Earprint touchscreen sensoring comparison between hand-crafted features and transfer learning for smartphone authentication

Jose-Luis Cabra[1]*, Carlos Parra[2], and Luis Trujillo[2]

[1] Fundación Universitaria Compensar, Avenida (Calle) 32 No. 17 – 30. Bogota, 111311, Colombia.
jlcabra@ucompensar.edu.co

[2] Pontificia Universidad Javeriana, Ak. 7 #40 - 62. Bogota, 110231, Colombia.
{carlos.parra, trujillo.luis}@javeriana.edu.co

Received: May 18, 2022; Accepted: July 27, 2022; Published: August 31, 2022

Abstract

The smartphone's lock screen is at a threshold between usability and comfort. For example, some smartphone users prefer not to use the sliding or acceptance call button, but a more secure and efficient way of picking up the phone instead. Others prefer the smoothest interaction possible with their devices for getting quick access to smartphone services. In this paper, from a smartphone authentication point of view, we propose using the touchscreen as an ear shape detector. This approach helps verify the right user for incoming calls, supporting user privacy, as well as avoiding any action approval through a button. In a one-against-all authentication scheme, looking for the best discrimination model, genuine and impostor data are evaluated with two different authentication engines: (i.) Transfer Learning (ii.) Different classifiers are fed by fused hand-crafted features like LBP, HoG, and LIOP. Previous to both authentication approaches execution, the ear shape is extracted by an own heuristic architecture to remove skin-related noises and highlight the region of interest. The classifier results of this paper confirm that Earprint guarantees user verification, reaching an accuracy of 97.7.

Keywords: Smartphone Authentication, Earprint, Capacitive images, Machine Learning, Touchscreen

1 Introduction

Smartphones have integrated computing, multimedia, Internet, sensors, and telephony into one device, selling USD 1,433.86 million units in 2021 [1]. Each smartphone generates and stores sensitive user data, such as access to bank accounts, legal information, contacts, and private messages, among others. For that reason, the smartphone without a locking tool creates a high-security vulnerability, leaving the smartphone open to anyone who wishes to open it.

Locking services take advantage of smartphone sensors (Figure 1) to validate the right user. However, each authentication approach for users, whether from knowledge or identity orientation, has its advantages and limitations [2, 3, 4]. For that reason, we propose to orientate the authentication to specific scenarios where variables such as environment, system behavior, digital friction, or sensor misuse, among others, can provide a better verification scope. This focus does not seek to undervalue current and robust traits, such as face and fingerprint; in contrast, it supports the opportunity of an authentication community where different authentication mechanisms can enrich a different context.

Journal of Internet Services and Information Security (JISIS), volume: 12, number: 3 (August), pp. 16-29
DOI:*10.22667/JISIS.2022.08.31.016*

*Corresponding author: Department of Telecommunication Engineering, Fundación Universitaria Compensar, Avenida (Calle) 32 No. 17 – 30. Bogota, 111311, Colombia. Tel: +57-311-822-86-36

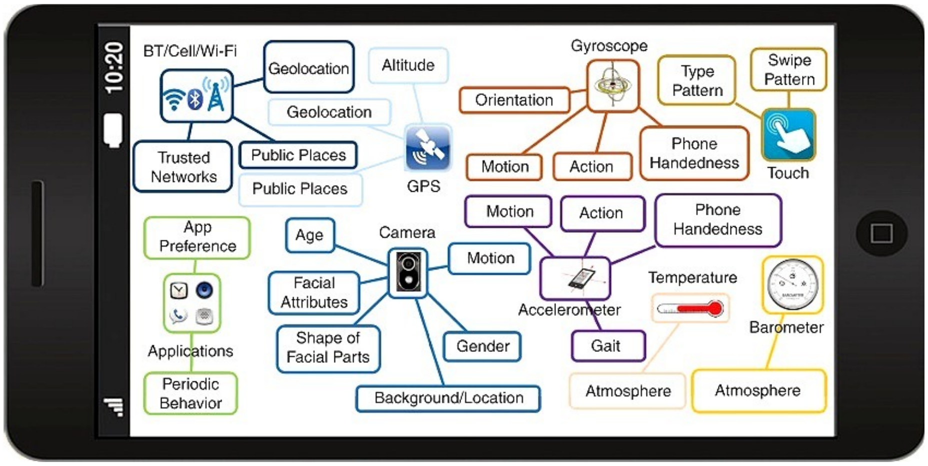


Figure 1: Information possibility from cell phone sensors [5]

All the smartphone's information and services belongs to the user. However, the same user may accept some information vulnerability, looking for fast and easy ways to access mobile phone services. In addition, some default smartphone services could get vulnerability features due to their intrinsic operability. In the case of an incoming call, most of the apps in charge allow anyone to take the call. On the other hand, users prefer the least digital friction possible, for that reason, replacing the sliding or acceptance calling button is a viable option.

The Earprint proposal aims to detect the shape of the ear through the touchscreen to verify the right user between the Genuine or Impostor categories. This approach for an incoming call helps integrate privacy and digital friction reduction requirements because, in one step, it would have the possibility to offer user verification and acceptance trigger, eliminating the need to tap a button. Clearly, as a first approximation, this scheme requires the frontal proximity sensor for ring attenuation and acquisition support, no headphones in the middle, and situates the ear over the capacitive touchscreen.

Previous studies have worked with capacitive images oriented to a person's identity, however, those miss an image preprocessing architecture proposal, because the capacitive sensor is sensitive to the skin contact, acquiring the signal of interest and noise. On the other hand, an open database of capacitive traits is a common limitation in this area; for that reason, current approaches contain scarce populations and samples. As a result, although related work has important classification accuracies, it is advisable to increase the population and samples by person, to robust and evaluate the classifier engine. Finally, for capacitive ears images, a larger comparative study with different machine learning algorithms is necessary, thus providing a general spectrum of the discriminative reach of different techniques that could support future research and researchers.

