Research Highlights (Required)

It should be short collection of bullet points that convey the core findings of the article. It should include 3 to 5 bullet points (maximum 85 characters, including spaces, per bullet point.)

- Deep Transfer Learning architectures are compared against shallow learning image descriptors under the same conditions.
- Deep transfer learning for fingerprint liveness detection is a viable way.
- XAI shows that deep transfer learning extracts complex nonlinear patterns able to discern minutiae for liveness detection.

Pattern Recognition Letters
journal homepage: www.elsevier.com

"A comparative study of shallow learning and deep transfer learning techniques for accurate fingerprints vitality detection"

Donato Impedovo [a]*, Vincenzo Dentamaro [a], Giacomo Abbattista [a],Vincenzo Gattulli [a], Giuseppe Pirlo [a]

[a]"University of Bari - Department of Computer Science, Via Edorado Orabona - 4, Bari 70121, Italy"

ABSTRACT

This work inspects deep learning architectures and shallow learning techniques to determine whether the image of a fingerprint is real (Live) or not (Fake). It is known that Deep Learning techniques deliver, in general, good accuracies being able to automatically extract relevant patterns, at the same time, it is also known that these algorithms require large amounts of data. For this reason, transfer learning aims to transfer the knowledge learnt over a huge dataset to a new, smaller dataset. In this work, because of the limited size of the LivDet2019 dataset, three well known deep learning architectures such as Inception V3, ResNet50 and NASNet Large have been modified to perform transfer learning from the huge imagenet dataset to the smaller LivDet2019.

The hypothesis at the very basis of this work is that the deep learning architectures trained on the huge imagenet dataset would learn to extract relevant patterns like lines, shapes, curves, jump between curves, etc… Later, the extracted knowledge, is fine-tuned on the LivDet2019 dataset to recognize fingerprint minuities as a non-linear combination of the previously learned patterns.

For sake of completeness, state of art shallow learning image descriptors, finetuned for fingerprint recognition, such as Binarized Statistical Image Features (BSIF), Local Phase Quantization (LPQ) and Weber Local Descriptor (WLD) are used for extracting features from the LivDet2019 dataset. The classification on each of these extracted features is performed both with a linear and non-linear (gaussian) support vector machine.

Accuracies suggest that both shallow learning and deep learning techniques are on par with the accuracies of reviewed works and thus transfer learning in fingerprint liveness detection is a feasible strategy that deserve attention and future research with the aim of increasing fingerprint detection accuracies.

Keywords: fingerprints, liveness detection, livedet2019, Deep Learning, SVM, BSIF, LPQ, WLD, Computer Vision

2012 Elsevier Ltd. All rights reserved.

1. Introduction

In this work, deep learning architectures and shallow learning techniques are used to determine whether the image of a fingerprint is real (Live) or not (Fake). The research deals with liveness detection and the developed techniques are integrated within biometric authentication systems in a variety of applications (Impedovo & Pirlo, 2021). Biometric traits such as fingerprints are particularly prone to spoofing attacks, a situation in which one person forges the identity of another person successfully, by spoofing another person' fingerprint. In this specific case, some of the most famous materials used for artificial impression replicas are Ecoflex, Latex, gelatin and wood glue (Orrù & Et al, 2019). A countermeasure deals with image processing for liveness detection task (Marasco & Ross, 2015).

Deep Learning (DL) has demonstrated to be a real game changer in computer vision, especially for image recognition task. (LeCun & Et al, 2015). In fact, convolutional neural networks are made of several convolutional layers stacked one upon the other and connected with neurons whose activation function is non-linear (usually ReLU), capable of extracting relevant patterns, such as lines, curves, shapes, jumps between curves, sinusoids and so on, going from simple patterns in early layers to more complex patterns (e.g. faces) in later (top) layers.

The main drawbacks of DL, together with high computational power required, is the amount of data required to train the models efficiently. It is known, in fact, that DL techniques requires millions of samples to learn complex patterns and infer predictions accurately. (Aggarwal, 2018)

Because of the limited amount of training data available for fake fingerprint detection, in this work it has been decided to use transfer learning (Tam & Et al, 2018). In transfer learning, a deep neural network architecture and its weights are trained on a big dataset. Once this process is completed, from the trained model, usually, the last classification layer (or latest layers depending on the use case) is removed and few new empty layers are added. The lasts added layers are fine-tuned (trained) on the new smaller dataset by freezing all the former layers weights. This is useful, because former layers learn high level representation of the

* Corresponding author. Tel.: +0-000-000-0000; fax: +0-000-000-0000; e-mail: donato.impedovo@uniba.it

underlying patterns, while the top layers are specialized on the new dataset. (Tam & Et al, 2018)

Given the above, the hypothesis at the very basis of this work is that the deep learning architectures trained on the huge ImageNet dataset containing more than 14 million samples of images (Krizhevsky & Et al, 2017) would learn to extract simple to complex patterns like lines, shapes, curves, jump between curves, up to faces, dogs, birds, cars and so on. Then, this extracted knowledge, is fine-tuned on the LivDet2019 (Orrù & Et al, 2019) dataset with the aim of recognizing fingerprint minuities as a non-linear combination of the previously learned patterns.

A comparison with state of the art non-deep (shallow) learning image descriptors, such as Binarized Statistical Image Features (BSIF), Local Phase Quantization (LPQ) and Weber Local Descriptor (WLD) is presented. The classification on each of these extracted features is performed both with a linear and non-linear (gaussian) support vector machine.

