**97.00**	
		linear	95.60	
C-value (regularization)	[0.1,1,10,100,1000]	0.1	88.95	100
		1	89.00	
		10	86.00	
		100	**96.10**	
		1000	95.00	
Stddev	[1,3.5,4.0]	1.0	93.40	3.5
		3.5	**97.00**	
		4.0	96.60	
Dropout rate	[0.1,0.2,0.3,0.35, 0.4]	0.10	95.65	0.30and 0.35
		0.20	96.20	
		0.30	**96.25**	
		0.35	**96.30**	
		0.40	96.20	
Activation (dense layer)	[ReLU, Sigmoid]	**ReLU**	**96.30**	ReLU
		Sigmoid	85.92	
Epochs (sequential model)	[20, 30, 40, 50]	20	95.20	40
		30	95.20	
		40	**96.30**	
		50	96.00	
Optimizer	['SDG', 'Adam']	SDG	70.25	Adam
		Adam	**96.90**	

The proposed HyFiPAD model is optimally tuned for various hyperparameters that demonstrate higher accuracy with selected parameters. For both the SVM classifiers, the kernel parameter chosen as RBF and a *c*-value of 100 provides best performance. On the other hand, the Stddev with a value of 3.5 result in 97.00% classification accuracy, dropout rate = 0.30 or 0.35 offers best detection results in the search space. For dense layer out of ReLU and sigmoid, former provides better performance, latter results in significant decline in accuracy by a factor of around 11%. Similarly, the model has been tuned with respect to different levels of epochs, where it provides an accuracy of 96.30% at a value of 40. For optimizing the loss in sequential model, our framework has been tested with two optimizers, namely Stochastic Descent Gradient (SDG) and Adam Delta (Adam), where the latter is chosen that tune the model with optimal performance of 96.90% classification accuracy.

4.3.2 HyFiPAD performance on LivDet 2009-LivDet 2015 datasets

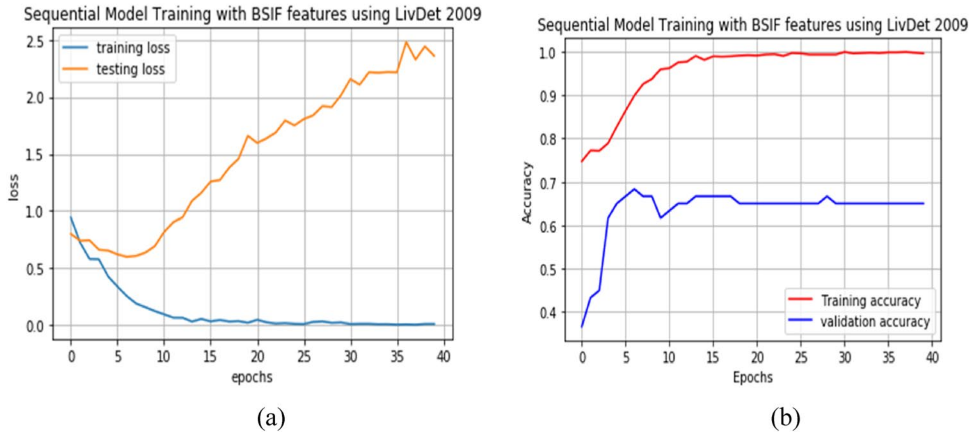
We evaluate the HyFiPAD individually on each selected benchmark dataset. The ACA and ACER values are computed for diverse combinations of image descriptors used in our technique and the combined results are illustrated in Table 4.

4.3.2.1 Experimental analysis with LivDet 2009 The combination of different feature descriptors is evaluated for samples acquired from three different sensors (i.e., Biometrika, CrossMatch, and Identix). The proposed LABP descriptor outperforms the existing CLBP in terms of ACA which is 80.7% compared to 61.0% for CLBP. It is obvious that LABP exhibits more discriminative powers than the CLBP descriptor. Moreover, the combination of CLBP and LABP that are trained on two SVM classifiers even provides an improved ACA of 84.4%. The additional improvement is achieved by augmenting a sequential model trained with BSIF features. The training and testing loss as well as accuracies across various epochs is shown in Fig. 13.

The BSIF features in our approach exploit the supremacy of simple sequential model. The effectiveness of the HyFiPAD technique is mainly due to the addition of the dual power of BSIF as well as the sequential model. The BSIF

Table 4 HyFiPAD performance with different feature sets on LivDet 2009–2015 datasets

Dataset	Methods	Classifier	ACA				
			Biometrika	CrossMatch	Identix	Average	
LivDet 2009	CLBP	SVM	60.2	63.9	59.1	61.00	
	Proposed LABP	SVM	79.8	80.0	82.4	80.70	
	CLBP + LABP	SVM + SVM	80.7	85.1	81.6	84.40	
	CLBP + BSIF	SVM + Sequential Model	91.0	89.0	86.3	88.76	
	Our HyFiPAD (CLBP + LABP + BSIF)	SVM + SVM + sequential model	95.3	97.2	96.9	96.44	
			Biometrika	Digital Persona	ItalData	Sagem	Average
LivDet 2011	CLBP	SVM	59.01	63.6	61.0	63.8	61.8
	Proposed LABP	SVM	79.0	80.12	78.01	79.3	79.1
	CLBP + LABP	SVM + SVM	83.2	82.9	79.5	77.08	80.6
	CLBP + BSIF	SVM + Sequential model	88.9	88.04	87.39	86.10	87.60
	Our HyFiPAD (CLBP + LABP + BSIF)	SVM + SVM + sequential model	94.18	97.3	97.18	96.9	96.31
			Biometrika	CrossMatch	ItalData	Swipe	Average
LivDet 2013	CLBP	SVM	62.01	62.20	60.0	63.2	61.85
	Proposed LABP	SVM	81.35	78.9	80.4	81.0	80.41
	CLBP + LABP	SVM + SVM	84.06	83.0	80.04	82.3	82.27
	CLBP + BSIF	SVM + Sequential model	86.0	89.28	86.5	90.0	87.94
	Our HyFiPAD (CLBP + LABP + BSIF)	SVM + SVM + sequential model	96.25	96.00	97.1	97.2	96.65
			Biometrika	Digital Persona	GreenBit	CrossMatch	Average
LivDet 2015	CLBP	SVM	60.2	61.9	63.1	60.7	61.47
	Proposed LABP	SVM	80.0	79.5	81.3	79.8	80.5
	CLBP + LABP	SVM + SVM	86.0	84.2	85.26	84.0	84.86
	CLBP + BSIF	SVM + Sequential model	88.9	88.0	90.1	90.4	89.35
	Our HyFiPAD (CLBP + LABP + BSIF)	SVM + SVM + sequential model	95.5	97.2	96.7	95.0	96.1



(a)

(b)

Fig. 13 **a** Training and testing loss. **b** Training and testing accuracy of sequential model

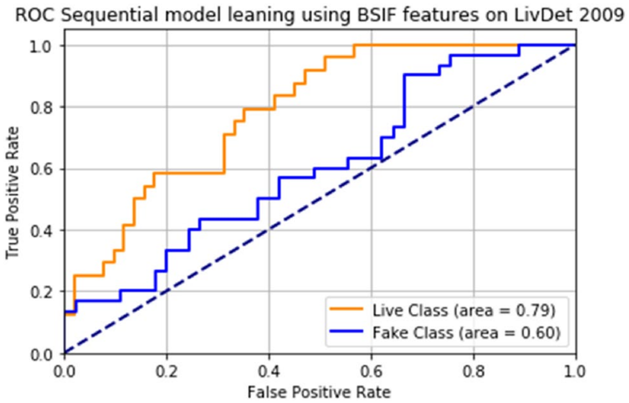


Fig. 14 The ROC curve of BSIF features trained on sequential model

features when augmented with the LABP and CLBP exhibit superior ACA of 96.44%.