In the next list, we show the series of contributions of this text.

1. We have constructed a database of Earprint images available for new research.

2. We propose a heuristic architecture to treat ear capacitive images.

3. During the image preprocessing stage, we found that the Laplacian filter helped to discriminate ROI straightforwardly.

4. We propose describing the capacitive sensor image with handcrafted features such as Local Binary Patterns, Histogram of Gradients, and Local Intensity Order Pattern.

5. We have compared the accuracy performance of ten different machine learning algorithms. The feature vector is built into an early fusion configuration that joins LBP, HoG, and LIOP.

6. In a one-against-all test for user verification purposes, the best classifier of our handcrafted feature vector is compared with GoogLeNET and ResNET Transfer Learning approaches.

The structure of this work continues with the related work on touch identity-related studies (Section 2). Next, the article deals with the data acquisition process (Section 3). The image preprocessing treatment step performed with the data is covered in Section 4. Then, in separated sections, the performance rates have been analyzed, by the use of hand-crafted features (Section 5) and Transfer Learning (Section 6) execution, where the results of the different classification approaches are compiled. Finally, in Section 7 we present the conclusion and future work of this research.

2 Related work

Since roughly 1999, capacitive touch pressure sensors began to emerge [6], and until today this technology has spread successfully to the consumer electronics market. From a classical biometric authentication perspective, the touchscreen sensor in smartphone technology by default does not recognize who is interacting with the device [7]. A first approximation of touch-verification is performed by capacitive fingerprinting, studying the user's impedance [8]. Sato et al. expanded the authentication concept to the biometrics of things, including electrodes in different objects using the acquired bio-impedance for user recognition [9]. Sato's objects were a hand-pad, a smartphone and a chair, reaching a result of 84.1% ± 5% with a sample size of 46 people.

Another approach to touch verification is focused on behavioral analysis that identifies user interaction patterns [10], finding attributes such as the user finger orientation [11]. Behavioral touch screen analytics can be oriented towards continuous or transparent authentication [2], taking advantage of the user's habits. Touch tap trace information can be extracted while using the virtual keyboard, any apps, or through operating system touch event APIs.

An innovative alternative in touch verification research is using the smartphone screen to read body parts [12]. This focusing has three orientations: body part [12, 13] and gestures [14] recognition and as a biometric trait reader (Table 1).

From a verification perspective, initially with hand-oriented approaches, Guo et al. [15] used the right hand fingers -without the thumb- as trait using a SVM-Quadratic kernel as classifier to perform both authentication and identification experiments, reaching an accuracy of 99.6% and 94.0%, as appropriate. In addition, this experiment performed a testing with clean and dirty user hands, showing no difference in the classifier performance. Similarly, Tartz et al. implemented hand biometrics [16], using raw touch capacitance data of 40 users obtained an EER of 2.5%, using the normalized correlation coefficient (NCC). Their data were captured during eight sessions, the first three for enrollment and the remaining ones for validation. However, it is not clear for us the amount of samples by person, we provide in Table 1 an approximate value. In the same line of research, Holz et al., Yahoo investigators, worked with body part images, employing the capacitive touch sensor of the smartphone of approximately 6 dpi. They scanned body parts such as the ear, fist, phalanges, palm grip, and finger grip capacitive images as physiological biometric data for smartphone authentication [17]. As a result, the five body parts described above have an identification rate of 99.52% and an FRR of 26.82%. A similar work by Rilvan et al. captured the ears, thumb, and four fingers. They classified and evaluated performance with SVM and Random Forest for identification and verification orientations[18]. Their best authentication rate was SVM with 98.84%, and the best identification score was 97.61% with Random Forest. Finally, Maheshwari et al. for an identification experiment [19] captured ear images of 38 users and achieved precision and sensitivity of 0.87 and 0.60 with KNN as a classifier. Into their refresh rate of 150 ms, they superpose five consecutives images extracting ear features with Scale-invariant feature transform (SIFT) that are upscaled with a CNN Super-Resolution Encoding.

Table 1: **Related work in bodyprint for authentication**

Ref	Year	N	Spl/Person	Focus	Technique	Perf.
[15]	2015	20	200	Fingers	SVM	Accy: 99.6%
[16]	2015	40	∼200	Hand shape	NCC	EER: 2.5%
[17]	2015	12	N/A (864 trials)	Ear, fist, palm and finger grip, phalanges	L2 distance	Accy: 99.52%
[18]	2017	21	$\frac{20}{bodypart}$	Ear, thumb, fingers	SVM → Random Forest →	Accy: 98.84% Accy: 97.61%
[19]	2018	38	40	Ear	KNN	Prec: 0.87 Sens: 0.60
[20]	2022	21	$\frac{20}{bodypart}$	Ear, thumb, fingers	DNN: UASTNet	Accy: 99.77%
Own	2022	32	77	Ear	CNN	Accy: 97.70%

In comparison to the related work and from an authentication point of view, this paper: (1) compares the classification results of the transfer learning and handcrafted features experiments for binary and grayscale images. (2) studies the comparison of the extraction of LBP, HOG, and LIOP handcrafted features with classical machine learning algorithms, selecting the best classifier, including the best combination of feature fusion. (3) performs the classification results comparison of transfer learning nets such as GoogleNet and ResNet. (4) database has a better number of participants and a major proportion of samples by a person. (5) provides an architecture proposal for image treatment.

The touch-screen adoption, like a biometric sensor, could enhance and expand the set of existing tools, to cover several scenarios to perform a successful smartphone authentication [21]. One alternative with this technology is to integrate the ear-printing into the user authentication process during an active call as an aid to verify the right user as a privacy measure.