The work is organized as follows: state of the art and related work is presented in Section 2; Used DL architectures and the transfer learning approach is sketched in Section 3. Shallow learning image descriptors are illustrated in Section 4. Experimental Setup is presented in Section 5. Results and reasoning are provided in Section 6. Conclusions and future research directions are illustrated in Section 7.

2. Related work

Fingerprint is, so far, the most used biometric trait thanks also to the ease implementation as well as easy data acquisition, and it is able to provide high recognition rate (Askarin & et al, 2020). Fingerprints as a biometric trait characterize the uniqueness of the models and remain unchanged throughout a person's life (Patel & et al, 2021). This application is also widely accepted by users because it does not present characters that are invasive for the end user (Win & et al, 2020).

Fingerprints are composed of epidermal ridges and valleys that usually run in parallel, forming the set of characteristics of the imprint. In the images obtained through the sensors, these crests are represented by dark lines, while the valleys are bright lines, and each of them is considered unique. Unfortunately, both the crests and the valleys are replicable, and therefore the uniqueness of the fingerprint no longer exists in this case. In particular, its copy takes place artificially through the use of other materials, which make it possible to obtain a rubbery finger and an accurate imitation of the minutiae of the original fingerprint.

There are several studies on this, which aim to build systems capable of detecting real fingerprints from artificial ones (Labati & et al, 2018), (Xin & et al, 2018), (Yuan & et al, Fingerprint Liveness Detection Using an Improved CNN With Image Scale Equalization, 2019).

Mainly there are two methods to detect these types of counterfeits: the first method is through appropriate *hardware* technologies, the second method through *software* technologies (Sousedik & Busch, 2014).

For hardware technologies, fingerprint sensors or "patches" are built to be added to the classic fingerprint readers, which allow additional information to be extracted during authentication, such as *skin distortion* (Antonelli & et al, 2006), the *presence of smell* (Baldisserra & et al, 2006) and *subcutaneous crest models* (Rowe & Sidlauskas, 2006) with *multispectral scanners*.

For *skin distortion*, it is meant that action of the user; this action consists in moving the finger by pressing it against the surface (eg Scanner), thus deliberately exaggerating the distortion of the skin. The approach respects privacy and requires no additional expensive hardware beyond a fingerprint scanner. The results in (Antonelli & et al, 2006) indicate that the approach is a very promising technique for making fingerprint recognition systems more robust against spoofing attempts. The authors in (Antonelli & et al, 2006) failed to find a way to render the system ineffective by finding a combination of techniques and materials that diminish its effectiveness. The disadvantage of this approach is that a better understanding of the relationship between false detection errors and identity verification errors would be needed (Antonelli & et al, 2006).

An innovative method based on the acquisition of *smell* through an electronic nose could be combined with the previous approach. This approach is based on the fact that the smell emanating from the surface of the human skin is different from that emanating from other materials (latex, silicone, gelatin) used for the falsification of fingerprints. Then, by means of an odor sensor, the odor signal is sampled and, by means of an algorithm (Baldisserra & et al, 2006) discriminates the odor of the leather from that of other materials. The results confirm that the approach is able to distinguish real fingerprints from artificial reproductions.

Considering the hardware approaches using *multispectral scanners*, prototype methods and systems for biometric detection have been also proposed (Rowe & Sidlauskas, 2006). The sapproach includes two subsystems: the first is a lighting system that receives diffused light from the textures of the skin, the second is a calculation unit that derives a multispectral image distributed in space from the light received at discrete wavelengths. The latter also compares the derived multispectral image with a multispectral database to identify the individual.

Software solutions, unlike hardware solutions, can usually work with any fingerprint scanned by a fingerprint reader, and therefore without the need to change the authentication systems already available. In this case, image processing algorithms are used to collect additional fingerprint information. There are several approaches in the literature, but they all rely on feature extractions that can be static or dynamic. For the static ones there are anatomical ones, such as the position and *distribution of the pores* (Labati & et al, 2018), (Marcialis & et al, 2010) or the *number of minutiae* (Abhishek & et al, 2015), they can be physiological characteristics, such as *sweating of the skin* (Marasco & Sansone, 2012), (Jin & et al, 2011), or dynamics such as the characteristics based on the *texture and distortion of the skin* (Alice & et al, 2011), (Jia & et al, 2007).

Software approaches based on anatomical features are mainly based on skin pores which can only be detected with high-definition images. These approaches then analyze the positions of the pores that characterize the vitality of the impression. In (Abhishek & et al, 2015) it was considered that in most cases, the pore frequency in the genuine fingerprint is lower than that in the fake fingerprint due to the manufacturing steps required for replication. The results reported in (Abhishek & et al, 2015) encourage the study of this feature which could have future benefits.

Considering the characteristic of software approaches based on *physiological features*, a system was tested in (Marasco & Sansone, 2012) which considered three different optical sensors. The overall system will be faster, because the information is extracted from a single image without scanning the test finger multiple times. The trial has also shown that there is a reduction in performance in the presence of new materials used for spoofing for the liveness detection through the use of a single functionality. This weakness is reduced thanks to the combination of characteristics based on morphology and perspiration. The

disadvantage of this approach is the high error rates achieved, which are still considered high when considering the use in a possible real scenario. Also due to the fact that tolerance with respect to a variation of the same characteristic depends on the discriminating power of the characteristic that receives the attack (Marasco & Sansone, 2012).