The training performance of the sequential model using only BSIF features in terms of ROC is illustrated in Fig. 14. The discrimination proportion of the BSIF for HyFiPAD is evidently interpreted from the better true positive rate (TPR) along with AUC values of 0.79 and 0.60 respectively for live and fake classes.

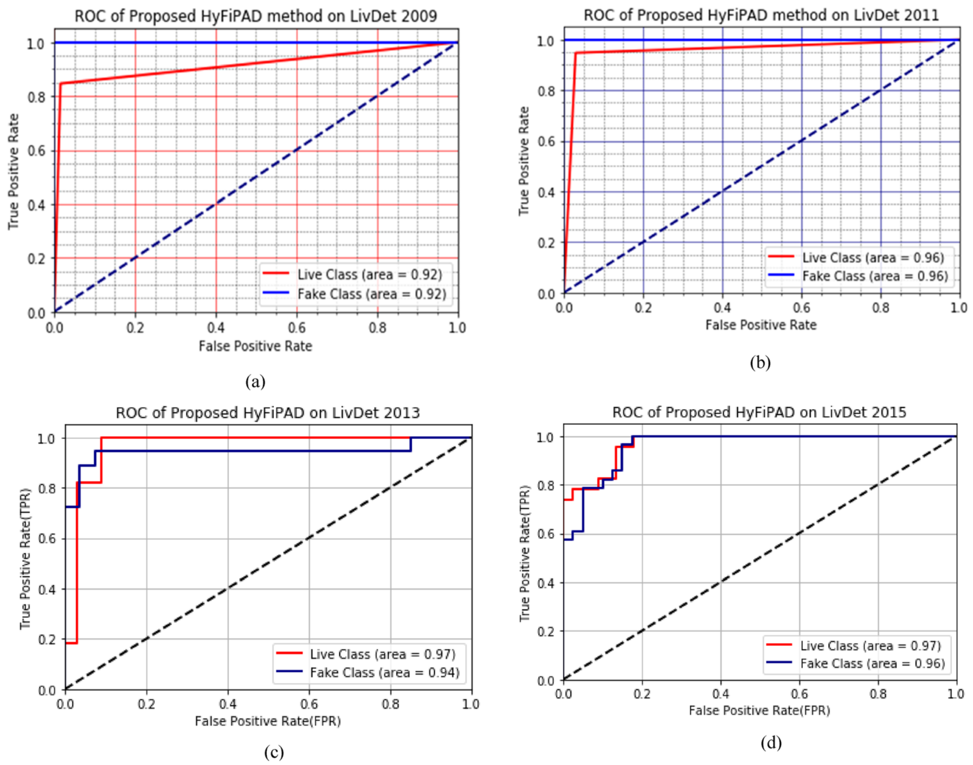
However, the proposed LABP improves ACA, but its complexity is relatively higher than CLBP. The general trend of ACA computed through a variety of combination of feature sets along with HyFiPAD demonstrate that CLBP result in the lowest ACA of 59.1% and the highest ACA is 97.2% for the HyFiPAD technique in case of CrossMatch sensor. The overall accuracy of our approach on the LivDet 2009 dataset offers ACA of 96.44% which confirms its robustness. An ROC curve of the HyFiPAD that is validated on LivDet 2009 is depicted in Fig. 15a. It indicates a higher TPR and FPR values in both the classes during a testing of the model. The AUC of 0.92 for both the live and fake classes that further indicates the effectiveness of the overall performance of our approach.

4.3.2.2 Experimental analysis with LivDet 2011 In this case, our proposed LABP descriptor performs better than CLBP with an ACA of 79.1%. Above all, interesting results are observed for CLBP on the Biometrika sensor with a lower ACA of 59.1% while the same descriptor exhibits a lower ACA of 59.1% in the case of the Identix sensor of LivDet 2009 dataset. For the proposed HyFiPAD approach, the images acquired by the ItalData sensor have shown the highest ACA of 97.18% whereas the overall accuracy of 96.31% is achieved. It is observed that the overall accuracy of the proposed technique is almost similar for both the datasets. The ROC curve of the HyFiPAD method when validated on LivDet 2011 is shown in Fig. 15b. A higher value of the TPR in both live and fake classes is reported during the testing of the model. A higher AUC of 0.96 as exhibited in the testing is quite evident of the excellent performance of our HyFiPAD method.

4.3.2.3 Experimental analysis with LivDet 2013 A similar protocol is used for evaluating our approach on LivDet 2013 dataset to explore the detection accuracies of various features descriptors. It may be clearly interpreted from Table 4 that our LABP descriptor adds a significant discriminating capability to the overall model for effectively classifying live and fake fingerprint traits. An additional improvement of 9.01% is achieved in the model with the proposed LABP descriptor as compared to ACA of 87.94% with a combination of CLBP and BSIF. The overall effectiveness of the proposed method is clearly indicated with a lower ACER of 2.88%. The ROC curve of HyFiPAD on LivDet 2013 is depicted in Fig. 15c which specifies its outstanding performance.

4.3.2.4 Experimental analysis with LivDet 2015 In yet another analysis, our approach is evaluated on a comparatively larger and more recent dataset comprising of additional generalized samples from both the classes. The results with LivDet 2015 further strengthen our claim that

Fig. 15 ROC curve performance of HyFiPAD on different datasets



(a)

(b)

(c)

(d)

Table 5 Cross-dataset performance of the HyFiPAD

Training dataset	Testing dataset	ACER (%)	ACA (%)
LivDet 2009	LivDet 2011	11.78	86.29
LivDet 2011	LivDet 2009	12.30	87.50
LivDet 2009	LivDet 2013	10.60	88.75
LivDet 2011	LivDet 2013	9.92	89.28
LivDet 2013	LivDet 2015	9.14	90.00
LivDet 2015	LivDet 2013	7.82	92.34
LivDet 2015	LivDet 2011	7.90	91.40

model effectively discriminates live and fake fingerprint traits brilliantly. The average accuracy of our approach is 96.1% which is approximately identical to results with other datasets. In the same manner, an additional classification accuracy of 6.75% is yield with augmentation of our proposed LABP descriptor with HyFiPAD. The performance in terms of ROC is highlighted in Fig. 15d with AUC for both the classes.

