Online-Journals.org

iJOE International Journal of Online and Biomedical Engineering

iJOE | eISSN: 2626-8493 | Vol. 20 No. 6 (2024) | OPEN ACCESS

https://doi.org/10.3991/ijoe.v20i06.47881

PAPER

An Optimized Effective Authentication Process for E-Health Application

Hedi Choura(✉), Faten Chaabane, Mouna Baklouti, Tarek Frikha

National Engineering School of Sfax, University of Sfax, Sfax, Tunisia

hedi.choura@enis.tn

ABSTRACT

Because of the availability of more than an actor and a wireless component in an e-health application, providing more security and safety to users of this type of applications is expected. Moreover, ensuring protection of data user available or shared within different services from any security attack becomes an important requirement. In this paper, we are interested essentially in the authentication process, and we propose an improved Landmark-based algorithm as a tool to extract, firstly, key features from analysed faces, and hence to accelerate the authentication operation. The suggested approach beats other state-of-the-art works in terms of accuracy and speed-up attaining time execution constraint, according to experimental evaluations.

KEYWORDS

e-health, intelligent, time execution, authentication, embedded

1 INTRODUCTION

Maintaining the privacy of care subjects requires safeguarding the confidentiality of health data, which is among the most private forms of personal information. As per [7], privacy pertains to granting access authorization or control to any kind of user, including but not limited to doctors, nurses, pharmacies, hospitals, and laboratories. In fact, accessing and sharing patient's information across several actors in an e-health application may lead inevitably to privacy disclosure. Therefore, one of the main challenges anticipated in e-health applications is guaranteeing the safety of information privacy. In this context, different levels of security measures were proposed by researchers ranging from access control to the availability of the shared data, its dependability and flexibility [18], [25]. To cope with these issues, several previous works were striving to improve the access control process. It aims at limiting and controlling the access to shared data by authorized application's users [23]. It consists of essentially three crucial requirements: user identification,

Choura, H., Chaabane, F., Baklouti, M., Frikha, T. (2024). An Optimized Effective Authentication Process for E-Health Application. *International Journal of Online and Biomedical Engineering (iJOE)*, 20(6), pp. 43–60. https://doi.org/10.3991/ijoe.v20i06.47881

Article submitted 2023-12-03. Revision uploaded 2024-01-19. Final acceptance 2024-01-20.

his authentication, and then his eventual authorization. Furthermore, it is undeniable that biometrics is arguably the most well-known means of access control, which has proven efficient authentication [5]. Iris, voice, face, fingerprint, and hand geometry recognition are the many biometrics. Our focus in the proposed work is on face authentication requirement, which aims to guarantee the identity claim prior to access by providing assurance regarding the legitimacy and authenticity of the requested user access. The primary goal of this study is to provide a quick authentication procedure appropriate for embedded e-health systems with a sizable user base. In fact, the suggested solution offers a quick and easy way to authenticate users in order to protect patient data publicly provided on the site. The authentication process consists of two successive levels: a learning phase applied to a set of key features extracted and stored from the faces data base in the first level and then use the same feature extraction algorithm to compare the input face to the existing faces and decide if the user is authorized or not to access. This approach offers a strong identification rate and facilitates an online learning process thanks to the concept of key feature extraction. In comparison to previous facial recognition systems suggested in the literature, it offers an effective decrease in computational expenses of the learning process step. The structure of the paper is as follows: In Section 2, relevant research on face recognition systems—particularly, embedded solutions for secure applications—is reviewed. A general facial recognition system's structure is described in detail in Section 3. We outline the various steps of the suggested Landmark-based authentication procedure in Section 4. The experimental data used to evaluate the suggested framework's learning effectiveness are presented in Section 5. We wrap up in Section 6 with a summary and the range of suggested additional work.

2 RELATED WORK

Because patient data is private and because remote applications, such as e-health ones, are becoming more and more common, researchers' primary concern has been improving user security. Furthermore, unauthorized users and even hackers pose a major risk to security and privacy. As a result, it becomes extremely difficult to provide effective e-health services while maintaining patient data availability, privacy, and validity. Without a doubt, the first prerequisite is access control, which facial recognition effectively ensures. This section examines related work from two perspectives: embedded solutions for some secured applications and access control mechanisms.

2.1 Access control techniques

We provide a classification of the current access control security techniques as the first phase of this work [7]. The majority of the examined methods come from two categories: biometric and non-biometric techniques.

1) Non biometric techniques: The use of a login password to verify a user's identity within an application is one of these approaches' primary contributions. In [20], writers created a password-based authentication technique for remote user access authentication. In [15], a research of telemedicine, ehealth, and wellness applications was suggested to demonstrate that the bulk of wireless sensor-based technique concentrates on engineering related challenges.

2) Biometric approaches: A different class of techniques suggests using facial recognition or fingerprint recognition technology, presuming that these methods can offer a singlemode factor authentication system. Using biometrics is enhancing security for all kinds of actors in health care applications, including patients, nurses, and doctors, as demonstrated by [3].

A combined solution based on a smart card with password authentication scheme utilizing biometrics approach and hash function is proposed in [16] and [21]. This method provided a safe and effective telecare solution. In order to improve security and privacy concerns, [13] summarizes e-health security difficulties and advancements employing biometrics technology applications. In this work, we present a novel facial recognition-based biometric identification access control method. This clever approach can quickly implement an online learning process and achieve high recognition rates.

