Chapter 6

Handcrafted V/S Deep Features

6.1 Introduction

The increasing applications of fingerprint authentication systems demand attention to Presentation Attack Detection (PAD) mechanisms. Fingerprint recognition systems are vulnerable to Presentation Attacks (PAs) created using simple fabrication materials or Presentation Attack Instruments (PAIs). These attacks invade one's own identity by using easily accessible and cheap materials. The current research on software-based PAD relies on various types of representation encodings or features extracted from fingerprint images [4, 6, 36]. In general, these features are used to train a classifier or an ensemble of base classifiers to predict the class label for the given fingerprint image. Representation codings of the fingerprint images are based on either handcrafted or deep learning-based features.

The texture-based feature extraction algorithms have proved to be useful for the liveness detection task. The handcrafted features acquired from these algorithms encode the given fingerprint's texture into a feature vector using a set of filters [152]. LBP [153], LPQ [106] and BSIF [152, 154] are the popular handcrafted feature descriptors that provide a promising solution to the liveness detection task. LBP combines structural and statistical information to process the fingerprint images, LPQ collects the phase information into a histogram to be used as a feature vector and BSIF studies the natural images to learn a fixed set of filters.

Deep learning-based features such as ResNet-50 [76] and VGG-19 [155] have been extensively used for image classification tasks [156–158]. These deep models can be used

for the feature extraction task making use of transfer learning, i.e. the models are pre-trained on ImageNet database [159] and are used for extracting deep feature vectors from input fingerprint images.

The current state-of-the-art PAD systems consider a particular type of features and train classifiers based on them. Focusing only on a specific type of features may result in low robustness. To simulate a real-world scenario, it is essential to develop PAD algorithms that are based on multiple types of features and are prone to variations across acquisition sensors and multiple PAIs. Therefore, in this study, we conduct an extensive set of experiments to provide a comparison between handcrafted and deep features for the fingerprint liveness detection task. In addition, we emphasize the importance of evaluating the performance of liveness detectors under cross-sensor and cross-dataset environments. By doing so, we evaluate the robustness and generalization abilities of the liveness detectors.

The major contributions made by this study are:

1. We present a comparative study on the performance of handcrafted features and deep features for fingerprint liveness detection.

2. We evaluate the performance of various features along with different classifiers under within-dataset, cross-sensor, and cross-dataset environments.

3. We analyze the achieved results and present a generic framework suitable for real-world fingerprint liveness detection.

6.2 Feature Representation

The efficiency of many image classification task depends highly on the representational coding of the images. These codings may either be local textural details or learned features. While handcrafted features are manually designed to overcome specific issues like illumination and variation in scale, deep features identify multiple levels of representation so that higher-level features can represent the semantics of the data. LBP is the most popular handcrafted feature because of its lower computational cost and ability to code fine details [160]. In other comparative studies of fingerprint spoof detection [161], LBP, LPQ and BSIF were observed to be better features than other features such as pores,

valleys wavelet, and curvelet GLCM. Therefore, these features are used for comparison in this study. On the other hand, ResNet-50 and VGG-19 are a popular choice for deep feature extractors. These models have been employed in various past studies and obtained better performance than other feature extraction models [162–164].

6.2.1 Handcrafted Features

1. **LBP:** Local Binary Pattern (LBP) is a local feature descriptor which uses texture as identification information for the perception of images [165]. LBP is a widely used descriptor for its robustness to monotonic illumination changes and low computational complexity. LBP characterizes a local region by considering a N dimensional difference vector d_n between the central pixel p_c and its neighbours p_n, where $d_n = \{p_0 - p_c, p_1 - p_c, .., p_{N-1} - p_c\}$ and N is the sampling number of neighbours [64].

2. **LPQ:** Local Phase Quantization (LPQ) is a local descriptor that embeds all spectrum information of the fingerprint in a small feature vector. LPQ is beneficial to use for its robustness against redundant and blurred information [74]. LPQ uses the blur invariance property of the Fourier phase spectrum. It is based on a short-term Fourier transform (STFT) computed over a rectangular N-by-N neighbourhood N_x for each pixel position x of the image f(x) [106].

3. **BSIF:** Binarized Statistical Image Features (BSIF) projects local image patches linearly onto a subspace to compute binary code for each pixel of the image [107,154]. The binary codes are useful for representing the image regions conveniently by using histograms. BSIF is constructed by binarizing the responses to linear filters learnt from natural images using independent component analysis (ICA) [166].