4.3.3 Cross-database and cross-sensors evaluation

Another significant aspect of any PAD method is its effectiveness in terms of generalization capability to unknown attacks. Therefore, we perform an experiment to evaluate the performance of our approach in cross-database and cross-sensor scenario. To generalize the PAD technique across unknown attacks, cross-database testing is performed where the model is trained and validated on different datasets consisting of fingerprint artifacts generated from different spoofing materials. In this test, we trained our model using images from a dataset and examined with samples from other dataset and the results are depicted in Table 5. The experiment focuses on testing the approach by using fingerprint artifacts generated from materials like Silicon, Gelatine, and Play-doh in LivDet 2009, Eco-flex, Latex, Gelatine, Silgum, and Wood-glue are used for creating fake fingerprints in LivDet 2011. Similarly, artifacts in the LivDet 2013 and LivDet 2015 are created by using Gelatine, Wood glue, Latex, Ecoflex, Modasil; and Ecoflex, Body double, Wood glue, Play-doh, and Latex, respectively. Table 5 shows that in seven different scenarios, dissimilar training and testing datasets are used to evaluate the performance of HyFiPAD. In the case of 2015 dataset, the ACA is above 90%. The highest accuracy is exhibited by the HyFiPAD in cross-database scenario when the model is trained with LivDet 2015 dataset and tested on LivDet 2013 samples. This may be because of the fact that LivDet 2015 dataset comprised of wide range of images acquired from different spoofing materials and sensing technologies. Additionally, when LivDet 2009 dataset is chosen either for training or testing purpose highest error rates are achieved. It may be due to deprived quality of images in LivDet 2009 dataset. Further, it can be noticed from Table 5 that the implication of

Table 6 Cross-sensor testing of HyFiPAD method on different datasets

LivDet 2009 Sensors		ACER (%)	ACA (%)
Training	Testing		
Biometrika	CrossMatch	19.02	80.0
Identix	CrossMatch	11.9	79.0
LivDet 2011 sensors			
Biometrika	Digital persona	12.0	81.67
Digital Persona	Sagem	13.9	81.3
Sagem	Digital persona	13.4	84.1
LivDet 2013 sensors			
ItalData	Swipe	11.5	86.6
CrossMatch	Biometrika	17.0	82.5
Swipe	ItalData	14.9	84.0
LivDet 2015 sensors			
GreenBit	Digital persona	12.4	87.07
Biometrika	GreenBit	16.82	82.70
CrossMatch	Digital persona	13.10	84.62

proposed HyFiPAD algorithm mitigates the unknown attack with moderate accuracy that depicts its good generalization capability.

An additional essential factor that pertains to a typical PAD model is to evaluate its performance on images captured under different sensing techniques and conditions. Thus, we experiment to validate our approach in cross-sensor scenario for sets of images acquired through different sensors in all the datasets. The performance in terms of ACER and ACA for cross-sensor testing is shown in Table 6. The results obtained in cross-sensor testing cover lower ACER compared to the cross-database scenario. Further, we carry out exhaustive experimentation for some combinations of training and testing image sets acquired with different sensors. The illustration in Table 6 infers that our approach exhibit better accuracy in GreenBit-Digital Persona cross-sensor testing scenario for LivDet 2015 followed by ItalData-Swipe pair with accuracy of 86.6% for LivDet 2013. In terms of ACER the highest error rates are reported in case of LivDet 2009 Biometrika–CrossMatch pair. Similarly, the CrossMatch–Biometrika pair in LivDet 2013 results in ACER of 17.0%. It may also be seen that the overall accuracy of the cross-sensor evaluation lies between 79.0 and 87.07% that suggests a reasonable generalization capability of our approach.

4.3.4 Ablation study

We perform several ablation experiments to evaluate HyFiPAD under all selected datasets successively by removing the key features and some of the layers from the sequential model. The performances as ACA under different scenario

Table 7 An ablation study for HyFiPAD approach

Ablated component	LivDet 2009		Average	LivDet 2011			Average	LivDet 2013				Average	LivDet 2015			Average
	Cross-Match	Biom-etrika	Identix		Biom-etrika	Digital persona	ItalData	Sagem		Biom-etrika	Cross-Match	ItalData	Swipe		Digital persona	Green-Bit	Cross-Match	
LABP	83.20	82.89	79.49	81.86	88.90	88.04	87.39	86.10	87.60	87.29	89.54	86.26	91.31	88.60	90.10	89.72	87.34	91.40
BSIF	80.70	85.10	81.60	82.46	83.20	82.90	79.50	77.08	80.67	83.60	82.03	81.04	83.20	82.46	82.47	83.22	84.26	85.20
CLBP	70.01	71.90	70.75	70.88	69.05	71.86	71.75	70.45	70.77	72.01	73.90	73.75	74.31	73.49	70.57	71.16	73.75	73.82
Dense layer2	95.09	96.79	97.01	96.29	95.00	97.25	96.90	97.10	96.56	96.06	95.48	96.76	97.32	96.40	93.80	95.25	97.27	96.70
Gaussian layer1	93.19	95.30	94.88	94.45	95.37	94.90	95.73	96.01	95.50	91.98	97.45	96.62	95.33	95.34	96.27	93.90	97.40	95.35
Dropout layer1	96.38	97.21	97.81	97.13	97.01	96.63	97.03	97.20	96.96	93.38	96.21	97.81	97.13	96.13	97.01	96.29	96.03	96.71
Dropout layer2	95.01	96.70	97.50	96.40	94.08	97.96	96.18	96.28	96.12	96.00	96.69	97.50	97.40	96.89	94.08	96.60	97.18	95.97

are listed in Table 7 for different sensors of LivDet 2009 to LivDet 2015 datasets. The ablation of our proposed LABP descriptor in HyFiPAD using LivDet 2009 results in an ACA of 81.86% which significantly degrades the performance of the model by 14.58%. Thus, it demonstrates the robustness of LABP features that offer additional discrimination power to the overall system. As opposite to this, in case of LivDet 2015 the average ACA with ablation of our LABP results in 91.40%. The successive ablation of existing CLBP and BSIF descriptor in HyFiPAD reduces the accuracy by 25.56 and 13.98%, respectively. An enhancement of 8.71 and 9.54% in the accuracy is indicated with an ablation of LABP features using LivDet2011 and LivDet 2015 dataset that further proves worth of the descriptor. Furthermore, ablation with the CLBP and BSIF feature descriptors from the system results in reduction of accuracy by an amount of 25.54 and 15.64%, respectively. The overall ablation with features in our approach exhibits the substantial potency of LABP in comparison with other descriptors in the model.

We also evaluated the effect on the HyFiPAD by ablating few layers in the sequential model that learns with BSIF features. The Dense layer2 ablation result in an ACA of 96.29% with LivDet 2009 and 96.56% LivDet 2011,whereas for LivDet 2013 and 2015 their results are 96.40 and 96.70% respectively, that is approximately identical to the overall performance of the model. However, Gaussian layer1 ablation declines the accuracy by an amount of ~ 1.0–2.0% (approx.). Surprisingly, a slight improvement in accuracy is observed with ablation of Dropout layer1 and a mild drop in accuracy with Dropout layer2 with all the datasets. The overall ablation experiments offer promising performance of the HyFiPAD model with different scenarios.