3 Data acquisition

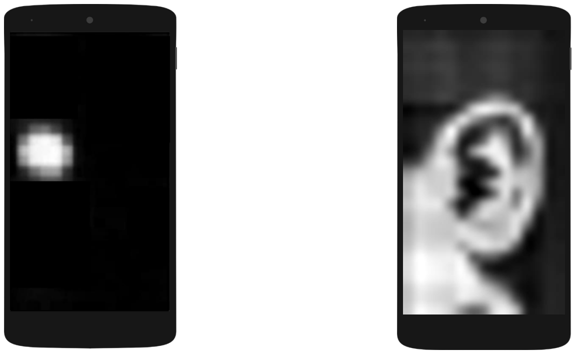
The data acquisition development is composed of two different steps: i.) Smartphone configuration and ii.) Earprint database generation.

3.1 Smartphone configuration

The target of this operation is to get the raw capacitive data of the whole touchscreen, instead of only the points provided by the upper layer of the Android touchscreen interface. Unfortunately, the required data is not directly provided, and for that reason, it is necessary to rebuild the operating system kernel and to modify the touch-driver.

The chosen smartphone in this work is the Nexus 5 (*hammerhead*). Initially, it was necessary to install Cyanogenmod, but this Android distribution was no longer available. So, it was necessary to migrate to LineageOS. This Operating System occupies around 4GB and its first compilations could take several hours. The next move for the installation of the new kernel was to unlock the bootloader. Then, we enabled root capabilities and the program ADB installed to be used in the Windows shell. Then, after installing a fastboot recovery, in the next power-on, Lineage could be installed with the operating system image files. Those image files were obtained after the compilation process.

The process to get the capacitance values must be done prior to the OS compilation. To begin, the touchscreen data and control interface is through I2C communication, which contains a set of commands given in the Synaptics RMI4 (Register Mapped Interface) specification, where each RMI command is located on specific memory pages. All touchscreen driver files are in the kernel space and the different RMI4 functions are not difficult to use and implement. The main RMI4 function to manage in this research is the #54 and it is recommended to check if the touchscreen contains the function requested. A common error is related to finding the specific map space of the registers on page 1 without forgetting to return to page 0, so as to not interfere with the other processes. The main I2C touch-related functions to use are *touch_i2c_write* and *touch_i2c_read*. The stream to return is 405 bytes due to the matrix sensor of 27 x 15. A great guide to understanding this section is in the Github of Isaac Zinda [22].



(a) 1^{st} image (b) Earprint

Figure 2: Images acquired with the smartphone touchscreen

3.2 Earprint database

There is no availability of a database with the number of people and requirements expected for the goals of this research. For that reason, we have created our database composed of 32 individuals. The subject's selection criteria was focused on people of legal age between 20 to 40 years old, physically healthy, without ear lesions or prior ear deformities at birth or due to fire, acid, or other related causes. The database can be accessed through this *link*.

For this experiment in an office environment, the Earprint sampling procedure began with the subjects in sitting position. Then, for smartphone manipulation, the person held the smartphone similarly to answering a call. But the main instruction to the participants was to put the touchscreen to the right ear using their right hand (Figure 3). As required, during the acquisition we asked that the ear area be free of hair, earrings, and other accessories that could interfere with the complete ear measure. At each individual session, we took around 35 captures per participant, striving to have at least three sessions per participant, but not all individual samples were homogeneous. Therefore, from a descriptive value, the general amount of samples by a person is 77 ± 25. Finally, the purpose of each sample was to select only where the ear is, which is the focus of section 4.

4 Image pre-processing

This stage aims to remove most of pixel groups that do not belong to the ear. To the best of our knowledge, related papers to this study do not add their image pre-processing procedures. During the acquisition stage, we realized that the raw-data collection covers the expected region of interest; however, it gathers representative noise too. For that reason, we propose the set of steps in Figure 4 to get the best sample for the classifier.

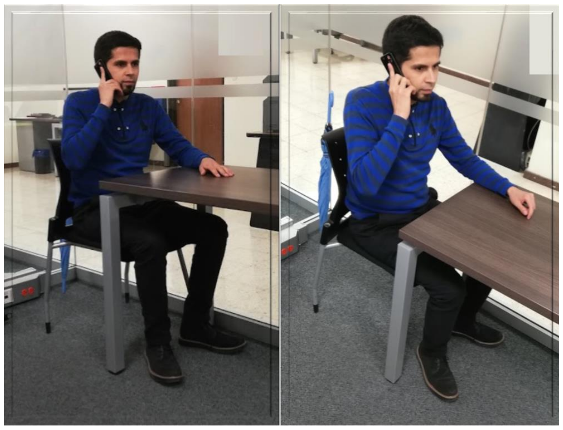


Figure 3: Earprint photo

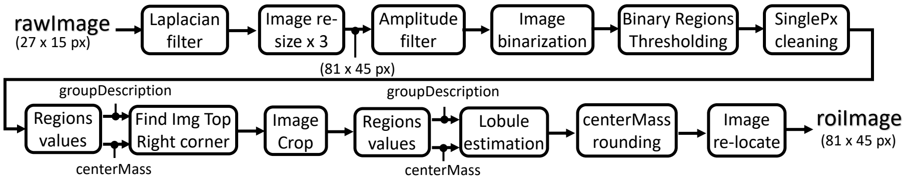


Figure 4: Preprocessing architecture

The Laplacian filter (LF) as the first block of ROI extraction helps to discriminate image conditions like the maintenance and bordering of filled regions and random capacitance values because of low contact with areas outside the ear section. The LF was outstanding compared with filters like: rectangular average, disk average, gaussian, LoG, sobel, prewitt, and unsharp approaches. The Laplacian of an image is defined with the second derivative, as equation 1 states. Using the 3x3 matrix the expression can be expanded as shown in equation 2, allowing its edge detector feature to be applied to the overall image. Finally, the raw image is subtracted with the Laplacian values, getting the first image extraction procedure (equation 3).

$$\bigtriangledown^2 f(x,y) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} \tag{1}$$

$$\bigtriangledown^2 f(x,y) = [f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1)] - 4f(x,y) \tag{2}$$

$$g(x,y) = f(x,y) - \bigtriangledown^2 f(x,y) \tag{3}$$

Then, the image passes through a three times zoom procedure using a bicubic interpolation. As next step, low capacitives values created with the short interaction of ear surrounding areas that creates a fog effect are removed with an amplitude filtering. After the resulting image is binarized using the Otsu's threshold.