6.2.2 Deep Features

1. **VGG-19:** VGG-19 is a pre-trained convolutional neural network with 19 weight layers [155]. The model is trained on over a million images from the ImageNet dataset with around 1000 object categories. This variant of VGG-19 architecture contains sixteen convolution layers, three fully connected layers, five max pool layers and a final softmax layer. From the input layer to the last max pooling layer, the network is used for feature extraction, while the rest of the network is used for

classification [167]. In this study, we use VGG-19 for extracting features from the fingerprint images.

2. **ResNet-50:** We use ResNet-50 [76] as one of the deep feature extraction methods for its lower complexity and high performance. ResNet-50 is pre-trained on ImageNet database and was fine-tuned for PAD classification.

6.3 Experimental Study

6.3.1 Feature Extraction

We extracted the handcrafted features using MATLAB. For LBP, LPQ and BSIF descriptors, we obtained feature vectors of size 10, 256 and 4096, respectively, for each image. We used Keras library of Python for ResNet-50[1] and VGG-19[2] implementation, training and validation. The deep feature vector of size 2048 and 512 was obtained for ResNet-50 and VGG-19, respectively, for each image.

6.3.2 Dataset

In this study we use LivDet 2017 [168] and LivDet 2015 [3, 169] datasets for evaluating the performance of various features with different classifiers. LivDet 2017 comprises approximately 3000 live and 3600 spoof images acquired from DigitalPersona, Orcanthus and Greenbit sensors for training and 5100 Live and 6100 Spoof images for testing. We use images acquired only from the DigitalPersona sensor of the LivDet 2015 dataset for cross-dataset evaluation. The complete description of the datasets used in this study is given in Table 6.1.

6.3.3 Classifiers

1. **SVM:** Support Vector Machines (SVMs) are efficient and robust learning models for the classification task. SVMs represent the instances as points in a vector space, separated based on their class. SVMs are widely used for various image classifica-

[1] `https://keras.io/api/applications/resnet/`
[2] `https://keras.io/api/applications/vgg/`

Table 6.1: Description of the LivDet datasets used in this study.

Dataset	Sensor	Train			Test		
		#Live	#Spoof	Fabrication Materials	#Live	#Spoof	Fabrication Materials
LivDet 2017	Digital Persona	999	1199	Body double, ecoflex, woodglue	1692	2028	Gelatin, latex, liquid ecoflex
	Orcanthus	1000	1180	Body double, ecoflex, woodglue	1700	2018	Gelatin, latex, liquid ecoflex
	Greenbit	1000	1200	Body double, ecoflex, woodglue	1700	2040	Gelatin, latex, liquid ecoflex
LivDet 2015	Digital Persona	1000	1000	Ecoflex, gelatin, latex, woodglue	1000	1000	Ecoflex, gelatin, latex, woodglue

tion applications, like fingerprint liveness detection [4, 5, 170], remote-sensing image classification [171], etc.

2. **Bagging:** Bootstrapped Aggregating (Bagging) is an ensemble-based classifier that works on a group of base classifiers. These base classifiers are trained on a bootstrapped version of the original training data. The final prediction is made by fusing the base classifiers' individual outcomes using majority voting [172, 173].

3. **Random Subspace:** Random Subspace is an ensemble-based method which constitutes a pool of classifiers by selecting subset of the feature vector pseudo-randomly [78].

4. **A-Stacking:** A-Stacking is an adaptive classifier based on ensemble learning. It focuses on generating a set of disjoint classifiers by considering the properties of data [6]. We use the same choice of base classifiers and clustering algorithm as used by [6], i.e., SVM, Voted Perceptron and Random Forest as base classifiers and K-means with $k = 3$ as the clustering algorithm.

6.3.4 Experimental Protocol

Accuracy: Accuracy is a general performance metric to test the correctness of the classification and is defined as:

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

where TP= number of Live fingerprints classified as Live; FP= number of Live fingerprints classified as Spoof; FN= number of Spoof fingerprints classified as Live; and, TN= number of Spoof fingerprints classified as Spoof.

APCER: In applications like spoof fingerprint detection, where false negatives (FN) have a huge impact on the performance and the cost of misclassification is high; it is essential to report the performance based on FN. Therefore, we report the Attack Presentation Classification Error Rate (APCER), which is defined as:

$$APCER = \frac{FN}{FN + TN}$$

In other words, APCER is defined as:

> "proportion of attack presentations using the same PAI species incorrectly classified as bonafide presentations in a specific scenario".