2.2 Embedded techniques for secured applications

Systems that are both effective with a low error rate and light in terms of execution time and algorithmic complexity can now be implemented thanks to the development of artificial intelligence applications in the field of image processing. There are two types of factors that impact the facial recognition quality: extrinsic and intrinsic. The physical condition of the human face is considered an intrinsic component, whereas extrinsic variables are those that give rise to a desire to alter the face's look. Ageing [11], facial expression [12], pose variation [9], partial occlusion [24], and lighting effects [14] are some of these factors.

Simultaneously, embedded systems have experienced a notable progression, with several platforms emerging as suitable and effective means of executing diverse applications. Certain platforms have been utilized for testing and implementation, while others have made development and the deployment of multimedia application-optimized architectures possible. The Xilinx Zynq UltraScale+ multiprocessor system-on-chip (MPSoC) [31] is one example of a heterogeneous computing system. It allows for the integration of multiple processing units into a SoC in order to accommodate the increasing demands of applications, including power and performance limitations.

Furthermore, FPGA (Field-Programmable Gate Array) platforms' hardware solutions enable the required speedup at a lower power consumption. In comparison to an Application Specific Integrated Circuit (ASIC), performance and power advantages can be achieved by coupling a programmable Central Processing Unit (CPU) with an FPGA-based accelerator. We cover some of the current methods for applied embedded face recognition in this research. We might mention the Viola Jones method's implementation as one example of an embedded face detection algorithm. Thanks to the suggested parallelism, the embedded experimentation of this algorithm based on a mixed HW/SW architecture in [17] allowed to have a more stable system than a standard PC in the case of an image with numerous faces.

Therefore, the implementation of a massively parallel architecture on FPGA becomes faster with an equivalent stability. The technology is undoubtedly less efficient than a graphics processing unit (GPU), but it uses less electricity. [6] described the usage of a heterogeneous architecture based on parallel processors with a hardware intellectual property.

The program is implemented utilizing two Advanced RISC Machines (ARM) processors on a multi-core Zynq platform. Compared to a traditional two ARM processor architecture, the face identification application yielded a 7.8-fold increase

in performance. A brand-new face detector designed for on-board systems, named EagleEye, has been proposed in [22].

Using convolution factorization, this effective method creates a network with dispersed connections based on the building of face detection networks with low computing cost and adequate capacity. With the input of VGA resolution, EagleEye operates at 21 frames per second on an embedded device based on the ARM Cortex-A53 (Raspberry Pi1 3b+ [30]) with the best accuracy compared to methods with the same order of computational complexity. On a different note, some people would rather design a processor specifically tailored to image processing applications using FPGAs. A particular heterogeneous multiprocessor architecture is realized using FPGAs.

This application, which is based on the GPU principle, enables the creation of a system that is competitive not just in terms of performance but also in terms of energy and space usage. This technology allows for a performance of 328 frames per second. The authors of [8] describe the development of a face recognition system based on a Raspberry Pi that makes use of traditional face detection and recognition methods including PCA (Principal Component Analysis) and Haar detection. On the other hand, they require ten trials at least to identify the correct face, meaning that one iteration of their algorithm can achieve 10% accuracy.

Additionally, authors only look for one face in their photos. The employed algorithm is traditional; it is not predicated on any method of artificial intelligence. In terms of speed and accuracy, [19] compares three distinct algorithms—PCA, Fisher face, and Local Binary Patterns Histograms (LBP)—for a time execution face recognition system. The Raspberry platform is a good option for developing biometric security systems, according to the authors' conclusion. A large data set used by the several tested algorithms fills the entire memory of the system. Furthermore, the most precise method, LBPH, achieves 90% accuracy.

The implementation of a facial monitoring system using a Raspberry Pi embedded with image processing techniques is described by the authors in [1]. The foundation of this job is the ability to recognize a face by its lips, nose, and eyes. As a result, the offered program can only be used for face tracking of a single individual. Our work presents an embedded facial recognition system that can quickly identify many faces in a single frame.

Our suggested technique enables multiprocessor architecture-based face detection and recognition. When compared to our ARM processor-based platform, the results are superior to those from a Personal Computer (PC) with robust capabilities. The suggested solution uses less memory space on the chip and achieves above 98% accuracy when compared to traditional deep learning and autonomous learning algorithms

3 THE GENERAL ARCHITECTURE

The three steps of a face recognition system are face detection, feature extraction, and face recognition, as shown in Figure 1.

Fig. 1. The components of a face recognition system [4]

A feature extraction stage comes after the detection face step. This crucial stage determines the accuracy of face recognition; it involves obtaining a feature vector from the identified face, which should be adequate for the final facial representation. It must check that this face is unique, and should be able to discriminate efficiently between two different individuals. Authentication involves comparing a face with another in order to approve the requested access.

According to [4], it is interesting to classify face recognition approaches into two major classes regarding how the given image is treated. In a first class, named global approach, the entire face is studied and will be projected onto a small-dimension subspace.

An interest-point-based sub class, in which we are interested in this paper, is named local recognition approach. It deals with only some key features of the face. The efficiency of this class of approach depends on the choice of features. This second class, called also feature-based class, can be classified into two sub classes:

- An interest-point-based sub class: where a set of points of interest is defined before extracting features.
- A local appearance-based sub class: it is a region-based method where the face is divided into regions before extracting local features.

4 THE PROPOSED FACE RECOGNITION SCHEME

Our suggestion is to use a time-execution method for facial recognition in a safe medical setting. As shown in Figure 2, the proposed system operates in two stages: the face recognition stage and the face learning stage. The suggested method, as stated in Section 3, is predicated on the Landmark-based algorithm, a face recognition algorithm that extracts features locally. We will compare the suggested method to both machine learning and deep learning-based facial recognition techniques in section 5. Before going into details about the design of the suggested system, we review the fundamentals of each approach in this section.