Once the image is binarized, it allows one to segment group of pixels identified as regions. We have identified that regions with a large number of pixels mostly correspond to the Earprint trace, for that reason, we applied a pixel group thresholding to remove small regions. With the purpose of smoothing the region's shape, we remove single pixels in horizontal and vertical orientations.

With the areas delimited, we proceeded to extract the image center of mass (c_m) and by region the next attributes: i.) Pixel number; ii.) Region center; iii.) Distance from region center to origin; iv.)

Region weight: $\frac{pixelNumber*100\%}{totalImagePixels}$; v.) Most right-column coordinate; vi.) Most top-row coordinate; vii.) Region distance to center of mass. In our case, the center of mass was calculated with the center of each region and using the #iv parameter, we obtained the average weighted image point.

With the region's group description, we proceeded to get the top-right corner. Then, the image is cropped with the row and column distance between the center of mass and the top right corner (Δd), creating a boundary box of ($c_{mx} \pm \Delta dx, c_{my} \pm \Delta dy$) and deleting pixels outside of the square. Now, due to different regions having been deleted, it is necessary to update again the regions' attributes and the c_m. For the next stage, we intended to find the lobule lowest inflection point, tracing the ear border and removing pixels that do not correspond to the curvature expected in the lobule. As a last step, we prioritized the regions that surround and are closer to c_m, so far and unconnected regions were removed. Finally, we moved the Earprint to the upper-right image corner, now ready for the classification stage.

5 Hand-crafted features study

With every image in the database and with new samples to evaluate, the next step is to generate specific attributes through different handcrafted techniques. Consequently, the set of features to extract in this paper are Local Binary Patterns (LBP), Histogram of Gradients (HoG), and Local Intensity Order Pattern (LIOP). Each feature extraction approach provides a different point of view of the image and could potentially generate enough discriminating information needed for verification.

5.1 Local Binary Patterns

LBP is a famous option to recognize textures in an image [23]. It also helps to appreciate the relationship between the pixel and its neighbors and how that interrelation is predominant in the whole image . LBP contains characteristics like a low computational complexity, and its resistance when there are intensity changes in the image gray color space.

This transformation requires a sweep of the whole image traversing pixel by pixel. In this sense, each pixel during the sweep is taken as the central pixel inside a grid of commonly 3x3 pixels. The next step with the grid values is through a threshold to generate a binary image. The thresholding process consists of taking the central pixel and comparing it with all of their grid neighbors. If the central pixel is bigger than its neighbor, the value will become a binary one in the thresholded grid, otherwise it will be zero and the resulting binary value will be located in the same neighbor position. Once the binary grid is created, each neighbor will contain a weight with a relation of 2^n, n being a number between 1 and 8. Next step, according to Figure 5, is to take the middle-left value of the binary grid as the greater one with $n = 8$ and in counterclockwise order, the value of n will decrease in $n-1$ for each next neighbor. Finally, a point to point multiplication between the matrix's weight and binary, followed by adding all values will provide the final LBP pixel. An example of this procedure is in Figure 5.

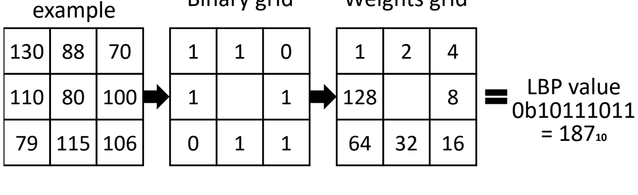


Image grid example

130	88	70
110	80	100
79	115	106

Binary grid

1	1	0
1		1
0	1	1

Weights grid

1	2	4
128		8
64	32	16

= LBP value 0b10111011 = 187_{10}

Figure 5: LBP transformation example

Once the image has been turned to LBP conditions, a histogram is performed with bins between 0 to 255 length. Using the uniform approximation, the histogram length changes to 58. Uniform values correspond with those which contain in their binary form one rising and one falling edge, for example, the number $159_{10} = 10011111_2$. This approximation requires an enhancement, it needs the addition of a block of $[m,n]_{pixels}$ and per block is extracted an histogram. At the same time, this block will recover the whole image with an overlapping criteria to add redundancy and to obtain sturdiness. Finally, all histograms are concatenated to generate the feature vector.

5.2 Histogram of Gradients

HoG utilizes the gradient of each pixel of the image [24]. Commonly, in computer vision this algorithm is well performed looking for human detection in association with SVM. With a predetermined grid, this technique studies the intensity change, taking advantage of the polar coordinates point of view.

The gradient study takes into account the magnitude and orientation with each image pixel. In HoG, the intensity change is done with the adjacent pixels in horizontal and vertical position, taking the difference in x and y axis separately, getting the parameters $\delta(x)$ and $\delta(y)$. The direction and magnitude expressions are in equations 4 and 5

$$\theta(x,y) = \arctan(\delta(y)/\delta(x)) \tag{4}$$

$$g(x,y) = \sqrt{\delta(x)^2 + \delta(y)^2} \tag{5}$$

The image is divided into cells of $m_1 \times n_1$ pixels, extracting from each pixel its gradient. But, as a gradient alone is not useful, it is necessary to group them with the insertion of a histogram analysis. So, each cell in the image is treated with the next stage. Each bin in the histogram will represent the angle distribution according to the number of intervals (N). For example, if $N = 9$, then each interval cover a $\delta(\theta) = 20°$, for unsigned degree intervals of $[0° - 180°]$. On the other hand, the amplitude by bin will be the sum of all $g(x,y)$ presented in the interval. This procedure needs to be done with each cell in the image. At the end, each cell will provide a different histogram required for the next phase.