Category-1: In this category, we evaluate the performance of various classifiers trained on the handcrafted and deep features under the within-dataset environment, i.e., the models are trained and tested on the instances captured from the same fingerprint sensor. Since LivDet 2017 was assembled for evaluating cross-material generalization ability, the test sets contain fingerprint images fabricated using new materials that are not used in the training set. Therefore, Category-1 inherently tests the models' robustness to the cross-material environment.

Category-2: This category is designed to demonstrate the performance under the cross-sensor environment. In this category, the models are trained and tested on images acquired using a different sensor. Category-2 is useful in evaluating the sensor-generalization ability of the models, which plays an important role in real-world scenarios.

Category-3: In this category, the performance of the model is evaluated under the cross-dataset environment. The models are trained on LivDet 2015 and tested on LivDet 2017 and vice-versa. As two different datasets are used for training and testing, Category-3 is useful for cross-dataset generalization.

6.3.5 Results

Table 6.2 shows the result under the within-dataset category, where the models are trained and tested on fingerprint images acquired using the same sensor in the same dataset. It is evident from Table 6.2 that ResNet-50 outperforms all other feature types by $\sim 3\%$ on Orcanthus sensor. Among handcrafted features, BSIF is the most efficient, and its performance is close to VGG-19.

In our experiments, for the images acquired from the Greenbit sensor, BSIF was most suitable, and it outperformed deep features by $\sim 3-4\%$. On the Digital-Persona sensor, BSIF handcrafted features are the most efficient as their performance is better than all of the deep features by $\sim 2\%$.

Table 6.3 shows the results obtained using various features along with different classifiers under the cross-sensor environment. For open-set evaluation, we train and test the models on images acquired using different sensors (e.g., training on Orcanthus and testing on DigitalPersona). As shown in Table 6.3, for the Orcanthus-DigitalPersona combination, VGG-19 outperforms all other feature types in terms of accuracy. BSIF obtained the lowest APCER. Also, LBP is the most efficient handcrafted feature in terms of accuracy. For the Greenbit-DigitalPersona combination, BSIF outperforms both the deep feature types and its handcrafted counterparts by a huge margin ($\sim 5-15\%$). LPQ is the most efficient in terms of APCER. For the Orcanthus-Greenbit combination, VGG-19 is the most efficient in terms of accuracy and outperforms the handcrafted features but obtains low APCER.

Table 6.4 shows the performance of various features under the cross-dataset environment, e.g., training on LivDet 2015 and testing on LivDet 2017. ResNet-50 performs most efficiently and outperforms the handcrafted features by the margin of $\sim 4-9\%$ when the models are trained on LivDet 2015 and tested on LivDet 2017. On the other hand, when the models are trained on LivDet 2017 and tested on LivDet 2015, BSIF is the most efficient type of features in terms of accuracy and APCER.

Table 6.2: Performance evaluation of hand-crafted and deep features in combination with different classifiers under Category-1.

Orcanthus								
Feature	SVM		Bagging (SVM)		RSM(SVM)		A-Stacking	
Type	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER
LPQ	82.73	0.22	82.33	0.22	83.22	0.2	82.87	0.21
LBP	84.1	0.12	84.13	0.11	79.93	0.17	83.83	0.13
BSIF	87.76	0.13	87.79	0.14	87.84	0.13	86.9	0.15
ResNet-50	90.16	0.12	90.67	0.11	90.53	0.11	90.91	0.10
VGG-19	87.29	0.17	86.62	0.18	87.46	0.17	87.81	0.16
Greenbit								
Feature	SVM		Bagging (SVM)		RSM (SVM)		A-Stacking	
Type	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER
LPQ	89.04	0.15	89.14	0.15	87.75	0.17	89.84	0.12
LBP	77.89	0.08	78.56	0.08	78.45	0.07	79.06	0.13
BSIF	92.11	0.07	91.68	0.07	91.74	0.07	91.79	0.07
ResNet-50	87.11	0.12	87.59	0.12	88.56	0.11	86.28	0.12
VGG-19	87.30	0.11	87.21	0.10	88.32	0.09	87.65	0.10
Digital Persona								
Feature	SVM		Bagging (SVM)		RSM (SVM)		A-Stacking	
Type	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER
LPQ	88.74	0.15	89.19	0.13	89.92	0.12	90.19	0.11
LBP	78.41	0.09	78.06	0.09	74.03	0.14	80.35	0.17
BSIF	92.2	0.06	92.12	0.06	92.04	0.05	92.15	0.05
ResNet-50	90.87	0.10	90.84	0.09	90.97	0.09	90.31	0.11
VGG-19	80.59	0.26	81.48	0.25	80.78	0.26	80.24	0.27

Table 6.3: Performance evaluation of hand-crafted and deep features in combination with different classifiers under Category-2. The experiments are performed by considering different sensors for training and testing and viceversa. The average of both experiments is reported.