4.3.5 Comparison of HyFiPAD against state-of-the-art methods

To address the problem of fingerprint anti-spoofing, several texture-based techniques are presented in the literature. Since our method is hybrid and uses the new proposed variant of local features together with existing descriptors along with SVM and a basic sequential model. To evaluate the effectiveness of the HyFiPAD, we compare it with similar state-of-the-art methods based on either single or multiple image feature descriptors. Here, we carry out the comparative analysis of HyFiPAD with both handcrafted and DL-based state-of-the-art methods evaluated on the identical

Table 8 Comparative analysis of HyFiPAD using LivDet 2009 dataset

Technique	Feature engineering		ACER (%)
	Handcrafted	DL-Based	
Image quality Assessment based [65]	✓	✗	8.20
Wavelet Markov Descriptor [17]	✓	✗	2.83
CNN + LBP [44]	✓	✓	4.75
CNN [48]	✗	✓	3.42
CNN Random [46]	✓	✓	3.35
CNN with patch-based voting approach [49]	✗	✓	5.00
Gradient-based texture features [6]	✓	✗	6.20
Ridge-Valley structure Local Quality features [9]	✓	✗	5.30
HyFiPAD	✓	✗	4.11

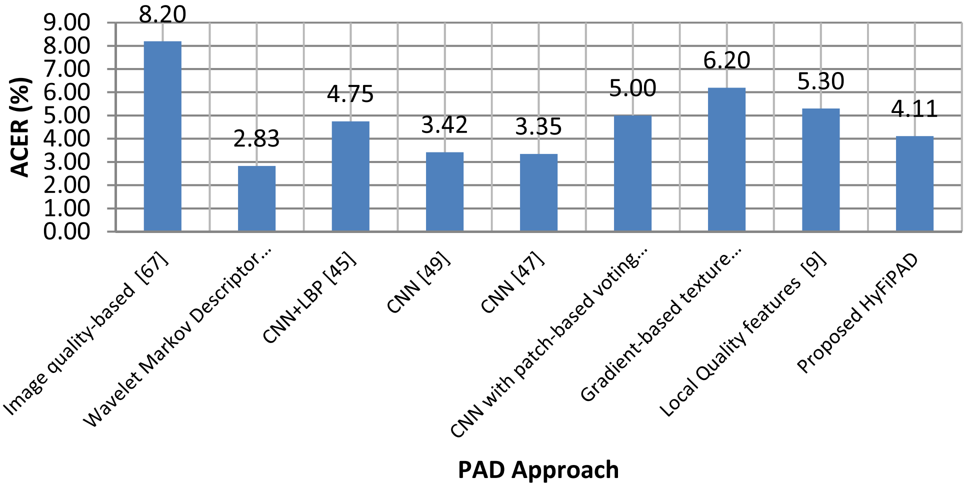


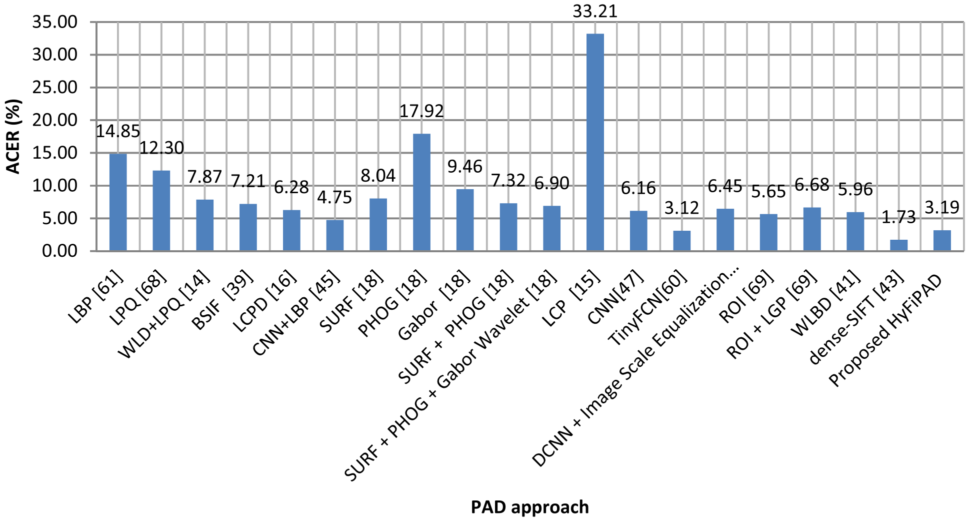
Fig. 16 Performance comparison of HyFiPAD on LivDet 2009 dataset

Table 9 The comparison of HyFiPAD using LivDet 2011 and LivDet 2013 dataset

Technique	Feature engineering		ACER (%)	
	Handcrafted	DL-Based	LivDet 2011	LivDet 2013
LBP [58]	✓	✗	14.85	4.90
LPQ [66]	✓	✗	12.30	–
WLD[14]	✓	✗	15.33	6.90
WLD + LPQ [14]	✓	✗	7.87	14.31
BSIF [39]	✓	✗	7.21	4.61
LCPD [16]	✓	✗	6.28	2.40
CNN + LBP [44]	✓	✓	4.75	–
SURF [18]	✓	✗	8.04	5.26
PHOG [18]	✓	✗	17.92	7.24
Gabor [18]	✓	✗	9.46	3.50
SURF + PHOG [18]	✓	✗	7.32	3.85
SURF + PHOG + Gabor [18]	✓	✗	6.90	4.20
LCP [15]	✓	✗	33.21	27.95
CNN Random [46]	✓	✓	6.16	5.71
TCAG [6]	✓	✗	6.63	7.96
WLBD [10]	✓	✗	5.96	1.89
TinyFCN[67]	✗	✓	3.12	3.12
DCNN + ISE [54]	✗	✓	6.45	3.70
ROI [68]	✗	✓	5.65	2.99
ROI + LGP [68]	✗	✓	6.68	2.96
DenseNet model[69]	✓	✗	–	1.73
Q-FFF [70]	✓	✗	–	2.15
FLDNet [56]	✓	✗	–	1.18
dense-SIFT [42]	✓	✗	1.73	–
LivDet Winner	–	–	22.92	13.36
HyFiPAD	✓	✗	3.19	2.88

datasets (e.g., LivDet 2009) in terms of ACER. The results are recorded in Table 8 and a comparative plot in Fig. 16 implies the effectiveness of our technique with an ACER of 4.11%. This performance noticeably outperform the other methods in [6, 9, 17, 19, 45, 47, 49, 50, 67], where superior performance is exhibited by [17] with an ACER of 2.83%, which is nearly close to our model. However, our model also shows comparable performance against the modern CNN-based approaches in [47, 49], but these approaches involve additional levy in terms of training overhead.