In the approaches based on features extracted from the *distortion of the skin* there are several approaches that require a change in the acquisition phase of the fingerprint: in (Antonelli & et al, 2006) the subject rotates the finger during recognition. In (Zhang & et al, 2007) different levels of finger pressure are required during authentication, this of course could be a problem in many real applications. On the other hand, there are also approaches that do not require particular inputs such as (Jia & et al, 2007) where several images are acquired from which characteristics are extracted. In particular, the characteristics describe how the scanning of the contact area changes in size and brightness when the finger is placed on the sensor.

Concerning deep learning architectures, (Plesh & Et al, 2019) uses the features extracted through Inception V3 to carry out the classification within a presentation attack system with fingerprints. The classification is evaluated using features extracted in static, dynamic mode and a fusion of static and dynamic. (Koshy & Mahmood, 2019) compared different deep learning techniques to carry out facial liveness detection. The System with Inception network achieved 100% accuracy in detecting viability with the NUAA dataset. (Fernandes & Et al, 2019) proposed an approach in which the analysis of the attributes is carried out to identify the region of the fingerprint for classification. The approach was tested on the ATVS dataset with different techniques including InceptionV3, ResNet50 and NASNetLarge, always obtaining excellent results.

(Zuo & Et al, 2020) compared different Convolutional Neural Networks to extract facial features and perform face liveness detection. Of the techniques compared on the NUAA dataset, the ResNet50 achieved 97.59% accuracy.

(Arora & Bhatia, 2020) compared different techniques for detect face spoofing including Inception-V3 and ResNet50 obtaining accuracy values above 99%.

3. Deep Learning architectures and Transfer Learning

In this section, three DL architectures previously trained on imagenet dataset are presented. In addition, the used transfer learning is illustrated in subsection 3.4.

The following architectures were chosen depending on the importance in the literature, number of parameters, size and the accuracy achieved on imagenet dataset and reported in Table 1 for completeness.

Table 1. Deep Learning Architectures

Architecture Name	size	Parameters	Top-5 Accuracy on Imagenet
NASNet Large	343 MB	88,949,818	0.960
ResNet 50	98 MB	25,636,712	0.921
Inception V3	92 MB	23,851,784	0.937

3.1. Inception V3

Inception-V3 (Szegedy & Et al, 2016) was originally developed in Google and V3 stands for its third release. The improvements brought by this new version include factorized convolutions, label smoothing, batch normalization and the use of the auxiliary classifier layer. This is used to propagate label information to former layers (lower) of the network.

3.2. ResNet 50

The ResNet-50 (He & Et al, 2016) architecture is composed by 5 "stages". Each stage is composed by sacking a convolution and an Identity block. Each identity and convolutional block have 3 convolution layers resulting in over 23 million trainable parameters.
ResNet brought important innovations into Deep Learning, and it is one of the most used architectures at time of writing. That is, because it introduced two major breakthroughs in computer vision:

1. It mitigates the "vanishing gradient" problem by reinjecting, from time to time, information to the flow.
2. It gives the network the possibility to learn the identity function with respect to the previous output. This ensures that the accuracy of learnt patterns in later layers is at least as good as the previous.

3.3. NASNet Large

NASNet Large deep neural network architecture differently from other architectures, is the result of an optimization process which takes the name of Neural Architecture Search (NAS). In NAS the network architecture and its weights are computed using reinforcement learning approach which aims to learn and decide the best choice of number of layers, type of layer and their hyperparameters. Authors (Zoph & Et al, 2018) developed the reinforcement learning algorithm and its reward function with the goal of automatically constructing a convolutional layer (called "cell") firstly trained on the CIFAR-10 dataset. Once the cell architecture was ready, it was later applied to the ImageNet dataset creating a novel architecture as the result of iteratively stacking copies of this cell, each optimized with their own hyperparameters.

3.4. Applied Transfer Learning

The last (top) two layers (the dense and the softmax ones) have been removed from each previously mentioned architecture. A 2D global average pooling layer has been then added followed by one dense layer with 32 neurons and ReLU activation function. Finally the softmax layer has been added to perform binary classification (real vs. fake). Being unknown the number of layers to freeze while fine-tuning, it has been decided to perform various tests each unlocking a different number of layers for each test with a step of 10% of the total layers. Thus, the first test will have just the last layer to be trained, the second test the 10% of layers as trainable, the third the 20% of layers as trainable and so on. The validation test is set to 15% of the training set. Labels are one-hot encoded.

4. Shallow Learning Image Descriptors

In this section, three different image descriptors, namely BSIF (Binarized Statistical Image Features), WLD (Weber Local Descriptor) and LPQ (Local Phase Quantization) are used to extract relevant features, later used for classification aims. For the classification, linear and non-linear (Gaussian) Support Vector Machine (SVM) has been used. The choice of SVM as the only non-deep classification technique is because it generally delivers high accuracy on these tasks, it is robust against the "curse of

dimensionality" problem which affects these kind of datasets (Ding & Ross, 2016) and because it has few hyperparameters to tune making the model less prone to overfitting and it makes the experiment highly reproducible. (Zhang & et al, 2019), (Topcu & Erdogan, 2019), (Merchant & et al, 2018), (Li & et al, 2018).

In fact, as will be shown in Section 5, SVM has been trained with default hyperparameters reaching on-par accuracies with DL techniques.

4.1. BSIF

BSIF computes a string of binary code for the pixels of a given image. The code value of a pixel is treated as a local descriptor of the intensity model of the image in the environment surrounding the pixel. In addition, the histograms of the pixel code values allow to characterize the properties of the textures within the sub-regions of images.