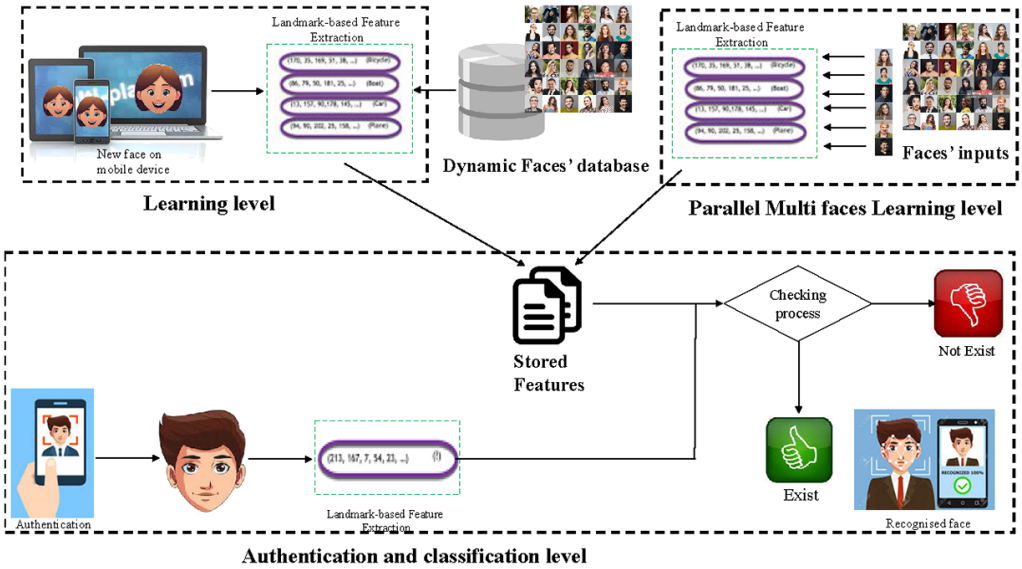

Fig. 2. The proposed system architecture

4.1 The deep learning-based face recognition approach

When considering the Deep Learning, or DL approach, it is crucial to define the artificial neural networks. The latter are algorithms that take several values as inputs to the system. These values are processed by several functions, and finally the whole system has a single value as an output [27]. Convolutional Neural Networks (CNN or ConvNet) are among the most often used varieties of deep neural networks [26]. A CNN is a well-suited architecture for processing 2D data, such as photos, because it uses 2D convolutional layers to summarize the learnt functionality with input data. The CNN makes it unnecessary to identify the features utilized in picture categorization by doing away with the necessity for human feature extraction. It operates by taking features straight out of pictures. Relevant features are acquired by the network during training on a set of images; they are not pre-trained. For computer vision applications like object classification, depth learning models achieve great accuracy because to this automatic feature extraction. CNNs use tens or hundreds of hidden layers to learn how to identify various aspects in an image. The intricacy of the taught image features grows withn each additional hidden layer.

1) The DL-based face learning step: The learning step is one important stage in calibrating the output. Using known output results, the network is initially given input values. Next, it is verified that the network consistently produces the desired outcome. It is carried out until it is set up and capable of producing the desired outcome, provided that this is not the case [27].

2) The DL-based face recognition step: The network behaves in this part as a black box. It will have as inputs new data (values unknown by the system) for which it will provide an output value [27]. Experiential learning allows the use of neural networks in several areas such as image recognition or stock market prediction as shown in Figure 3.

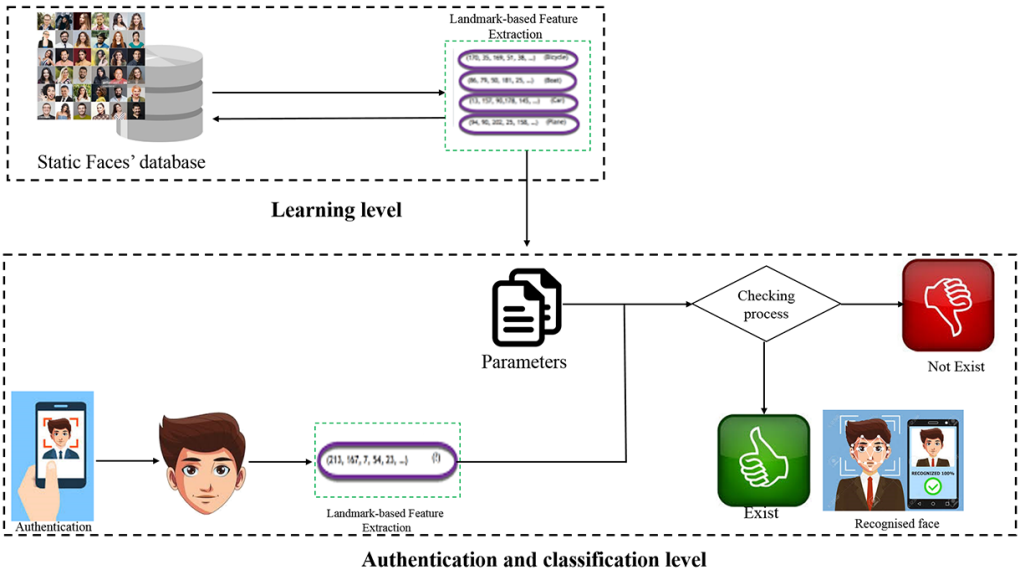

Fig. 3. The architecture of a DL-based framework

The DL-based face recognition offers more than one advantage including: the features are automatically deduced and optimized for the desired result. It has an efficient robustness to natural variations of automatically learned data, and provides a flexible learning architecture to be adaptable to new problems in the future.