Fig. 17 Performance comparison of HyFiPAD on LivDet 2011 dataset



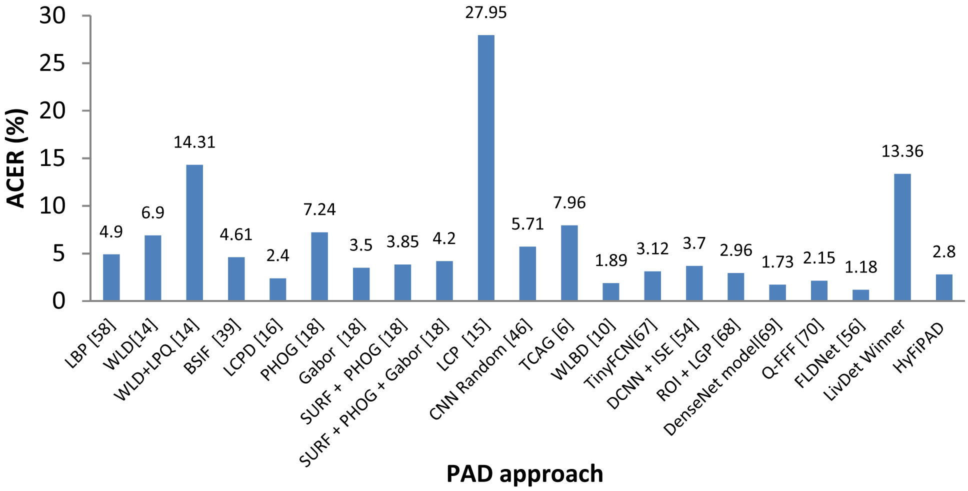


Fig. 18 A comparison of HyFiPAD with related approaches on LivDet 2013 dataset

Table 10 A comparative analysis of HyFiPAD on LivDet 2015 dataset

Technique	Feature engineering		ACER (%)
	Handcrafted	DL-Based	
LBP [58]	✓	✗	11.82
WLD [14]	✓	✗	14.27
LCP [15]	✓	✗	13.61
MSDCM (deep residual 14)	✓	✗	7.59
WLBD [10]	✓	✗	9.68
HoG [71]	✓	✗	5.27
DCNN [72]	✗	✓	0.97
SlimResCNN [53]	✗	✓	3.11
ROI [68]	✗	✓	5.32
ROI + LGP [68]	✗	✓	6.11
DenseNet model [69]	✗	✓	1.61
Q-FFF [70]	✓	✗	6.58
FLDNet [56]	✗	✓	2.22
HyFiPAD	✓	✗	2.97

Though, Park et al. [49] achieve an ACER of 3.42% that is comparatively smaller than our results, but their PAD model is trained with a smaller dataset of merely 523 fingerprint images.

Likewise, Table 9 depicts a comparison of our approach opposed to the existing state-of-the-art methods that were evaluated on LivDet 2011 and LivDet 2013 datasets with ACER protocol. Figure 17 illustrates the comparative results of techniques evaluated on LivDet 2011, it may be noticed that, HyFiPAD yields ACER of 3.19%, which is further an evident of its efficiency.

The HyFiPAD exhibits superior performance compared to majority of the PAD techniques on both the texture and deep learning-based counterparts. On the other hand, it reveal almost equivalent performance against some approaches that utilize TinyFCN (ACER = 3.12%) and dense-SIFT algorithm (ACER = 1.73%) where former is a deep learning-based method and latter exploits scale-invariant texture features.

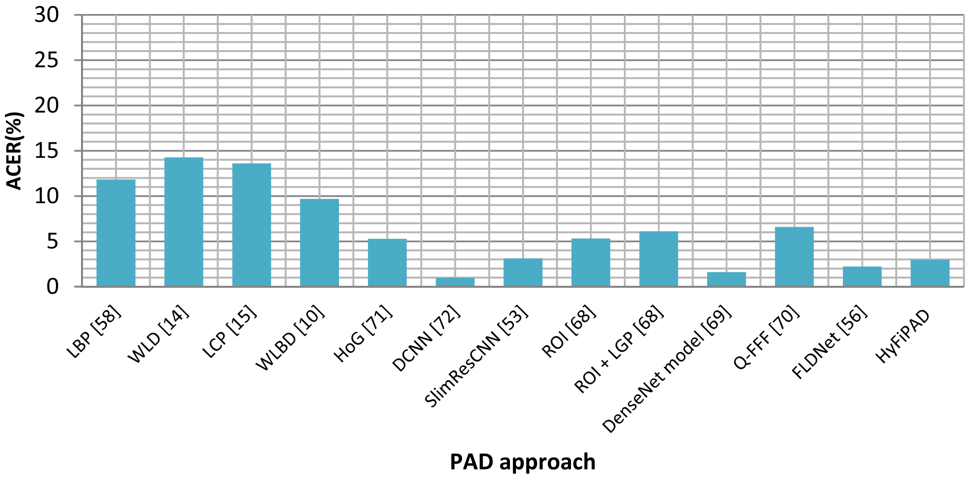


Fig. 19 Comparative analysis of HyFiPAD on LivDet 2015 dataset

The comparison of HyFiPAD against state-of-the-art techniques evaluated on LivDet 2013 dataset is depicted in Fig. 18. The diagram clearly demonstrates that the HyFiPAD outperforms the existing handcrafted feature-based PAD techniques except for WLBD where the ACER is 1.89% that is almost close to 2.8% for our method. In case of DL-based techniques, the DenseNet and FLDNet models surpasses our approach with an ACER of approximately 1.0%.

At last, the HyFiPAD is compared against the techniques that are evaluated on LivDet 2015 dataset and results are depicted in Table 10. The HyFiPAD outperforms the state-of-the-art PAD approaches.

From Fig. 19, it is may be inferred that the results of DCNN [72], DenseNet [69], and FLDNet [56] models are superior to our HyFiPAD approach. The core reason behind this better performance is that these models exploit deep level features from the fingerprint images to build CNN architecture. In other cases, our technique overtakes the existing PAD techniques. The handcrafted feature-based techniques show higher ACER value as compared to our resultant ACER of 3.19%.

The overall experimental analysis of the novel HyFiPAD can put forward several advantages: (i) the HyFiPAD explores the proposed LABP micro-textural image features with existing CLBP and BSIF descriptors. (ii) The power of a basic sequential model trained on BSIF features along with two SVM classifiers work as an integrated hybrid model with better accuracy and lower error rate. (iii) Our technique provides a promising performance in the case of both cross-database and cross-sensor scenario that supports a good generalization capability of the proposed approach to unknown attacks. (iv) The proposed LABP helps to improve the discriminative power of combination of CLBP and BSIF by an average accuracy of 8.71%. (v) The HyFiPAD outperforms the state-of-the-art PAD methods in terms of its efficiency on the majority of the selected datasets.

5 Conclusions

This research work addresses the fingerprint presentation attack detection mechanism by using a hybrid approach, where the newly proposed LABP features are coalesced with two other existing CLBP and BSIF image feature descriptors to boost the classification accuracy. The local threshold used in proposed LABP helps to extract more refined discriminative information from the local region that grants enhanced discrimination capabilities in comparison with earlier version of LBP. Furthermore, a majority voting ensemble is created on the three trained classifiers to attain the final decision. Our analysis over four datasets reflects that the performance of our HyFiPAD technique outperforms the existing similar state-of-the-art FiPAD methods. The extensive experiments confirmed that our method exhibit stable performance in generalization to unknown attacks. Although our modified LBP descriptor (LABP) helps to improve the classification accuracy of the hybrid model, one of the limitations includes its comparatively higher complexity that may be improved in future. Besides, we plan to evaluate our HyFiPAD approach on few more recent datasets such as LivDet 2017 and LivDet 2019. An additional future work directs to improve the accuracy of HyFiPAD model particularly in the cases of cross-sensor and database scenarios, as it will more effectively deal with the problem of unknown attacks.

Appendix

See Appendix Table 11.