Orcanthus- Digital Persona								
Feature	SVM		Bagging (SVM)		RSM (SVM)		A-Stacking	
Type	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER
LPQ	54.02	0.17	52.12	0.27	54.50	0.27	62.14	0.22
LBP	59.94	0.16	60.82	0.07	57.31	0.39	67.52	0.10
BSIF	53.24	0.03	53.53	0.02	53.63	0.02	64.64	0.08
ResNet-50	47.23	0.27	47.33	0.27	51.10	0.28	69.91	0.14
VGG-19	65.29	0.33	63.24	0.29	65.60	0.44	69.61	0.35
Greenbit- Digital Persona								
Feature	SVM		Bagging (SVM)		RSM (SVM)		A-Stacking	
Type	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER
LPQ	73.01	0.08	74.07	0.08	70.78	0.01	70.17	0.04
LBP	54.92	0.39	56.52	0.35	45.52	1.00	66.08	0.09
BSIF	84.90	0.19	84.00	0.18	85.36	0.17	80.54	0.06
ResNet-50	69.77	0.11	70.86	0.09	69.71	0.20	74.57	0.18
VGG-19	62.15	0.45	59.70	0.43	62.47	0.44	65.92	0.21
Orcanthus- Greenbit								
Feature	SVM		Bagging		RSM (SVM)		A-Stacking	
Type	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER
LPQ	49.81	0.04	48.87	0.12	50.02	0.53	54.73	0.04
LBP	56.11	0.26	52.84	0.24	55.40	0.29	62.20	0.27
BSIF	53.10	0.19	53.65	0.18	53.99	0.13	61.40	0.07
ResNet-50	55.84	0.31	55.63	0.26	56.54	0.35	70.06	0.07
VGG-19	59.51	0.45	56.38	0.51	59.84	0.41	68.39	0.07

Table 6.4: Performance evaluation of hand-crafted and deep features in combination with different classifiers under Category-3.

Train: LivDet2015, Test: LivDet2017								
Feature	SVM		Bagging (SVM)		RSM(SVM)		A-Stacking	
Type	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER
LPQ	54.46	0	54.46	0	54.46	0	61.5	0.06
LBP	54.54	0	54.54	0	54.49	0	54.54	0
BSIF	55.21	0	55.08	0	55.19	0	58.68	0.04
ResNet-50	61.66	0.04	63.60	0.08	59.02	0.05	69.41	0.03
VGG-19	49.11	0.92	46.21	0.98	50.05	0.88	71.24	0.04
Train: LivDet2017, Test: LivDet2015								
Feature	SVM		Bagging (SVM)		RSM (SVM)		A-Stacking	
Type	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER	Accuracy	APCER
LPQ	68.95	0.08	60.85	0.04	66.05	0.08	63.60	0.04
LBP	55.80	0.43	55.75	0.43	50.00	1.00	55.80	0.43
BSIF	78.45	0.03	79.85	0.04	78.05	0.03	74.85	0.22
ResNet-50	72.80	0.19	72.16	0.15	71.68	0.20	73.48	0.22
VGG-19	69.12	0.17	66.00	0.15	71.36	0.29	69.80	0.28

6.3.6 Discussion

In this section, we analyze the obtained results by considering various features along with different classifiers. A thorough analysis is required to infer the results carefully. We present a pictorial interpretation of the comparative results of handcrafted and deep features in Figure 6.1.

It is evident that BSIF outperformed other handcrafted features in Category-1. In Category-2, LBP is better in two of the three cases (i.e., Orca+Green and Orca+Dig) and BSIF in the remaining (i.e., Green+Dig). In Category-3, BSIF outperforms other handcrafted features on the 2017-15 combination by a significant margin, and on 2015-17, it lags behind LPQ by a small margin. LBP and LPQ describe each pixel's neighbourhood by a binary code obtained by first convolving the image with a manually pre-defined set of linear filters and then binarising the filter responses. However, BSIF computes a binary code for each pixel by linearly projecting local image patches onto a subspace, whose basis vectors are learnt from natural images via independent component analysis, and by binarizing the coordinates on this basis via thresholding. Therefore, BSIF could have better efficacy over other descriptors in image representation.