However, it has some limitations when considering complexity and computation costs. In fact, it needs a very large amount of data (approximately 10,000 images per class) in order to obtain the best possible performance. The result is a static database. If a new class is added or deleted, a new learning process is necessary through a network reconstruction. Besides, due to the complexity of data models requiring massive parallel calculations, the decision is extremely costly in terms of calculation and execution time. This very time-consuming process requires multiple machines or GPUs. This significantly increases the cost to users (hardware and energy costs).

4.2 The machine learning-based face recognition step

Machine Learning, or ML, is another class of facial recognition techniques. This kind of data analysis entails imparting to computers the ability to learn from experiences, which comes easily to people. ML algorithms don't rely on a fixed equation as a model; instead, they "learn" information directly from data using computational techniques. As the amount of samples available for learning rises, the algorithms adjust and get more effective. The algorithms of machine learning (ML) detect inherent patterns in the data, producing valuable insights that facilitate improved decision-making and more precise forecasting [29].

1) The ML-based Face learning step: ML is a statistical learning tool or method for analyzing and identifying various data models. In ML, each instance of a data set is characterized by a set of attributes. In order to design a robust facial recognition system, the image must go through a series of processes before moving to the identification stage [29].

2) The ML-based Face recognition step: To classify a model in a ML process, a classifier is used. The latter uses the characteristics of an object to identify the class to which it belongs, as shown in Figure 4.

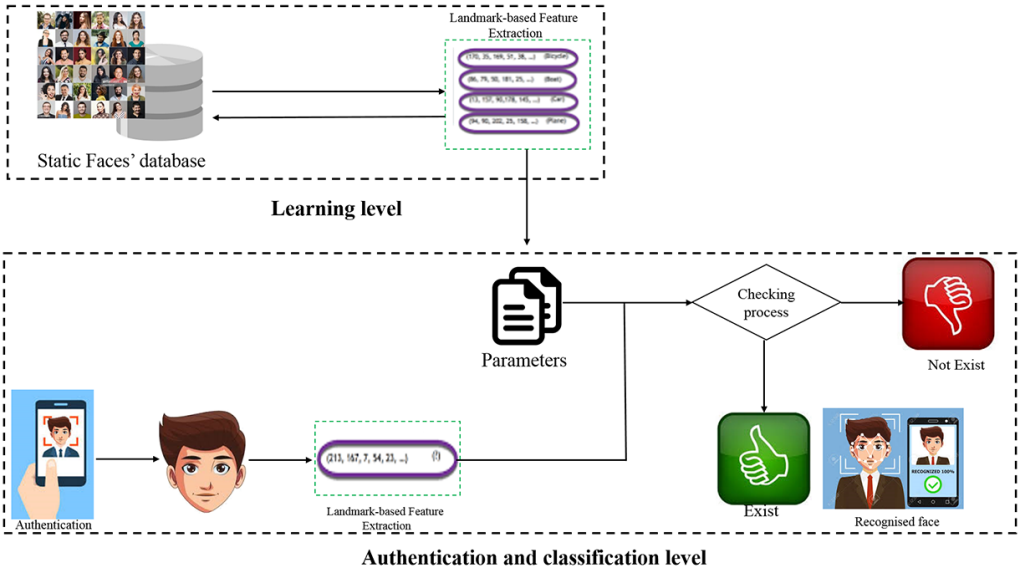

Fig. 4. The architecture of ML-based recognition framework

Many benefits come with ML-based face recognition, including the ability to quickly identify patterns and trends by sifting through vast amounts of data and identifying certain patterns and trends that a human would not notice. There is

no need for human involvement. Indeed, machines' capacity for learning makes it feasible to both improve algorithms and make predictions [29]. As autonomous algorithms gain experience, decisions get better, more accurate, more efficient. However, ML-based recognition approach has also some limitations related to its power and popularity: the necessity of a large amount of inclusive, unbiased and good quality data that needs to be classified requires a very high processing time. It requires significant time for learning and development in order to achieve sufficient accuracy and relevance. And, the probability of being prone to errors is low, where the major challenge is none other than the ability to interpret with good precision the results generated by the algorithms.

4.3 ML-based recognition approaches versus DL-based ones: a short comparison

In this subsection, we compare between ML-based recognition approaches and DL-based ones to focus on limitations of both of them. As shown in Table 1, an automatic learning workbench flow starts with the manual extraction of the relevant functionality from images. The features are then used to create a model that categorizes the objects in the image. With a flow of deep learning work, the relevant functionalities are automatically extracted from the images. In addition, deep learning performs an end-to-end learning, which means that a network receives raw data and a task to accomplish, such as classification, that it learns to do automatically. We can summarize that: the learning phase requires a very high processing time, large hardware resources (large GPU processors and memory).

The size of the database is very high and static, i.e. each addition or deletion of a new face generates a new learning process. There is no solution that ensures learning and classification in a short period of time. Each face requires 1000 samples (700 for learning and the rest for testing). Adding or removing a new face can result in a revision at the architecture level, almost requiring the involvement of a domain specialist and increasing the time-to-market. Hence, the approach we propose focuses on the problems and challenges faced by the classical learning and classifying faces methods, and offers an efficient solution for them.