Table 11 A summary of state-of-the-art fingerprint PAD techniques

Year	Author(s)	Handcrafted features-based	Deep learning-based	Key concept	Classifier	Datasets	Performance
2010	Marasco and Sansone [28]	✓	✗	Image Quality Features	LDA	LivDet 2009, ATVS	ACA = 90%
2013	Gragnaniello et al. [14]	✓	✗	WLD	SVM	LivDet2009 and LivDet 2011	Results demonstrated superiority
2013	Ghiani et al. [39]	✓	✗	BSIF	ICA	LivDet 2011	7.215%
2013	Pereira et al. [73]	✓	✗	Spatial surface coarseness	SVM	LivDet 2011	12.8%
2014	Zhang et al. [13]	✓	✗	Wavelet Analysis/LBP	SVM	LivDet2011 and LivDet2013	ACA, LivDet 2011 = 88.53% and LivDet2013 = 88.98%
2014	Gragnaniello et al. [16]	✓	✗	LCPD	SVM	LivDet 2011	5.7%
2014	Gragnaniello et al. [17]	✓	✗	Wavelet Markov descriptor	SVM with RBF	LivDet 2009	2.83%
2014	Nogueira et al. [44]	✓	✓	CNN and LBP	–	LivDet 2009, LivDet 2011, and LivDet 2013	4.75% for CNN and 9.67% for LBP
2015	Wang et al. [45]	✗	✓	DCNN with voting strategy	–	LivDet 2011 and LivDet 2013	Better overall results are achieved
2016	Xia et al. [6]	✓	✗	Co-occurrence array construction	SVM	LivDet2009 and LivDet2011	TCGA have better classification as compared to SCGA
2016	Dubey et al. [18]	✓	✗	PHOG + SURF and Gabor wavelet	SVM, Random forest	LivDet 2011, LivDet 2013	EER = 3.95% for LivDet 2011 and for LivDet 2013 is 2.27%
2016	Park et al. [48]	✗	✓	CNN	–	LivDet 2009	3.42%
2016	Nogueira et al. [46]	✗	✓	CNN	–	LivDet 2009, LivDet 2011 and LivDet 2013	ACA = 97.1%
2017	Kim [15]	✓	✗	LCP	SVM	ATVS and LivDet2009	ACA for ATVS = 95.63% for LivDet2009 = 86.83%
2017	Jung and Heo [51]	✗	✓	CNN	–	LivDet 2015	Accuracy degeneration of Digital Persona was lowest
2017	Lazimul and Binoy [50]	✗	✓	CNN with image enhancement	–	–	ACA = 98%
2018	Xia et al. [10]	✓	✗	WLBD	SVM	LivDet 2009, LivDet 2011 and LivDet 2015	5.96%, 1.89% and 9.67% respectively
2018	Jiang and Liu [8]	✓	✗	ULBP	SVM	LivDet 2013	21.20%
2019	Souza et al. [52]	✓	✓	DBM	SVM	LivDet2013	ACA = 85.82%
2019	Agarwal et al. [41]	✓	✗	LBP	Ensemble	LivDet 2011	ACA lies between 70%-80%
2019	Uliyan et al. [55]	✗	✓	RBM and DBM	–	LivDet 2013 and LivDet 2015	3.6% for LivDet 2013
2019	Zhang et al. [53]	✗	✓	SlimResCNN	–	LivDet 2011, LivDet 2015 and LivDet 2017	ACA = 95.25%
2019	Yuan et al. [54]	✗	✓	DCNN with image scale equalization	–	LivDet 2011 and LivDet 2013	6.45% for LivDet 2011 and 3.7% for LivDet2013
2019	Park et al. [67]	✗	✓	FCN and SqeezeNet	–	LivDet 2011, LivDet 2013 and LivDet 2015	0.27% of improvement in error detection rate

Table 11 (continued)

Year	Author(s)	Handcrafted features-based	Deep learning-based	Key concept	Classifier	Datasets	Performance
2020	Zhang et al. [56]	✗	✓	Lightweight CNN	–	LivDet 2013 and LivDet 2015	1.76% for LivDet 2015
2020	Zhang et al. [57]	✗	✓	Fingerprint matching and liveness detection fusion	–	LivDet 2019	ACA = 96.88%
2021	Gonzlez-Soler et al. [42]	✓	✗	Dense-SIFT Descriptors	SVM	LivDet2011 to LivDet2019	ACA = 96.17%

Acknowledgements The LivDet datasets employed in this research are kindly provided by Clarkson University for which authors express their gratitude.

Declarations

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

References

1. Jain, A.K., Ross, A., Prabhakar, S.: An Introduction to Biometric Recognition. IEEE Trans. Circuits Syst. Video Technol. **14**(1), 4–20 (2004)
2. Jain, A.K., Ross, A.: Handbook of Biometrics, pp. 1–22. Springer, London (2008)
3. Minaee, S., Azimi, E., Abdolrashidi, A.: Fingernet: Pushing the limits of fingerprint recognition using convolutional neural network. arXiv preprint arXiv:1907.12956 (2019)
4. Ratha, N.K., Connell, J.H., Bolle, R.M.: An analysis of minutiae matching strength. In: Proc. AVBPA 2001, 3rd International Conference on Audio- and Video-Based Biometric Person Authentication, pp. 223–228 (2001)
5. Ghiani, L. et al.: LivDet 2013 fingerprint liveness detection competition 2013. In: 2013 international conference on biometrics (ICB), pp. 1–6 (2013)
6. Xia, Z., Lv, R., Zhu, Y., Ji, P., Sun, H.: Fingerprint liveness detection using gradient-based texture features. Signal, Image Video Process. **11**(2), 381–388 (2017)
7. Ghiani, L., Hadid, A., Marcialis, G.L., Roli, F.: Fingerprint liveness detection using local texture features. IET Biometr. **6**(3), 224–231 (2017)
8. Jiang, Y., Liu, X.: Uniform local binary pattern for fingerprint liveness detection in the gaussian pyramid. J. Electr. Comput. Eng. **2018**, 1–9 (2018)
9. Sharma, R.P., Dey, S.: Fingerprint liveness detection using local quality features. Vis. Comput. **35**, 1393–1410 (2019)
10. Xia, Z., Yuan, C., Lv, R., Sun, X., Xiong, N.N., Shi, Y.Q.: A novel weber local binary descriptor for fingerprint liveness detection. IEEE Trans. Syst. Man, Cybern. Syst. **50**(4), 1526–1536 (2020)
11. Moon, Y.S., Chen, J.S., Chan, K.C., So, K., Woo, K.C.: Wavelet based fingerprint liveness detection. Electron. Lett. **41**(20), 2–4 (2005)
12. Nikam, S.B., Science, C., Science, C.: Texture and wavelet-based spoof fingerprint detection for fingerprint biometric systems. In: 1st International Conference on Emerging Trends in Engineering and Technology Texture, pp 675–680 (2008)
13. Zhang, Y., Fang, S., Xie, Y., Xu, T.: Fake fingerprint detection based on wavelet analysis and local binary pattern. In: Sun, Z., Shan, S., Sang, H., Zhou, J., Wang, Y. (eds.) Biometric recognition, CCBR (2014) Lecture notes in computer science, pp. 191–198. Springer, Cham (2014)
14. Gragnaniello, D., Poggi, G., Sansone, C., Verdoliva, L.: Fingerprint liveness detection based on weber local image descriptor. In: 2013 IEEE workshop on biometric measurements and systems for security and medical applications, September (2013)
15. Kim, W.: Fingerprint liveness detection using local coherence patterns. IEEE Signal Process. Lett. **1**(1), 1–5 (2016)
16. Gragnaniello, D., Poggi, G., Verdoliva, L.: Local contrast phase descriptor for fingerprint liveness detection. Pattern Recogn. **48**(4), 1050–1058 (2015)
17. Gragnaniello, D., Poggi, G., Sansone, C., Verdoliva, L.: Wavelet-Markov local descriptor for detecting fake fingerprints. Electron. Lett. **50**(6), 439–441 (2014)
18. Dubey, R.K., Goh, J., Thing, V.L.L.: Fingerprint liveness detection from single image using low-level features and shape analysis. IEEE Trans. Inf. Forensics Secur. **11**(7), 1461–1475 (2016)
19. Reddy, P.V., Kumar, A., Member, S., Rahman, S.M.K., Mundra, T.S.: A new antispoofing approach for biometric devices. IEEE Trans. Biomed. Circuits Syst. **2**(4), 328–337 (2008)
20. Drahanský, M., Nötzel, R., Funk, W.: Liveness detection based on fine movements of the fingertip surface. In: Proceedings of the 2006 IEEE Workshop on Information Assurance United States Military Academy, West Point, pp. 42–47 (2006)
21. Baldisserra, D., Franco, A., Maio, D., Maltoni, D., Bologna, U.: Fake fingerprint detection by odor analysis. In: Z. D. and J. A. K. (eds.) Advances in Biometrics, ICB 2006, Lecture Notes in Computer Science, pp. 265–272. Springer, Berlin, Heidelberg (2006)
22. Martinsen, Ø.G., Member, S., Clausen, S., Nysæther, J.B., Grimnes, S.: Utilizing characteristic electrical properties of the epidermal skin layers to detect fake fingers in biometric fingerprint systems—a pilot study. IEEE Trans. Biomed. Eng. **54**(5), 891–894 (2007)
23. Galbally, J., Fierrez, J., Cappelli, R.: An introduction to fingerprint presentation attack detection. In: Marcel, S., Nixon, M., Fierrez, J., Evans, N. (eds.) Handbook of Biometric Anti-Spoofing. Advances in Computer Vision and Pattern Recognition. Springer, Cham (2019). https://doi.org/10.1007/978-3-319-92627-8_1
24. Coli, P., Marcialis, G.L., Roli, F.: Fingerprint silicon replicas: static and dynamic features for vitality detection. Int. J. Image Graph. **8**(4), 495–512 (2008)
25. Bhanarkar, A., Doshi, P.: Joint time frequency analysis based liveness fingerprint detection. In: 2013 IEEE second international