In general, this approach offers good features for image processing. Furthermore, statistical independence provides the justification for the proposed independent quantization of the response vector elements. Therefore, expensive vector quantization is not necessary (Ghiani & et al, 2013).

4.2. LPQ

In the LPQ model, the descriptor uses locally computed phase information in a window for each image position. The phases of the four low-frequency coefficients are de-correlated and uniformly quantized in an eight-dimensional space. Next, a histogram of the resulting code words is created and used as a function in classifying textures. Ideally, the low-frequency phase components are shown invariant with respect to the symmetrical center blur. The method is still highly insensitive to blur at this point, although this ideal invariance is not fully achieved due to the finite size of the window. Furthermore, the method is also invariant to uniform changes in illuminations, since only the phase information is used.

Local Phase Quantization is similar to Binarized Statistical Image Features in that it can be used in texture recognition tasks in a similar way to BSIF. However, Binarized Statistical Image Features instead of heuristic code constructions, is based on natural image statistics and this improves its modeling capability. (Yuan & et al, Fingerprint liveness detection based on multi-scale LPQ and PCA, 2016).

In (Yuan & et al, Fingerprint liveness detection based on multi-scale LPQ and PCA, 2016) it is shown that multi-resolution analysis is a useful method for extracting texture characteristics when capturing the vividness of fingerprints. The method is not yet fully studied and requires countless future tests, at the moment the inexperience of the method is to be considered a disadvantage.

4.3. WLD

WLD is based on the fact that the human perception of a model depends not only on the change of a stimulus (such as sound, lighting) but also on the original intensity of the stimulus. WLD consists of two components: differential excitation (ε) and orientation (Θ).

For a given pixel, the Weber Local Descriptor (WLD) differential excitation component is calculated based on the ratio of two terms: one is the relative intensity difference of a current pixel with respect to its neighbors, the other is the intensity of the current pixel. With the differential excitation component, an attempt is made to extract local salient patterns in the input image. Furthermore, for each pixel of the input image, two components of the WLD function are computed.

By combining the WLD per pixel function, it is possible to represent an input image (or image region) with a histogram, called WLD histogram. Therefore, WLD is a dense descriptor. For a given image, the two components are used to build a concatenated WLD histogram: a target pixel is selected in the fingerprint and the difference in color intensity between the target pixel and its neighbors is computed (Gragnaniello & et al, 2013).

In (Gragnaniello & et al, 2013) WLD has been combined with other descriptors, with better results if the combined descriptors take into account the characteristics of the images in the other class. An advantage could be the combination of WLD with Local Phase Quantization (LPQ) where the results could improve significantly (Gragnaniello & et al, 2013).

.

5. Experiment setup

5.1. Dataset

The dataset here adopted is the LivDet2019 (Orrù & Et al, 2019). The LivDet2019 dataset is composed of two subsets obtained from the LivDet2017 dataset, namely "Orcanthus Certis2 Image" and "Green Bit DactyScan84C", plus another subset obtained from the LivDet2015 dataset, or "Digital Persona U.are.U 5160".

The sizes and resolutions of the images captured by the different subsets are not the same. Table 2 shows the characteristics of the subsets.

Each subset is divided into two parts, a training part and a testing part. Since only the training parts related to all three subsets were available, the available subsets were divided into training and testing using a split of 50% -50% using different subjects per train-test split. This means that different people were inserted in training and testing sets. This is to avoid bleeding of information: e.g. different fingerprints coming from same person in both train and test.

The subsets contain both real fingerprints and fake fingerprints, the latter made using different materials. Table 3 shows the materials used for the different subsets and the number of registered fingerprints; pictorial representation of fingerprint material is reported in Figure 1.

Table 2. Characteristics of the subsets

Subsets	Model	Resolution [dpi]	Image Size [px]	Format	Type
Green Bit	Dacty Scan84C	500	500x500	BMP	Optical
Orcanthus	Certis2 Image	500	300x419	PNG	Thermal swipe
Digital Persona	U.are.U 5160	500	252x324	PNG	Optical

Table 3. Number of samples for each subset and materials

Subsets	Live	Wood Glue	Ecoflex	Body Double	Latex	Jelly
Green Bit	1000	400	400	400	-	-
Orcanthus	1000	400	400	400	-	-
Digital Persona	1000	250	250	-	250	250

Figure. 1. Types of fingerprints present in the LiveDet2019 dataset

5.2. Experiment

For the deep learning architectures, all official subset training and test dataset splits were respected for practical state of art comparison. No data augmentation was performed. For transfer learning two datasets were used: imagenet (Krizhevsky & Et al, 2017) with over 14 million images and ATVS (Galbally J., 2014). All used networks were trained in a end-to-end fashion on these datasets. The models created, are then used as pre-training and fine-tuning.

In order to use imagenet as pre-training dataset, fingerprint images were imported as color rgb images. That is because the pre-pruned weights on imagenet accept only rgb-colored images as input. The ATVS dataset contains instead 3168 black and white images, thus no rgb conversion was performed and networks were trained on b/w images.

The validation is performed on 15% of the training set. For all architectures Adam optimizer was used. All models were fine-tuned (trained in transfer learning fashion) for 100 epochs.