Table 1. Characteristics-based comparison of two learning methods

Feature	DL Approach	ML Approach
Roughness	Yes	No
Architecture flexibility	Yes	No
Database quantity	Huge	Average
Database type	Static	Dynamic
Database size	Some GigaBytes	Few GigaBytes
Learning: Processing Time	Very High	Very Low
Learn: Memory	High Capacity	Medium Capacity
Learning: Resources	Requires GPUs	Classical CPUs
Classification: Processing Time	High Burnout	Very High
Rating: Memory	Medium Capacity	High Capacity

(Continued)

Table 1. Characteristics-based comparison of two learning methods *(Continued)*

Feature	DL Approach	ML Approach
Classification: Resources	Classical CPU	Requires GPUs
Human intervention	Yes	No
Decision	Expensive	Least expensive
Recognition rate	92%	96.8%

4.4 The proposed Landmark-based approach

1) Landmark-based face detection: Most of the current Landmark-based facial detection algorithms are predicated on the assumption that a face has already been detected, as stated in [10]. Actually, the identified face can give a rough estimate of the scale and location of the face. These algorithms do have certain drawbacks, though. Firstly, face detection is not very accurate, especially when dealing with highly variable photos. As such, there is a connection between this failure and the majority of facial landmark detection algorithms failing.

Additionally, the precision of face detectors affects the accuracy of facial landmark detection. With regard to the regression-based approaches example, the mean shape is placed in the center of the face bounding box to construct the beginning form. The scale is also calculated from the bounding box. Hence, rapid face detectors are typically chosen in order to assure facial landmark detection. The Viola-Jones face detector (VJ) has been the most wellknown face detector. Face detection is made quick and effective by using the entire image and the learning process.

The structure utilized by the part-based techniques is marginally different. They view the face as an entity made up of various elements that are constrained by space. The techniques of region-based convolutional neural networks (RCNNs) are another family of face detection algorithms. Their basic approach is to suggest a region component that finds potential face regions, followed by a detection component that further refines the suggested regions for face detection. In contrast to the current face detectors that offer real-time detection, RCNNs have high computational costs (about 5 frames per second with GPU), even though they are more accurate—especially for images with large posture, lighting, and occlusion fluctuations.

Joint landmark localization and face detection are handled by a third class of techniques. In this instance, a connect between face detection, face alignment, and even head posture estimation is accomplished using a deformable component model. Then, once the confidence based on the current face shape is less than a predetermined threshold, the bounding box is rejected and the face shape is revised roundly.

2) The annotation phase of Landmark: The methods for facial landmark detection are suggested for use with individual face photographs and, by extension, with facial image sequences or videos. These algorithms' tracking strategy is based on landmark-based detection. The process of manually labeling the ground-truth facial landmark positions on facial pictures is known as facial landmark annotation. The face key points and interpolated landmarks are the two categories of face landmarks.

Therefore, the prominent landmarks on the face, like the corners of the eyes, nose tip, mouth corners, etc., are the facial key points. Their local form patterns are distinct. On the other hand, the face contour of the interpolated Landmark ones can also be derived by joining the key points, as shown in Figure 5. This indicates that its main function is to gather separate software components, each of which has comprehensive debugging options and comprehensive documentation.

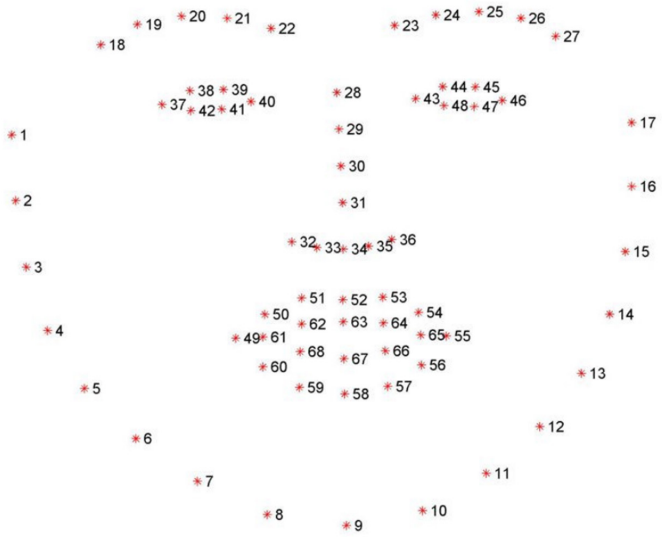

Fig. 5. The location of the 68 coordinates that map the points on a person's face [2]

4.5 Landmark-based feature extraction algorithm

Conventional techniques for face recognition rely on the visible spectrum (vs), as illustrated in Figure 3. These techniques struggle with issues like object lighting, position fluctuations, emotion changes, and facial masks. Regretfully, the object identification and verification performance is diminished by these constraints. The face has unique characteristics that make it a detectable item. We use the learning step and create the parameters to be incorporated in an embedded file, then we compare them with the ones contained in this file. The multi-face recognition block, as shown in Figure 2, makes the frame-by-frame comparison with the input allowing to make a decision and to recognize the face (adapted or not). Each face has specific characteristics that are different from one person to another. We focus on the 68 key coordinates that map the points on a person's face. We extract the features from the facial image (see Figure 2) at the entrance and compare with the embedded file when we zoom on the block called Multiface. We deduce then, that this method is more rapid and offers efficient performance compared to the other ones. Figures 6 and 7 illustrate flowcharts of the learning phase and the classification phase applied in our approach.