Now, it is necessary to do two enhancements in the bin amplitude. The first one is considering the contribution of the same pixel when it is near to two different intervals. In that case, it is applied a weight known as $W_n = 1 - (\theta(x,y) - \theta(n))/\delta(\theta)$. The second one is the application of a spatial interpolation evaluating the case when some pixel inside a cell has an influence in another cell due to their closeness. In this situation we applied a second weight known as $W^x_{i,j}$ and $W^y_{i,j}$. Each weight by axis $W^{axis}_{i,j}$ is calculated by $1 - \delta(i,j)^{axis}/\delta(axis)$. $\delta_(i,j)^{axis}$ is the difference by axis among the current pixel and the cell center, over the normalization parameter $\delta(axis)$, which corresponds to the distance between the centers of two adjacent cells. Thereby, the final value in the amplitude bin by gradient will be $h(n) = \sum_{(x,y) W_n * W^x_{i,j} * W^y_{i,j} * g(x,y)}$. Furthermore, it is defined as a block composed of $[m_2, n_2]$ cells. Each block is normalized independently and it will provide a set of $m_2 \times n_2$ histograms. Next step, the block sweeps completely the image with a specified overlapping. Finally, all the histograms are concatenated to generate the feature vector.

5.3 Local Intensity Order Pattern

LIOP is the third feature extraction technique to be used in this paper [25]. This one poses an interesting feature description construction based on the permutation encoding given the pixel intensity order in an image. First, the image is divided into different regions called *local path*. Then, each local path is split up into subregions named *ordinal bins*. Finally, a histogram is built for each ordinal bin after sweeping all pixels.

In the beginning, the image passes through a Gaussian filter σ_p, which prepares it for the relative order moment. Then comes the application of an affine covariant region detector that helps to situate the feature position and calculate approximately the affine shape of its neighborhood. The two most popular detectors are Harris-affine, which detects blob-like structures and the Hessian-affine, which detects corner structures. The discovered regions are normalized in circular areas of a planned diameter for feature extraction. Finally, new Gaussian filtering with sigma σ_n is applied because of the noise produced by interpolation in the normalization step.

Now, to apply the regions division in each one of the local paths, the first step is ordering all the pixels in an ascending order. Then, making use of the Fan et al. algorithm [26], the local patch is equally quantized in B ordinal bins according to the ordered pixel sequence.

To construct the feature vector, with each of the *ordinal bins*, it is necessary to follow the next procedure. First, it is required to define N neighbors to read during the pixel swept. Then, we need to generate a permutation table with the sequence given by $N!$ elements, being the first one the succession from 1 to N, identifying each one with their respective index in ascending order. Thereby, with each pixel during the ordinal bin swept, each neighbor must be labeled with a number from 1 to N. Then, without losing the previous indexation, the neighbors given their pixel intensity must be put in an ascending order. This ordering will give us a sequence number, according to the pixel labels, to be found in the index table, getting the index. Consequently, per analysis point, the LIOP vector of length $N!$ is initialized with zeros and in the position of the obtained index, is added one point. Finally, at the end of the scan, the *ordering bin* LIOP vector will contain all indexes that have been additive scores due to the ordering sequences. Fan et al. suggest to add a weighting function to estimate the degree of dissimilarities between the analyzed pixel and their neighbors. To conclude, the feature descriptor vector will concatenate the LIOP vector of all the ordering bin, of all the local paths of the image.

5.4 Classification approaches and best results

Previous sections have described three different ways to describe the image feature vector that includes texture (LBP), intensity order (LIOP), and pixel magnitude-direction (HOG). All of them have been implemented in this research. Each one provides an enriching point of view, but its discriminating performance is proven by making use of different machine learning approaches.

5.4.1 Methodology

The methodology implemented in this section is guided by Figure 6. This scheme is performed twice, one with the images in binary and the other with gray images.

As the first step, images of the database are described by the techniques LBP, HOG, and LIOP. This set of features vectors is evaluated both individually and together, discriminating the data through different Machine Learning (ML) algorithms. In addition, to evaluate the verification performance between Genuine and Impostor classes, the experiment also will test the hypothesis that individual feature evaluations do not overcome the performance of an early fusion configuration that joins different data categories. The classification during training and test stages uses a 70-30% data distribution, in a one-against-all configuration for each user.

5.4.2 Classification results

Following the proposed methodology (Figure 6), the classification procedure to obtain the best discriminant score contains two moments to compare the results. First, each handcrafted approach is evaluated separately, in contrast, the second moment uses an early fusion evaluation procedure. The set of Machine Learning algorithms considered are Ensemble Subspace Discriminant, Ensemble Subspace Knn,

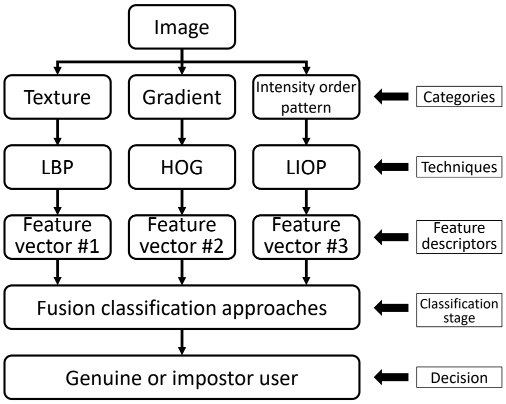


Figure 6: Methodology sequence

Knn Cosine, Knn Fine, Knn Weighted, Linear Discriminant, Logistic Regression, SVM Cubic, SVM Linear, and SVM Quadratic.