Concerning shallow learning techniques, the same train-test split of each original subset was respected for comparison purposes. Each test was repeated 10 times and the average results were reported. Extracted features, from each descriptor, were normalized with z-score before feeding input to the SVM classifier respecting train-test separation and avoiding bleeding of information from the training set to the test set. No feature selection, nor dimensionality reduction technique was used. The SVM was trained with C=1.0. When Gaussian (rbf) kernel was tested, the gamma hyperparameter was adaptively chosen. SVM was trained in one versus all fashion.

Accuracies, precision, recall and Average Classification Error (ACE) (Tolosana R., 2020) have been reported for each test along with the percentage of unlocked layer for deep neural network trained in transfer learning fashion.

6. Results and Reasoning

6.1. Results

Results reported in Table 4 suggest that both shallow learning and deep learning are performing with high accuracies on all the datasets. More specifically Linear SVM with BSIF delivers high accuracy in almost all the dataset used. Other image descriptors such as WLD and LPQ have oscillating accuracies that strongly depend on the dataset used. On Green Bit dataset the state of art FSB algorithm **(Orrù & Et al, 2019)** is only 0.05 percentual points more accurate than the Linear SVM + BSIF image descriptor used in this work. Deep learning approaches have outperformed the state of the art on some datasets and they are in line with the state of the art on Green Bit dataset. For Green Bit, when it is fine-tuned unlocking 60% of all its frozen layers, it delivered the 97.27% of accuracy. This result is still lower than state of art. The drop in performance with respect to the Linear SVM with BSIF may be due to the conversion of the image (originally one channel) to RGB and a higher reduction in size, with respect to the images contained in Orcanthus and Digital Persona. More specifically, the GreenBit images are 500x500 while the network accepts 224x224 as input, this is a 50% (approximately) reduction in size, which implies that some information is lost. The images of the other datasets, Orcanthus and Digital Persona, are 419x300 and 324x252, respectively, thus the reduction is lower if compared to the previous case. In fact, NasNet Large trained with 70% of layers unlocked achieved an accuracy of 95.70% on Digital Persona, which is higher than the state of the art, while Inception V3 with 50% of unlocked trainable layers achieved the state of art on Orcanthus dataset performing 99.17% of accuracy. These results suggest that deep learning techniques with transfer learning are applicable, but their accuracy suffer from the various image resize and compression due to pre-processing.

Concerning Shallow Learning, the overall training time varies between 10 and 30 seconds, the evaluation in most cases is even under a second or at most 4 seconds. Training and testing times are not comprehensive of feature extraction phase, which lasted a few seconds. The times are computed on a per test basis (entire set of files used for training and testing). Performance previously reported are referred to Python 3.6 and sklearn running on a quad core Intel Xeon E3-1505M processor with Windows 10. Concerning Deep Learning, the training time ranges between 10 - 12 minutes for ResNet50 V2 and Inception V3, while for NASNet Large, due to its large size, the training takes approximately 30 minutes. Evaluation with ResNet50 and Inception V3 takes between 5 and 10 seconds. With NASNet Large, however, it takes between 25 and 40 seconds. Deep Neural Networks were trained using Keras with Tensorflow 2 backend on Google Colab with Nvidia P100 GPU having 3584 cuda cores. Although execution times are compliant with real time processing, it is worth noting that the system is a non-optimized prototype.

Table 5 shows comparison with some state of the art techniques. It is important to state that, even if almost all reviewed works performed 50/50 split ratio with user split, randomness could lead some variation in the used training and test datasets. I tcan be observed that Deep Learning techniques have fewer oscillating accuracies with respect to the dataset and for Orcanthus and Digital Persona outperform the state of art accuracies. Another interesting result is that the NASNet Large, the neural network whose architecture was not created by a human being, but is the output of a reinforcement learning process,

in almost all cases delivers the highest accuracy, nonetheless reached the state of art accuracy on Digital Persona.

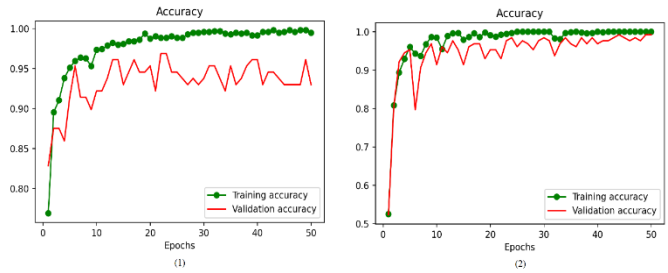

Figure 2. Inception V3 training and validation accuracy on Oracnthus dataset. (1) on the left without transfer learning, (2) on the right with 50% of unlocked layer and imagenet transfer learning.

Figure 2 shows the trend of Accuracy in training and validation on Orcanthus using Inception V3, without transfer learning (on the left) and with 50% layers of the base unlocked (on the right) using imagenet. Some overfittings can be observed in case 1 on the left, on the other hand, it is less pronounced in configuration 2 on the right.. The two configurations obtained, respectively, an accuracy of 96.55% and 99.17%. Thus transfer learning by unlocking

different layers helped even in mitigating overfitting while training.