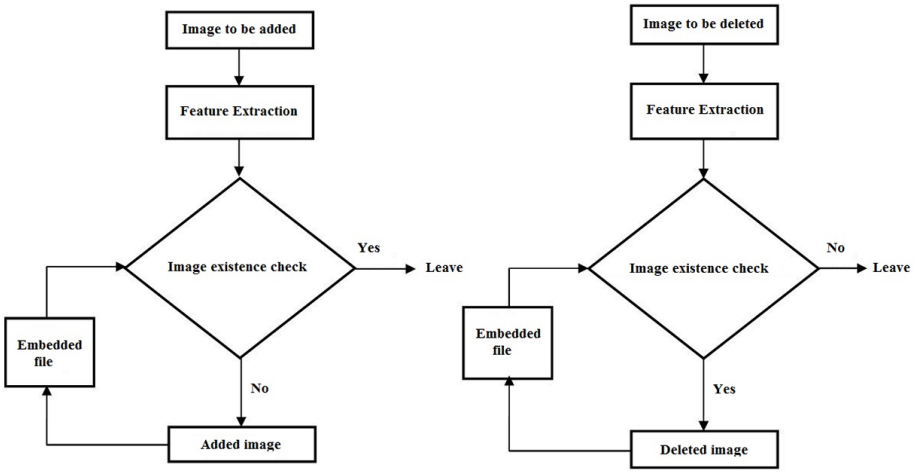

Fig. 6. Learning organization chart

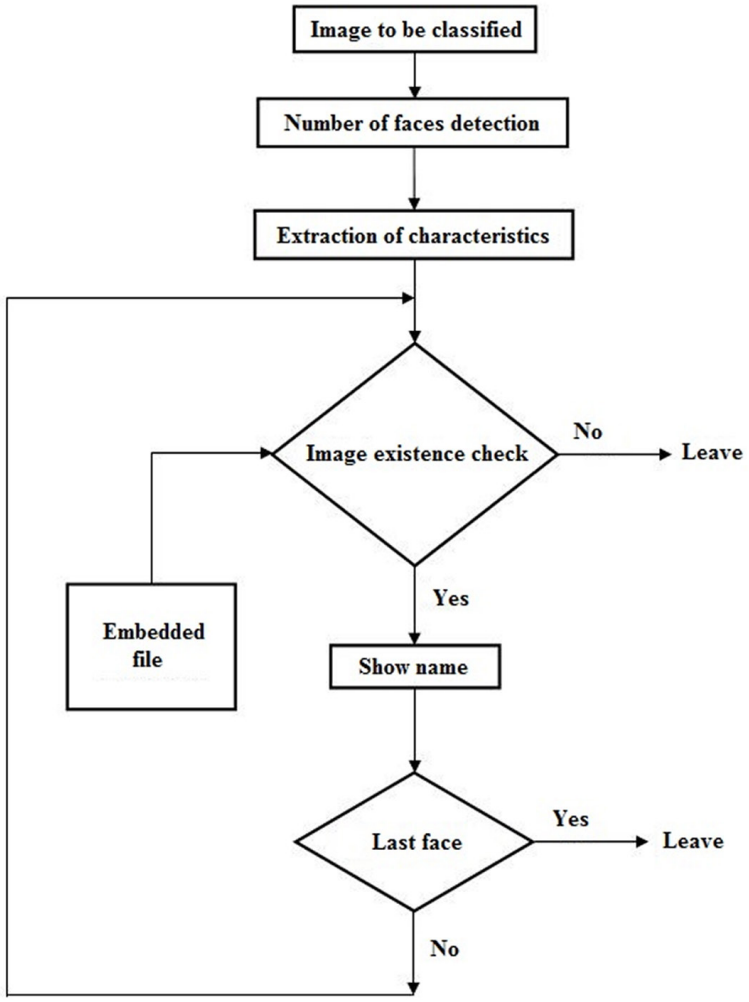

Fig. 7. Classification flowchart

5 EXPERIMENTAL RESULTS

We evaluated the built algorithm on many platforms to verify the efficacy of our suggested integrated smart facial recognition solution. We first concentrate on 5000 photos from the Face Images With Marked Landmark Points collection in our work. This dataset provides a building block for a number of applications, including biometrics and facial recognition, monitoring faces in photos and videos, analyzing facial expressions, and identifying dysmorphic facial indicators for medical diagnosis [32].

In order to determine whether or not to identify the face (adapted or not), the recognition block compares the input image with the 5000 photos. On the markers that are labeled in nature, the accuracy of the model is 99.38%. Written in the C++ programming language, DLIB is a flexible software library that runs on multiple platforms. Concepts from component-based software engineering and contract design have a big impact on its design [28]. Our system leverages the state-of-the-art facial recognition created with in-depth learning from DLIB (Digital Library). It is suggested to compare a dual-core Cortex-A9 processor operating at 650 MHz with an ARM Cortex A53 and a traditional Intel I5 processor.

We present the various implementation results with a focus on two performance criteria: “execution time” and “memory size” of the three platforms: “Personal

Computer", "Raspberry pi 3" and "PYNQ Z1". The most important one is the execution time since we aim to provide an efficient authentication process. The second is the consumed memory size. It is important to minimize the size given the constraints and resource limitations of the embedded system.

In Table 2, the execution time and image base size are exponential, requiring large resources (huge GPUs and memory sizes) and a significant amount of time for execution. For example, it takes more than an hour and a half to classify an image that contains three faces under the Raspberry Pi platform. This is very cumbersome and needs to be optimized. In order to achieve real-time results, we not only improve the software but also implement an optimized hardware architecture.