conference on image information processing (ICIIP-2013), pp. 166–169 (2013)

26. Galbally, J., Alonso-Fernandez, F., Fierrez, J., Ortega-Garcia, J.: A high performance fingerprint liveness detection method based on quality related features. Futur. Gener. Comput. Syst. **28**(1), 311–321 (2012)
27. Galbaly, J., Alonso-Fernandez, F., Fierrez, J., Ortega-Garcia, J.: Fingerprint liveness detection based on quality measures. In: 2009 1st IEEE International Conference on Biometrics, Identity and Security (BIdS), pp. 1–9 (2009)
28. Marasco, E., Sansone, C.: An anti-spoofing technique using multiple textural features in fingerprint scanners. In: 2010 IEEE Workshop on Biometric Measurements and Systems for Security and Medical Applications, pp. 8–14 (2010)
29. Espinoza, M., Champod, C., Margot, P.: Vulnerabilities of fingerprint reader to fake fingerprints attacks. Forensic Sci. Int. **204**(1–3), 41–49 (2011)
30. Ghiani, L., Marcialis, G.L., Roli, F.: Fingerprint liveness detection by local phase quantization. In: 21st International Conference on Pattern Recognition (ICPR 2012), no. ICPR, pp. 537–540 (2012)
31. Manivannan, N., Balachandran, W.: Active pore detection for liveness in fingerprint identification system. In: 19th Telecommunications forum TELFOR 2011, pp. 619–622 (2011)
32. Lu, M., Chen, Z., Sheng, W.: A pore-based method for fingerprint liveness detection. In: 2015 International Conference on Computer Science and Applications A, pp. 14–15 (2015)
33. Antonelli, A., Cappelli, R., Maio, D., Maltoni, D.: A new approach to fake finger detection based on skin distortion. In: Zhang, D. (ed.) ICB 2006, LNCS, Springer, Berlin, Heidelberg, pp. 221–228 (2006)
34. Jia, J., Cai, L., Zhang, K., Chen, D.: A new approach to fake finger detection based on skin elasticity analysis. In: S.-W. L. and Li, S.Z. (eds.) ICB 2007, LNCS 4642, pp. 309–318. Springer, Berlin, Heidelberg (2007)
35. Abhyankar, A., Schuckers, S.: Integrating a wavelet based perspiration liveness check with fingerprint recognition. Pattern Recogn. **42**, 452–464 (2009)
36. Abhyankar, A., Schuckers, S.: Modular decomposition of fingerprint time series captures for the liveness check. Int. J. Comput. Electr. Eng. **2**(3), 1793–8163 (2010)
37. Parthasaradhi, S.T.V., Derakhshani, R., Hornak, L.A., Schuckers, S.A.C.: Time-series detection of perspiration as a liveness test in fingerprint devices. IEEE Trans. Syst. Man, Cybern. C Appl. Rev. **35**(3), 335–343 (2005)
38. Jia, X.Y., Cao, K., Zang, Y., Zhang, N., Dai, R., Zhu, X., Tian, J.: nMulti-scalelocal binary pattern with filters for spoof fingerprintdetection. Inf. Sci. **268**, 91–102 (2014). https://doi.org/10.1016/j.ins.2013.06.041
39. Ghiani, L., Hadid, A., Marcialis, G.L., Roli, F. : Fingerprint liveness detection using binarized statistical image features. In: 2013 IEEE 6th International Conference on Biometrics: Theory, Applications and Systems (BTAS), pp. 1–6 (2013)
40. Li, Q., Chan, P.P.K.: Fingerprint liveness detection based on binarized statistical image feature with sampling from Gaussian distribution. In: Proceedings of the 2014 International Conference on Wavelet Analysis and Pattern Recognition, pp. 13–16 (2014)
41. Agarwal, S., Chowdary, C.R.: A-stacking and a-bagging: adaptive versions of ensemble learning algorithms for spoof fingerprint detection. Expert Syst. Appl. **146**, 113160 (2019)
42. González-soler, L.J., Gomez-barrero, M., Chang, L., Pérez-suárez, A., Busch, C., Member, S.: Fingerprint presentation attack detection based on local features encoding for unknown attacks. IEEE Access **9**, 5806–5820 (2021)
43. Minaee, S., Abdolrashidi, A., Su, H., Bennamoun, M., Apr, C.V.: Biometric recognition using deep learning : a survey (2021)
44. Nogueira, R.F., De Alencar, R.: Evaluating software-based fingerprint liveness detection using convolutional networks and local binary patterns. In: 2014 IEEE Workshop on Biometric Measurements and Systems for Security and Medical Applications (BIOMS) Proceedings, Rome, pp. 22–29 (2014)
45. Wang, C., Li, K., Wu, Z., Zhao, Q.: A DCNN based fingerprint liveness detection algorithm with voting strategy. In: (2015) Chinese Conference on Biometric Recognition, pp. 241–249 (2015)
46. Nogueira, R.F., Lotufo, R.D.A., Machado, R.C.: Fingerprint liveness detection using convolutional neural networks. IEEE Trans. Inf. Forens. Secur. **11**(6), 1206–1213 (2016)
47. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional convolutional neural networks. Nips **1**, 1–35 (2012)
48. Park, E., Kim, W., Li, Q., Kim, J., Kim, H.: Fingerprint liveness detection using CNN features of random sample patches. In: BIOSIG 2016, Lecture Notes in Informatics, Geslleschaft fur Informatic, Bonn (2016)
49. Toosi, A.; Cumani, S. Bottino, A.: CNN Patch–Based Voting for Fingerprint Liveness Detection. In: Proceedings of the 9th International Joint Conference on Computational Intelligence - IJCCI, ISBN 978-989-758-274-5 ISSN 2184-2825, pp. 158-165 (2017). https://doi.org/10.5220/0006582101580165
50. Lazimul, L.T.P., Binoy, D.L.: Fingerprint liveness detection using convolutional neural network and fingerprint image enhancement. In: 2017 Int. Conf. Energy, Commun. Data Anal. Soft Comput., pp. 731–735 (2017)
51. Jung, H.Y., Heo, Y.S.: Fingerprint liveness map construction using convolutional neural network. Electron. Lett. **54**(9), 8–9 (2018)
52. De Souza, G.B., Felipe, D., Gonc, R., Papa, P., Marana, A.N.: Deep features extraction for robust fingerprint spoofing attack detection. JAISCR **9**(1), 41–49 (2019)
53. Zhang, Y., Shi, D., Zhan, X., Cao, D., Zhu, K., Li, Z.: Slim-ResCNN : a deep residual convolutional neural network for fingerprint liveness detection. IEEE Access **7**, 91476–91487 (2019)
54. Yuan, C., Xia, Z., Jiang, L.: Fingerprint liveness detection using an improved CNN with image scale equalization. IEEE Access **7**, 26953–26966 (2019)
55. Uliyan, D.M., Sadeghi, S., Jalab, H.A.: Anti-spoofing method for fingerprint recognition using patch based deep learning machine. Eng. Sci. Technol. Int. J. **2**, 264–273 (2020)
56. Zhang, Y., Pan, S., Zhan, X., Li, Z., Gao, M., Gao, C.: FLDNet: light dense CNN for fingerprint liveness detection. IEEE Access **8**, 84141–84152 (2020)
57. Zhang, Y., Gao, C., Xu, Y., Qiu, H.: A score-level fusion of fingerprint matching with fingerprint liveness detection. IEEE Access **8**, 183391–183400 (2020)
58. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray scale and rotation invariant texture classification with local binary patterns. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformat.) **1842**, 404–420 (2000)
59. Guo, Z., Zhang, L., Zhang, D.: A completed modeling of local binary pattern operator for texture classification. IEEE Trans. Image Process. **19**(6), 1657–1663 (2010)
60. Kannala, J., Rahtu, E.: BSIF: binarized statistical image features. In: Proc. - Int. Conf. Pattern Recognit., pp. 1363–1366 (2012)
61. Vapnik, V.F., Lerner, A.Y.: Pattern recognition using generalized portraits. Autom. Remote Control **24**(6), 774–780 (1963)
62. Marcialis, G.L. et al.: First international fingerprint liveness detection competition-LivDet 2009. In: Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformat.), vol. 5716 LNCS, pp. 12–23 (2009)
63. Yambay, D., Ghiani, L., Denti, P., Marcialis, G.L., Roli, F., Schuckers, S.: LivDet 2011—Fingerprint liveness detection competition 2011. In: Proc.—2012 5th IAPR Int. Conf. Biometrics, ICB 2012, pp. 208–215 (2012)