Table 4. Performances comparison

Model	Dataset	Precision	Recall	Accuracy	ACE (Tolosana R., 2020)	Unlocked trainable layers (%)	Time in seconds
Gaussian SVM + WLD	Digital Persona	0.8907	0.8800	0.8769	0.1015	Not applicable	28
Linear SVM + WLD	Digital Persona	0.8643	0.8584	0.8556	0.1030	Not applicable	23
Gaussian SVM + BSIF	Digital Persona	0.9506	0.9399	0.9395	0.08038	Not applicable	31
Linear SVM + BSIF	Digital Persona	0.9501	0.94	0.9393	0.08040	Not applicable	37
Gaussian SVM + LPQ	Digital Persona	0.943	0.931	0.9289	0.1056	Not applicable	21
Linear SVM + LPQ	Digital Persona	0.9219	0.9139	0.9122	0.1060	Not applicable	32
Gaussian SVM + WLD	Orcanthus	0.9253	0.9231	0.9232	0.0828	Not applicable	24
Linear SVM + WLD	Orcanthus	0.8499	0.8468	0.8466	0.0993	Not applicable	11
Gaussian SVM + BSIF	Orcanthus	0.987	0.9868	0.9868	0.0136	Not applicable	32
Linear SVM + BSIF	Orcanthus	0.9875	0.9872	0.9872	0.0135	Not applicable	22
Gaussian SVM + LPQ	Orcanthus	0.9669	0.9663	0.9663	0.0402	Not applicable	31
Linear SVM + LPQ	Orcanthus	0.9639	0.9636	0.9636	0.0410	Not applicable	25
Gaussian SVM + WLD	GreenBit	0.9181	0.9114	0.9115	0.0733	Not applicable	27
Linear SVM + WLD	GreenBit	0.9056	0.9036	0.9036	0.0745	Not applicable	21
Gaussian SVM + BSIF	GreenBit	0.9927	0.9926	0.9926	0.0078	Not applicable	29
Linear SVM + BSIF	GreenBit	0.9968	0.9968	0.9968	0.0077	Not applicable	23
Gaussian SVM + LPQ	GreenBit	0.9918	0.9917	0.9917	0.010	Not applicable	24
Linear SVM + LPQ	GreenBit	0.9887	0.9885	0.9885	0.013	Not applicable	16
Inception V3	Digital Persona	0.929	0.9401	0.9341	0.0659	50%	994
ResNet 50	Digital Persona	0.9244	0.9540	0.94	0.0620	40%	618
NASNet Large	**Digital Persona**	**0.9507**	**0.9640**	**0.9570**	**0.0430**	**70%**	**2071**
Inception V3	**Orcanthus**	**0.9880**	**0.9972**	**0.9917**	**0.0110**	**50%**	**1088**
ResNet 50	Orcanthus	0.9899	0.9820	0.9872	0.0110	70%	666
NASNet Large	Orcanthus	0.9939	0.9720	0.9845	0.0165	20%	1992
Inception V3	GreenBit	0.9586	0.9720	0.9682	0.0301	60%	988
ResNet 50	GreenBit	0.9502	0.9920	0.9727	0.0256	60%	602
NASNet Large	GreenBit	0.9896	0.9480	0.9718	0.0301	70%	2282

Table 6 shows the overall accuracies on the LivDet 2019 dataset by averaging accuracies for each technique on the three datasets. It can be observed that pre-training using imagenet greatly increases the accuracy of the systems. This is due to the great amount of instances available in Imagenet (over 14 millions) with respect to the 3168 images in ATVS, which were insufficient to extract simple and complex patterns to be reused while performing transfer learning, especially on early layers of the networks.

These results suggest that initial hypothesis of using transfer learning for transferring knowledge learnt over the huge ImageNet dataset on the new small LivDet2019 datasets is effective. For completeness, it should be acknowledged that because of the limited number of samples in the LivDet 2019 datasets (2000-2200 max), accuracies really scrape decimal numbers. Most likely, the use of data augmentation techniques, would further improve accuracy.

In order to understand what the deep learning models have learnt visually, an explainable artificial intelligence technique named LIME (Ribeiro & Et al, 2016) has been used. LIME is the acronym of local interpretable model-agnostic explanations. LIME is a technique that falls within the explainable artificial intelligence field (Samek & Et al, 2019). It is one of the most used local surrogate model technique. In surrogate models, simple explainable models (such as linear regression) are trained to approximate the predictions of the underlaying neural network model by focusing only on local information. In this case, the local surrogate model extracts surrounding pixels of a point and then try to fit them with a simple explainable model that seems to provide the highest correlation with the answer provided by the deep neural network. This technique has been used for image explanation (Samek & Et al, 2019).

Table 5. Performances comparison

Work Name	Algorithm Name	Dataset	Liveness Accuracy [%]
Orrù & et al, 2019	**FSB**	**Green Bit**	**99.73**
This Work	**Linear SVM + BSIF**	**Green Bit**	**99.68**
Zhang & Gao et al, 2020	Score Level Fusion Fingerprint Matching	Green Bit	98.61
Zhang & et al, 2019	Slim-ResCNN	Green Bit	97.81
This Work	*ResNet 50*	*Green Bit*	*97.27*
Zhang & et al, 2019	Slim-ResCNN	Digital Persona	95.42
This Work	**NASNet Large**	**Digital Persona**	**95.70**
Zhang & Gao et al, 2020	Score Level Fusion Fingerprint Matching	Digital Persona	94.30
This Work	Gaussian SVM + BSIF	Digital Persona	93.95
Orrù & et al, 2019	PADUnkFv	Digital Persona	93.63
Orrù & et al, 2019	JungCNN	Orcanthus	99.13
This Work	Linear SVM + BSIF	Orcanthus	98.72
This Work	**Inception V3**	**Orcanthus**	**99.17**
Zhang & Gao et al, 2020	Score Level Fusion Fingerprint Matching	Orcanthus	93.98

Table 6. Overall LivDet 2019 accuracy comparison with various unlocked layers using pre-trained weights from ATVS and Imagenet.