The results for individual handcrafted features got its best discriminatory score using HOG and SVM Linear as classifier, reaching 97.7% and 94.6% for gray and binary images. However, LBP and LIOP with SVM Linear classification have close results. The other evaluated algorithms and their values are described in Table 2.

Table 2: **Separated Handcrafted feature evaluation**

Type	**HC conf.**	**ens. Subs. Disc.**	**ens. Subs. Knn**	**Knn Cosine**	**Knn Fine**	**Knn Wht.**	**Linear Disc.**	**Logis. Reg.**	**SVM Cubic**	**SVM Linear**	**SVM Quad.**
B	HOG	91.2	91.8	90.5	85.0	75.1	94.0	91.7	92.3	**94.6**	92.4
G	HOG	95.5	94.1	94.5	89.9	78.5	97.1	96.1	96.5	**97.7**	96.8
B	LBP	91.0	90.4	90.9	53.5	25.6	93.1	91.4	93.2	**94.3**	93.4
G	LBP	95.2	95.0	94.7	74.7	57.6	**97.3**	96.3	96.4	97.2	96.7
B	LIOP	74.6	87.3	87.4	84.5	81.4	87.2	84.0	89.1	**90.1**	88.9
G	LIOP	82.9	89.5	90.6	89.2	86.1	90.1	86.7	92.3	92.5	**92.8**

On the other hand, the classification under fusing handcrafted features has SVM Linear as a leader in the discrimination performance. This algorithm had an accuracy of 97.7% joining HOG with LIOP using gray images and 94.7% with binary images. Other values of the experiment can be found in table 3. However, comparing the results of Table 2 and Table 3, there is no evidence of an improvement after the early fusing, therefore, for this experiment, single descriptor evaluation is a better descriptor than fused configurations. Regarding the initial hypothesis in this section, our data supports that fused feature vectors do not provide a better classification than handcrafted features evaluated separately. As a result, HOG had the best results. The next section will study the performance of Transfer Learning on the dataset.

6 Transfer Learning study

In this project, we propose the use of Transfer Learning to train our model, taking advantage of the image augmentation option and the robustness of the pre-trained model. Then, we will compare the Transfer Learning outcomes with the accuracy obtained with the handcrafted features set.

Two different experiments per Net evaluate the model response when it is developed with binary and

Table 3: **Fused Handcrafted feature evaluation**

Type	HC conf.	ens. Subs. Disc.	ens. Subs. Knn	Knn Cosine	Knn Fine	Knn Wht.	Linear Disc.	Logis. Reg.	SVM Cubic	SVM Linear	SVM Quad.
G	HOG & LBP	96.1	95.2	94.6	81.4	61.8	97.3	96.3	96.9	**97.6**	97.1
B	HOG & LBP	92.5	91.9	91.5	76.9	57.3	**94.4**	92.6	92.9	93.9	92.9
G	0,7HOG & 0,3LBP	96.1	95.2	94.6	81.4	61.8	97.3	96.3	96.8	**97.6**	97.0
B	0,7HOG & 0,3LBP	91.6	92.1	88.8	76.9	57.3	**93.8**	91.9	92.6	93.9	92.5
B	HOG & LBP & LIOP	92.6	92.1	91.6	76.9	57.3	**94.3**	92.7	92.9	93.9	92.9
G	HOG & LBP & LIOP	95.9	95.0	94.7	81.6	61.8	97.3	96.3	96.9	**97.6**	97.0
B	0.2HOG & 0.1LBP & 0.7LIOP	91.5	92.1	89.7	76.8	57.3	93.9	92.3	92.7	**93.9**	92.3
G	0.2HOG & 0.1LBP & 0.7LIOP	96.1	94.8	94.1	81.5	61.7	97.3	96.4	96.9	**97.6**	96.4
G	LBP & LIOP	95.6	94.9	95.0	74.8	57.3	**97.4**	96.0	96.5	97.3	96.7
B	LBP & LIOP	92.1	92.0	92.7	67.1	47.1	94.0	92.2	93.8	**94.6**	93.7
G	0.3LBP & 0.7LIOP	95.0	94.8	94.6	74.8	57.6	97.4	96.1	**96.4**	97.2	96.0
B	0.3LBP & 0.7LIOP	87.2	92.2	89.7	67.5	48.7	93.9	92.1	93.8	**94.6**	93.1
B	0.3HOG & 0.7LIOP	91.3	91.5	90.7	85.2	75.0	93.7	92.0	92.4	**94.7**	92.4
G	0.3HOG & 0.7LIOP	95.4	93.7	94.5	89.9	78.5	97.2	96.1	96.5	**97.7**	96.8
B	HOG & LIOP	91.3	91.4	90.7	85.2	75.0	93.8	91.7	92.4	**94.7**	92.4
G	HOG & LIOP	95.4	93.8	94.5	89.9	78.5	97.1	96.0	96.5	**97.7**	96.9

gray images, being GoogleNET and ResNet the pre-trained models used and compared for this work. The results indicate that ResNet had better accuracy, with 97.2% over 96.2% of GoogleNET, both working with gray images. Table 4 contains the complete configurations and results.

Although the ResNet response is important, it is not yet superior to the score provided by the combination of HOG as feature descriptor and SVM Linear classifier with gray images, as Table 4 states.

These results suggest the possibility of using Earprint as a smartphone authentication option, not to replace current approaches but to incorporate a new tool for the final user en route to a cooperative authentication ecosystem. This authentication option would not require additional hardware, maintaining the same product cost. Additionally, the Earprint authentication service could be extended for a *secure call* option allowing continuous user monitoring.