Table 2. Comparison of the classic solution on PC, Raspberry Pi and PYNQ

Classic Solution versus Proposed Solution (5000 Images BD)						
	PC		Raspberry Pi		PYNQ	
	Execution Time (Seconds)	Size (MegaBytes)	Execution Time (Seconds)	Size (MegaBytes)	Execution Time (Seconds)	Size (MegaBytes)
One Face	105	800	1970	800	950	800
Two Faces	212	800	3940	800	2000	800
Three Faces	318	800	5910	800	2950	800

Table 3 highlights the obtained results. The on-board application reaches an acceleration speed of 23.56 × average compared to classic implementation. These outcomes show how well the suggested optimized strategy works while keeping the file size to a minimum of 8 MegaBytes. Figures 8 and 9 compare the execution time of classical and optimal implementations on Raspberry Pi on the one hand and PYNQ on the other hand. The obtained results demonstrate the efficiency of our optimized algorithm.

Table 3. Classic solution versus our proposed optimized solution

Classic Solution versus Proposed Solution (5000 Images BD)						
	PC		Raspberry Pi		PYNQ	
	Execution Time (Seconds)	Size (MegaBytes)	Execution Time (Seconds)	Size (MegaBytes)	Execution Time (Seconds)	Size (MegaBytes)
One Face	105	800	4.39	8	2	8
Two Faces	212	800	9.62	8	4.5	8
Three Faces	318	800	12.84	8	6.25	8

Figure 10 shows the amount of embedded memory required in classical and optimized algorithms for all platforms. The results confirm the optimality of our algorithm requiring a minimal amount of memory, considered as an important constraint in embedded systems. This is well explained by the fact that our proposed algorithm is based on an integrated parameter file, which includes all the necessary features and parameters to ensure face detection. This eliminates the need to store all image data and allows for fast detection.

Table 4 presents a detailed comparison between the classical approaches and our approach. The proposed approach is characterized by its robustness and flexibility at the architectural level, without human intervention and at a lower cost in terms of decision-making. It works with a dynamic image base, of small size and low quality.

Learning and classification require low resources with a short processing time. The recognition rate is 99.38%.

Table 4. Comparison according to characteristics between classical and our approach

Feature	Deep Learning	Machine Learning	Our Approach
Robustness	Yes	No	Yes
Architecture	Yes	No	Yes
Database Quantity	Huge	Medium	Low
Database Type	Static	Dynamic	Dynamic
Database Size	Dozens of GigaBytes	Some GigaBytes	Some MegaBytes
Learning: Processing Time	Some Hours	Some Minutes	Some Seconds
Learning: Memory	High capacity	Medium capacity	Low capacity
Learning: Resources	Requires GPUs	Classical CPUs	Classical CPUs
Classification: Processing Time	Very Low	Very High	Short or Medium
Classification: Memory	Medium Capacity	High Capacity	Low Capacity
Classification: Resources	Classic CPU	Requires GPUs	Classic CPU
Human intervention	Yes	No	No
Decision	Expensive	Less expensive	Less expensive
Recognition rate	92%	96.8%	99.38%

We have proposed an embedded face recognition approach optimized for machine learning system considering the realtime constraint. Classical approaches are based on: loading the folder that contains the images (DataSets), loading the image to be tested, encoding the loaded image into a feature vector. We take the following actions for each image in the Datasets [32]: uploading photos and turning them into a vector feature set. The image to be examined is then contrasted with the loaded image. Next, we compare how similar the two photos are to one another.

With regard to time execution, this algorithm has an exponential execution time which depends on the number of dataset images. Moreover, another weakness tightly tied to the tested images is that the image to be tested must contain one and only one face. The dataset size is also exponential as it depends on the number of dataset images.

For each facial recognition run, we have to upload and encode all the images from the datasets. Now, compared to the previous approaches, the proposed approach provides some improvements: In fact, the learning and classification operation is separated by saving the characteristic vector of each image in a well determined embedded file, which requires a file size of 8 MB instead of 800 MB compared to the classical method. The loading of the characteristic vectors saved in the file requires 0.1 seconds (instead of 100 seconds).

It is possible to test on an image that contains more than a face. In case of adding a new image to the dataset, the operation requires 0.5 seconds (constant) whereas in the classical approach, this processing is exponential (number of images * 0.5 + 0.5). Besides, when deleting an image from the dataset, this operation requires 0.5 seconds (constant) compared to number of images * 0.5 – 0.5 (exponential). The image loading phase is eliminated each time.

The embedded application has been implemented on a multiprocessor platform based on an ARM processor. Experimental results showed that the proposed implementation was more than 23 times faster than the same approach applied on a PC.