ConvNet base	Unlocked Layers (%)	Accuracy pre-training ATVS (%)	Accuracy pre-training Imagenet (%)
ResNet50 V2	0	69,18	93,62
	20	58,33	94,61
	40	56,47	96,11
	60	55,85	96,04
	70	56,47	82,03
Inception V3	0	64,29	91,84
	20	57,74	95,28
	40	52,49	95,80
	60	53,85	95,72
NASNet Large	0	67,31	94,18
	20	57,65	95,48
	40	53,96	95,66
	70	55,98	96,90

The output of Lime for two images belonging to the Orcanthus dataset for both a Fake and Live fingerprint is reported in Figure 3. Both images were correctly classified as Fake and Live respectively by the ResNet 50 model. It is interesting to see that the same model has focused its "attention" on different areas to check if the input image is fake or not. In facts, green area shows the portions of the image that contributed the most to the correct classification of the image and the red area the portion of images that contributed less. This means that some areas of the image, are more important than others. But this also means that the model could recognize high level complex patterns like the set of curves in the top green area of Figure 3 (b) and minutiae in center and lower right area of Figure 3 (a). This interpretation of what the DL model has learnt, even if not perfect (note the probably green artifact in the right bottom of Figure 3 b) because of the linearity of the surrogate model, suggests that transfer learning in fingerprint liveness detection is an important tool that deserves further deepening. In fact, thanks to transfer learning, the model was able of recognizing fingerprint minuities as a non-linear combination of the previously learned patterns, even if the learning process happened on another dataset.

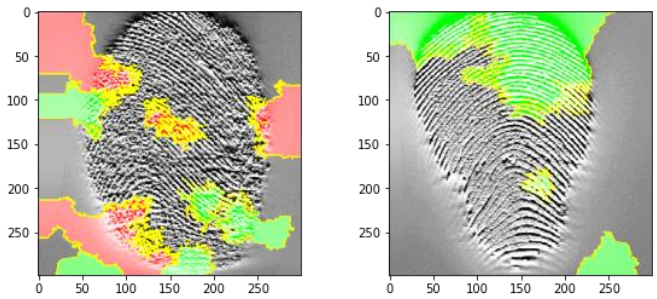

Figure. 3. LIME output for ResNet 50 on Orcanthus dataset. It shows fingerprints correctly identified as Fake (a) and correctly identified as Live (b).

7. Conclusions

In this work standard image descriptors and deep learning techniques have been compared under the same conditions for the aim of fingerprint liveness detection. Both deep learning model trained in transfer learning fashion and shallow learning (image descriptors and SVM classifier) delivered good accuracies on par with the state of the art. In particular for Orcanthus and Digital Persona datasets, respectively Inception V3 and NasNet Large achieved the state of the art accuracies.

This conclusion suggests that the hypothesis of using deep transfer learning for learning simple to complex patterns from a huge dataset, and later bring this knowledge to this small dataset is effective and deserve further deepening. Interesting is also the result of the NASNet Large, this network (not produced by a human being) was capable of achieving almost always the highest accuracies.

In future, ad-hoc deep neural networks models developed with Neural Architecture Search and specifically fine-tuned on fingerprint dataset could develop an ad-hoc network that is trained, in a transfer learning fashion, to focus its "attention" to fingerprint minutiae and thus decrease the chance of a spoofing attack.

References

Abhishek, & et al., 2015. A Minutiae Count Based Method for Fake Fingerprint Detection. *Procedia Computer Science. 10.1016/j.procs.2015.08.061, 58*, 447 - 452.

Aggarwal, C. C., 2018. Neural network and deep learning. *Book, Springer*.

Alice, Y., & et al., 2011. Fingerprint Liveness Detection. *Department of Computing Science Simon Fraser University*.

Antonelli, A., & et al., 2006. Fake Finger Detection by Skin Distortion Analysis. *IEEE TIFS, 1(3)*, 360–373.

Arora, S., & Bhatia, M. P., 2020. Fingerprint Spoofing Detection to Improve Customer Security in Mobile Financial Applications Using Deep Learning. *Arabian Journal for Science and Engineering*, 45:2847-2863.

Askarin, M. M., & et al., 2020. Reduced contact lifting of latent fingerprints from curved surfaces. *Journal of Information Security and Applications, 53*.

Baldisserra, D., & et al., 2006. Fake Fingerprint Detection by Odor Analysis. *In Proc. ICB. Springer*.

Ding, Y., & Ross, A., 2016. An ensemble of one-class SVMs for fingerprint spoof detection across different fabrication materials. *In 2016 IEEE International Workshop on Information Forensics and Security (WIFS)*, pp. 1-6.

Fernandes, S., & Et al., 2019. Directed Adversarial Attacks on Fingerprints using Attributions. *2019 International Conference on Biometrics (ICB), Crete, Greece*, pp 1-8.

Galbally J., M. S., 2014. Image Quality Assessment for Fake Biometric Detection: Application to Iris, Fingerprint and Face Recognition. *IEEE Trans. on Image Processing*, 710-724.

Ghiani, L., & et al., 2013. Fingerprint Liveness Detection using Binarized Statistical Image Features. *IEEE 6th International Conference on Biometrics: Theory, Applications and Systems, BTAS 2013, 6712708*.

Gragnaniello, D., & et al., 2013. Fingerprint liveness detection based on Weber Local image Descriptor. *IEEE Workshop on Biometric Measurements and Systems for Security and Medical Applications, BioMS, 6656148*, 46-50.