Table 4: **Results of Transfer Learning models**

Type	NET	Sensitivity	Specificity	Precision	Accuracy
B	GoogleNET	94.0% ± 9%	93.0% ± 8%	86.7% ± 14%	93.5% ± 6%
B	ResNet	93.7% ± 10%	96.6% ± 5%	93.5% ± 9%	96.0% ± 4%
B	HOG	96.4% ± 9%	93.8% ± 8%	87.5% ± 15%	94.6% ± 6%
G	GoogleNET	95.0% ± 9%	95.9% ± 5%	92.4% ± 7%	96.2% ± 3%
G	**ResNet**	**96.4% ± 5%**	**97.3% ± 3%**	**94.5% ± 6%**	**97.2% ± 2%**
G	**SVM Linear(HOG)**	**96.6% ± 5%**	**98.0% ± 3%**	**96.0% ± 7%**	**97.7% ± 2%**

7 Conclusions and future work

Smartphones contain a set of already embedded sensors that can provide support to new forms of device unlocking for authentication services. This work evaluates the ear-shape acquired from the smartphone's touchscreen as a verification trait for evaluating if the current user is a genuine or impostor user; this functionality may then be used for picking up an incoming call. With the use of capacitive images provided by the touchscreen, different strategies to get the best discrimination of the information were applied, as the study of descriptors like LBP, LIOP, HOG and Transfer Learning pre-trained nets. Our best approach achieved an accuracy of 97.7% in a verification context. After the classification process, the data confirm that it is possible to differentiate users by their Earprint. In addition, the HOG descriptor alongside SVM Linear classifier works better than GoogleNET and ResNET Transfer Learning models. In future work, we propose to test the authentication performance stability for (1) non-complete ear acquisition -calling scenario- (2) different human activities and positions (3) out of office environments (4) capacitance changing conditions in the skin, like sweat, for example. Our motivation is to continue to expand smartphone authentication tools to improve user privacy.

Acknowledgments

The authors would like to acknowledge mainly the Fundación Universitaria Compensar, including their Telecommunications Engineering Department, Research Department, English Area, and Bilingual Department, for providing time and support to complete this paper. Also to the cooperation of all partners within the Centro de Excelencia y Apropiación en Internet de las Cosas (CEA-IoT) project. The authors would also like to thank all the institutions that supported this work: the Colombian Ministry for the Information and Communications Technology (Ministerio de Tecnologías de la Información y las Comunicaciones—MinTIC) and the Colombian Administrative Department of Science, Technology and Innovation (Departamento Administrativo de Ciencia, Tecnología e Innovación—Colciencias) through the Fondo Nacional de Financiamiento para la Ciencia, la Tecnología y la Innovación Francisco José de Caldas (Project ID: FP44842-502-2015). An additional acknowledgment corresponds to David Bobby Alvarado.

References

[1] S. O'Dea. Number of smartphones sold to end users worldwide from 2007 to 2021, August 2022. `https://www.statista.com/statistics/263437/global-smartphone-sales-to-end-users-since-2007/` [Online; Accessed on August 10, 2022].

[2] J.-L. Cabra, C. Parra, D. Mendez, and L. Trujillo. Mechanisms of authentication toward habitude pattern lock and ecg: An overview. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)*, 13(2):23–67, June 2022.

[3] J.-L. Cabra Lopez, C. Parra, L. Gomez, and L. Trujillo. Sex recognition through ecg signals aiming toward smartphone authentication. *Applied Sciences*, 12(13):1–16, June 2022.

[4] J.-L. Cabra, D. Mendez, and L. C. Trujillo. Wide machine learning algorithms evaluation applied to ecg authentication and gender recognition. In *Proc. of the 2nd International Conference on Biometric Engineering and Applications (ICBEA'18), Amsterdam, Netherlands*, pages 58—64. ACM, May 2018.

[5] V. Patel, R. Chellappa, D. Chandra, and B. Barbello. Continuous user authentication on mobile devices: Recent progress and remaining challenges. *IEEE Signal Processing Magazine*, 33(4):49–61, July 2016.

[6] W. H. Ko and Q. Wang. Touch mode capacitive pressure sensors. *Sensors and Actuators A: Physical*, 75(3):242–251, June 1999.

[7] M. A. Chowdhury, W. Mciver, and J. Light. Data association in remote health monitoring systems. *IEEE Communications Magazine*, 50(6):144–149, June 2021.

[8] C. Harrison, M. Sato, and I. Poupyrev. Capacitive fingerprinting: exploring user differentiation by sensing electrical properties of the human body. In *Proc. of the 25th ACM symposium on User interface software and technology (UIST'12), Cambridge, MA, USA*, pages 537–544. ACM, October 2012.

[9] M. Sato, R. S. Puri, A. Olwal, Y. Ushigome, L. Franciszkiewicz, D. Chandra, I. Poupyrev, and R. Raskar. Zensei: Embedded, multi-electrode bioimpedance sensing for implicit, ubiquitous user recognition. In *Proc. of the 36th Conference on Human Factors in Computing Systems (CHI'22), New Orleans, LA, USA*, pages 3972–3985. ACM, April-May 2017.

[10] M. Frank, R. Biedert, E. Ma, I. Martinovic, and D. Song. Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication. *IEEE Transactions on Information Forensics and Security*, 8(1):136–148, January 2013.

[11] S. Mayer, H. V. Le, and N. Henze. Estimating the finger orientation on capacitive touchscreens using convolutional neural networks. In *Proc. of the 3rd ACM International Conference on Interactive Surfaces and Spaces (ISS'17), Brighton, UK*, pages 220–229. ACM, October 2017.