Whereas ResNet-50 promises to be a more suitable type among deep features for this particular application, it performs better than VGG-19 in two of the three cases (i.e., Digital and Orcanthus sensors) in Category-1, in one of the three cases under Category-2 and both the cases for Category-3. The superiority of ResNet-50 could be due to its unique architecture involving skip connections that avoid the problem of vanishing gradients. We evaluated these features' performances under various complex environments to test the robustness and observed a significant drop in performance during open-set evaluation. The performance drop is consistent across all the studied features, but BSIF and ResNet-50 obtain the lowest drop and perform reasonably well.

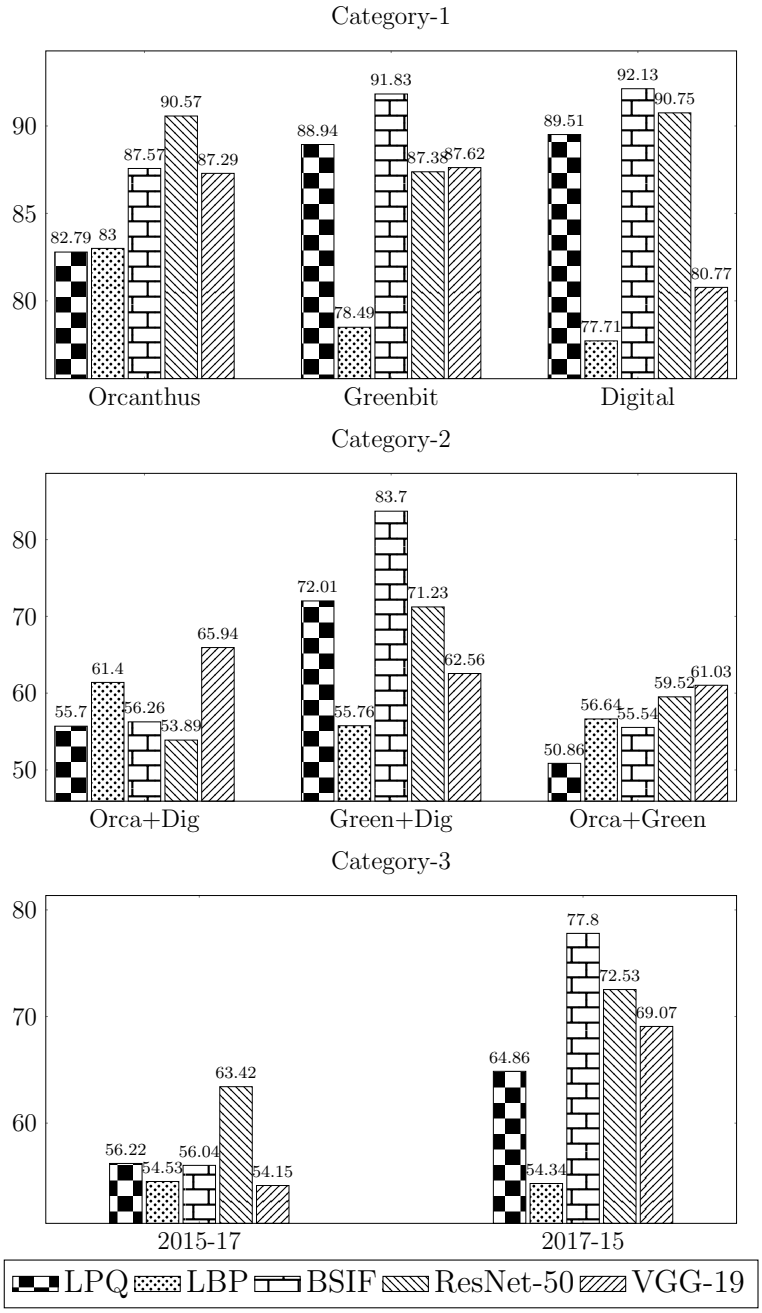

Figure 6.1: Accuracy comparison of various handcrafted and deep features under three environments. The accuracy is averaged across various classifiers used in the study.