He, K., & Et al., 2016. Deep residual learning for image recognition. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778.

Impedovo, D., & Pirlo, G., 2021. Automatic Signature Verification in the Mobile Cloud Scenario: Survey and Way Ahead. *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 1, pp. 554-568, 1 Jan.-March 2021,

Jia, J., & et al., 2007. A new approach to fake finger detection based on skin elasticity analysis. Proc. ICB, 309-318.

Jin, C., & et al., 2011. Fingerprint liveness detection based on multiple image quality features. Information Security Applications, 6513, 281-291.

Koshy, R., & Mahmood, A., 2019. Optimizing Deep CNN Architectures for Face Liveness Detection. MDPI Entropy, 21, 423.

Krizhevsky, A., & Et al., 2017. Imagenet classification with deep convolutional neural networks. Communications of the ACM, 60(6), 84-90.

Labati, D. R., & et al., 2018. A novel pore extraction method for heterogeneous fingerprint images using Convolutional Neural Networks. Pattern Recognition Letters, 113, 58-66.

LeCun, Y., & Et al., 2015. Deep learning. Nature, 521(7553), 436-444.

Li, J., & et al., 2018. Deep convolutional neural network for latent fingerprint enhancement. Signal Processing: Image Communication, 60, 52-63.

Marasco, E., & Sansone, C., 2012. Combining perspiration-and morphology-based static features for fingerprint liveness detection. Pattern Recognition Letters, 33(9):, 1148–1156.

Marasco, E., Ross, A., 2015 A Survey on Anti-spoofing Schemes for Fingerprint Recognition Systems. *ACM Computing Surveys*, 47, 2.

Marcialis, G. L., & et al., 2010. Analysis of fingerprint pores for vitality detection. In Proc. 20th ICPR, 1289–1292.

Merchant, K., & et al., 2018. Deep Learning for RF Device Fingerprinting in Cognitive Communication Networks. IEEE Journal on Selected Topics in Signal Processing, 12(1), 160-167.

Nanni L., B. S., 2019. General Purpose (GenP) Bioimage Ensemble of Handcrafted and Learned Features with Data Augmentation. Arxiv Pre Print.

Orrù, G., & Et al., 2019. LivDet in Action - Fingerprint Liveness Detection Competition 2019. 2019 International Conference on Biometrics (ICB),Crete, Greece, 1-6.

Patel, R. B., & et al., 2021. Biometric Fingerprint Recognition Using Minutiae Score Matching. Lecture Notes on Data Engineering and Communications Technologies, 52, 463-478.

Plesh, R., & Et al., 2019. Fingerprint Presentation Attack Detection utilizing Time-Series, Color Fingerprint Captures. 2019 International Conference on Biometrics, ICB 2019, 8987297.

Ribeiro, M., & Et al., 2016. Why should I trust you?" Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp. 1135-1144.

Rowe, R. K., & Sidlauskas, D. P. (2006). Multispectral biometric sensor. US Patent 7,147,153.

Samek, W., & Et al., 2019. Explainable AI: interpreting, explaining and visualizing deep learning. Springer Nature, Vol. 11700.

Sousedik, C., & Busch, C., 2014. Presentation attack detection methods for fingerprint recognition systems: A survey. IET Biometrics, 3, 219-233.

Szegedy, C., & Et al., 2016. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2818-2826.

Tam, C., & Et al., 2018. A survey on deep transfer learning. In International conference on artificial neural networks. Springer, Cham., pp. 270-279.

Tolosana R., G.-B. M.-G., 2020. Biometric Presentation Attack Detection: Beyond the Visible Spectrum. IEEE Transactions on Information Forensics and Security.

Topcu, B., & Erdogan, H., 2019. Fixed-length asymmetric binary hashing for fingerprint verification through GMM-SVM based representations. Pattern Recognition, 88, 409-420.

Win, K. N., & et al., 2020. Fingerprint classification and identification algorithms for criminal investigation: A survey. Future Generation Computer Systems, 110, 758-771.

Xie J., Z. M., 2019. Handcrafted features and late fusion with deep learning for bird sound classification. Ecological Informatics, 74-81.

Xin, Y., & et al., 2018. Multimodal Feature-Level Fusion for Biometrics Identification System on IoMT Platform. IEEE Access, 6, 21418-21426.

Yadav D., K. N., 2018. Fusion of Handcrafted and Deep Learning Features for Large-scale Multiple Iris. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW).

Yuan, C., & et al., 2016. Fingerprint liveness detection based on multi-scale LPQ and PCA. China Communications, 13, 60-65.

Yuan, C., & et al., 2019. Fingerprint Liveness Detection Using an Improved CNN With Image Scale Equalization. IEEE Access, 7, 26953-26966.

Zhang, S., & et al., 2019. Improving Wi-Fi fingerprint positioning with a pose recognition-assisted SVM algorithm. Remote Sensing, 11(6), 652.

Zhang, Y., & et al., 2007. Fake finger detection based on thin-plate spline distortion model. Advances in Biometrics (LNCS), 4642, 742-749.

Zoph, B., & Et al., 2018. Learning transferable architectures for scalable image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 8697-8710.

Zuo, Y., & Et al., 2020. Face Liveness Detection Algorithm based on Livenesslight Network. 2020 International Conference on High Performance Big Data and Intelligent Systems (HPBD&IS) Shenzhen, China, pp. 1-5.