[12] I. Guarneri, A. Castorina, S. Battiato, and G. M. Farinella. Pca based shape recognition for capacitive touch display. In *Proc. of the 26th IEEE International Conference on Consumer Electronics (ICCE'13), Las Vegas, NV, USA*, pages 596–597. IEEE, January 2013.

[13] H. V. Le, T. Kosch, P. Bader, S. Mayer, and N. Henze. Palmtouch: Using the palm as an additional input modality on commodity smartphones. In *Proc. of the 36th Conference on Human Factors in Computing Systems (CHI'18), Montreal QC, Canada*, pages 1–13. ACM, April 2018.

[14] R. Wang, C. Yu, X.D. Yang, W. He, and Y. Shi. Eartouch: Facilitating smartphone use for visually impaired people in mobile and public scenarios. In *Proc. of the 37th Conference on Human Factors in Computing Systems (CHI'19), Glasgow, Scotland Uk*, pages 1–13. ACM, May 2019.

[15] A. Guo, R. Xiao, and C. Harrison. Capauth: Identifying and differentiating user handprints on commodity capacitive touchscreens. In *Proc. of the 2015 International Conference on Interactive Tabletops Surfaces (ITS'15), Madeira, Portugal*, pages 59–62. ACM, November 2015.

[16] R. Tartz and T. Gooding. Hand biometrics using capacitive touchscreens. In *Proc. of the 28th ACM Symposium on User Interface Software & Technology (UIST'15), Charlotte, NC, USA*, pages 67–68. ACM, November 2015.

[17] C. H., S. Buthpitiya, and M. Knaust. Bodyprint: Biometric user identification on mobile devices using the capacitive touchscreen to scan body parts. In *Proc. of the 33rd ACM Conference on Human Factors in Computing Systems (CHI'15), Seoul, Republic of Korea*, pages 3011–3014. ACM, April 2015.

[18] M. A. Rilvan, K. I. Lacy, Md S. Hossain, and B. Wang. User authentication and identification on smartphones by incorporating capacitive touchscreen. In *Proc. of the 35th International Performance Computing and Communications Conference (IPCCC'16), Las Vegas, NV, USA*, pages 308–315. IEEE, December 2016.

[19] M. Maheshwari, S. Arora, A. M.Srivastava, A. Agrawal, M. Garg, and S. Prakash. Earprint based mobile user authentication using convolutional neural network and sift. In *Proc. of the 14th International Conference on Intelligent Computing (ICIC'18), Wuhan, China*, volume 10954 of *Lecture Notes in Computer Science*, pages 874–880. Springer Verlag, July 2018.

[20] Md S. Hossain, M. T. Islam, and Z. Akhtar. Incorporating deep learning into capacitive images for smartphone user authentication. *Journal of Information Security and Applications*, 69(1):103290, September 2022.

[21] J. Moreno-Ospina, F. Valencia-Quintero, Omar León-García, M. Steibeck-Domínguez, N. Moreno-Cáceres, and M. Yandar-Lobon. *La industria 4.0, desde la perspectiva organizacional*. Fondo Editorial Universitario Servando Garcés de la Universidad Politécnica Territorial de Falcón Alonso Gamero, 2019.

[22] I. Zinda and M. Goel. RainCheck, August 2015. `https://ubicomplab.github.io/RainCheck/index.html` [Online; Accessed on Jamuary 16, 2022].

[23] M. Pietikäinen, A. Hadid, G. Zhao, and T. Ahonen. *Computer Vision Using Local Binary Patterns*. Springer, 2011.

[24] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. of the 15th IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA*, pages 886–893. IEEE, June 2005.

[25] Z. Wang, B. Fan, and F. Wu. Local intensity order pattern for feature description. In *Proc. of the 12th International Conference on Computer Vision (ICCV'11), Barcelona, Spain*, pages 603–610. IEEE, November 2011.

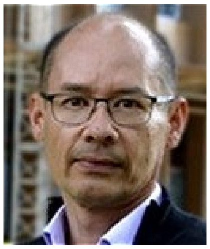
[26] B. Fan, F. Wu, and Z. Hu. Aggregating gradient distributions into intensity orders: A novel local image

descriptor. In *Proc. of the 21th conference on Computer Vision and Pattern Recognition (CVPR'11), Colorado Springs, CO, USA*, pages 2377–2384. IEEE, June 2011.

Author Biography

Jose-Luis Cabra López received the BSc. Eng. in Electronics Engineering in 2011 from the Universidad Nacional de Colombia, Colombia, and the MSc. in Electronics Engineering in 2015 from the Pontificia Universidad, Bogotá, Colombia. From 2020 to the current date, he is a full-time professor of the Telecommunication Engineering Department at the Fundacion Universitaria Compensar. Currently, he is pursuing his doctorate degree at the Pontificia Universidad Javeriana, Bogotá. His research interests include Real Time Operating Systems, Digital Electronics, Embedded Hardware & Firmware Design, IoT, Biomedical Embedded Systems, and Embedded Machine Learning.

Carlos Alberto Parra Rodríguez is a full professor in the Department of Electronics at the Pontificia Universidad Javeriana, Colombia. He is PhD from the Universite De Toulouse III (Paul Sabatier), Francia. Prof. Parra has experience in the application of data science in robotics, perception, intelligent systems, and real-time computer vision. He has published over fifty papers.



Luis Carlos Trujillo Arboleda is an assistant professor in the Department of Electronics Engineering at the Pontificia Universidad Javeriana, Bogotá, Colombia. He received his MSc. (2005) in Telematics from the Universidad del Cauca, Cali, Colombia, his ME. (2008) from the Universidad de Los Andes, Bogotá, Colombia, and his BSc. (1992) in Electronics Engineering from the Universidad del Cauca, Cali, Colombia. Currently he is the general manager of the CEA-IoT. His research interests include Internet of Things (IoT), wireless sensor networks, network design and software defined networks.