64. Mura, V., Ghiani, L., Marcialis, G.L., Roli, F., Yambay, D.A., Schuckers, S.A.: LivDet 2015 fingerprint liveness detection competition 2015, In: 2015 IEEE 7th Int. Conf. Biometrics Theory, Appl. Syst. BTAS 2015, pp. 0–5 (2015)
65. Galbally, J., Marcel, S., Fierrez, J.: Image quality assessment for fake biometric detection: application to iris, fingerprint, and face recognition. IEEE Trans. Image Process. **23**(2), 710–724 (2014)
66. Ojansivu, V., Heikkil, J.: Blur insensitive texture classification using local phase quantization ville. In: ICISP, LNCS 5099. Springer, Berlin, Heidelberg, pp. 236–243 (2008)
67. Park, E., Cui, X., Hai, T., Nguyen, B., Kim, H.: Presentation attack detection using a tiny fully convolutional network. IEEE Trans. Inf. Forens. Secur. **PP**(C), 1 (2019)
68. Yuan, C., Xia, Z., Sun, X., Member, S., Wu, Q.M.J.: Deep residual network with adaptive learning framework for fingerprint liveness detection. IEEE Trans. Cogn. Dev. Syst. **PP**(C), 1 (2019)
69. Jian, W.E.N., Zhou, Y., Liu, H.: Densely connected convolutional network optimized by genetic algorithm for fingerprint liveness detection. IEEE Access **9**, 2229–2243 (2021)
70. Alshdadi, A., Mehboob, R., Dawood, H., Alassafi, M.O., Alghamdi, R., Dawood, H.: Exploiting Level 1 and Level 3 features of fingerprints for liveness detection. Biomed. Signal Process. Control **61**, 102039 (2020)
71. Yuan, C., Sun, X.: Fingerprint liveness detection using histogram of oriented gradient based texture feature. J. Internet Technol. **19**, 1499–1508 (2018)
72. Chugh, T., Member, S., Cao, K., Jain, A.K., Fellow, L.: Fingerprint spoof buster: use of minutiae-centered patches. IEEE Trans. Inf. Forens. Secur. **6013**(c), 1–13 (2018)
73. Pereira, L.F.A., Pinheiro, H.N.B., Cavalcanti, G.D.C.: Spatial surface coarseness analysis: technique for fingerprint spoof detection. IEEE Int. Conf. Image Process. (ICIP) **49**(4), 13–14 (2016)

Deepika Sharma is presently pursuing her Doctorate in the Department of Computer Science and Information Technology, Central University of Jammu, India, in the field of Biometrics Security. She received her Master degree in Computer Applications (MCA) from Department of Computer Science and Information Technology, Central University of Jammu, Jammu and Kashmir, India, in the year 2017. Her research interests include biometric security, pattern recognition, machine learning, and deep learning.

Arvind Selwal works in the Department of Computer Science and Information Technology, Central University of Jammu, India. He obtained his B.Tech. and M.Tech. degree in Computer Science and Engineering from Kurukshetra University, Haryana, India. He received his Doctorate in Computer Science and Engineering from I.K. Gujral Punjab Technical University, Punjab, India, in the year 2017. His research interests include machine learning, biometric security, digital image processing, lightweight cryptography, and soft computing. He has contributed more than 30 research articles in reputed International Journals/ conference proceedings that are indexed in databases such as Web of Science, Scopus, DBLP, and etc. He has authored a book titled 'Fundamentals of Automata Theory and Computation.' He is an active member of Computer Society of India (CSI). He is undertaking two research projects on computational security from funding agencies like DRDO, New Delhi, India.