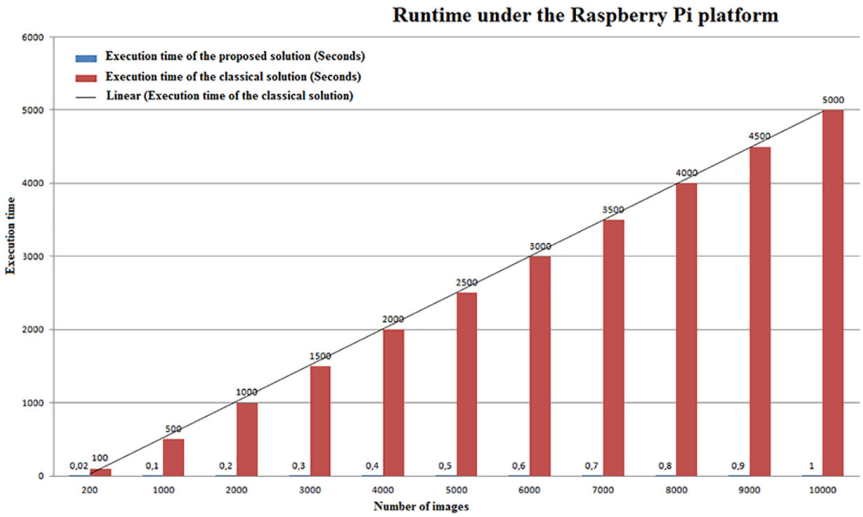

Fig. 8. Runtime results with Raspberry Pi

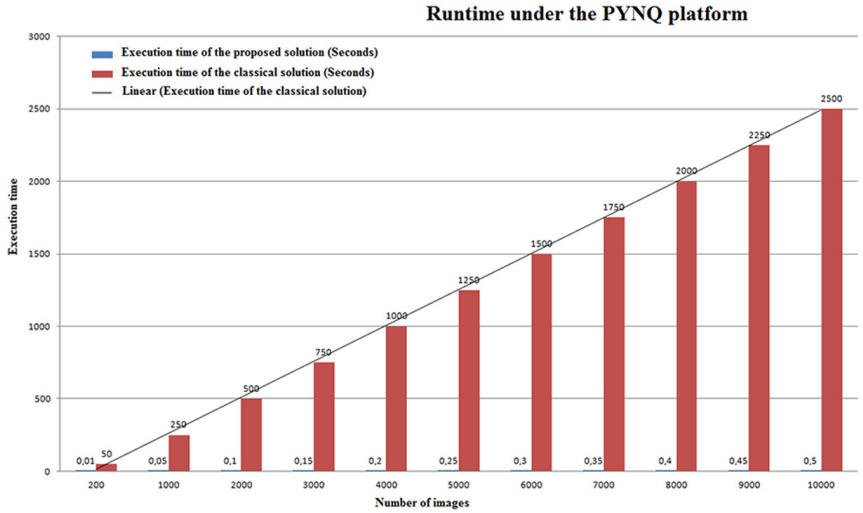

Fig. 9. Runtime results with PYNQ Z1

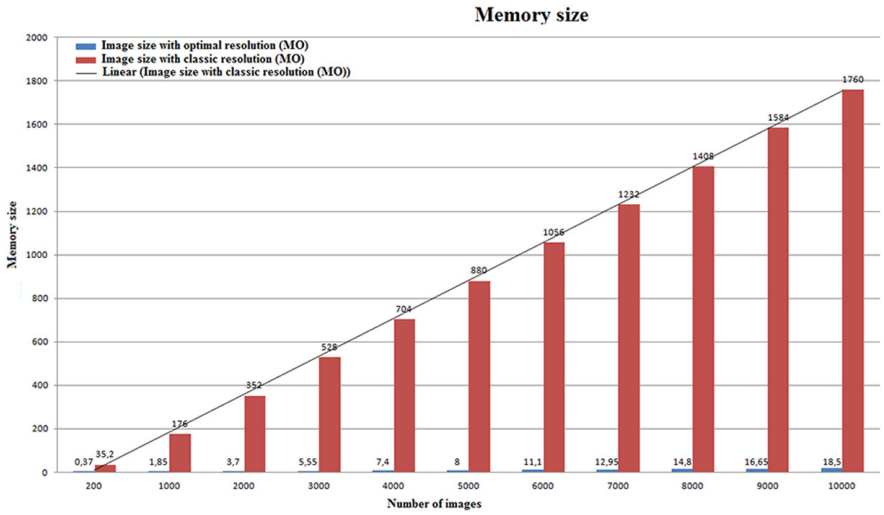

Fig. 10. Memory size of face recognition algorithms under Raspberry Pi

6 CONCLUSION

Face detection and recognition is an area of research that has been constantly evolving in terms of face detection rate, algorithm complexity and implementation in hostile environments with harsh constraints such as real-time, limited resources, etc. In this paper, a learning-based authentication approach using the Landmark algorithm was proposed for embedded devices. The implementation of this algorithm on a low-resource multiprocessor embedded platform has allowed to obtain results that meet the time execution constraints with a recognition rate of 99.38%. Such authentication process will be integrated in an e-health platform for secure sharing of medical documents. Another future work is to improve the embedded architecture and further optimize the software face recognition algorithm by using heterogeneous multiprocessor architectures with coprocessors. For critical applications, in particular executed on systems under hard real-time constraints, approximate computing [25] would allow to speed up the execution and achieve a more efficient result.

7 REFERENCES

[1] A. A. Shah, Z. A. Zaidi, B. S. Chowdhry, and J. Daudpoto, "Real-time face detection/monitor using Raspberry Pi and Matlab," in *IEEE 10th International Conference on Application of Information and Communication Technologies (AICT)*, 2016. https://doi.org/10.1109/ICAICT.2016.7991743

[2] B. Nataliya, B. Oleg, and S. Nataliya, "Performance evaluation and comparison of software for face recognition, based on Dlib and OpenCV library," in *2018 IEEE Second International Conference on Data Stream Mining Processing (DSMP)*, 2018, pp. 478–482.

[3] X. Cheng and M. Li, "The authentication of the grid monitoring system for wireless sensor networks," 2013.

[4] C. Mejda, E. Akram, B. Wajdi, and C. Ben Amar, "A survey of 2D face recognition techniques," *Computers*, vol. 5, no. 4, p. 21, 2016. https://doi.org/10.3390/computers5040021

[5] D. Jose, R. Víctor, and C. Amir, "Biometric access control for e-health records in prehospital care," in *Proceedings of the Joint EDBT/ICDT 2013 Workshops*, 2013, pp. 169–173. https://doi.org/10.1145/2457317.2457345

[6] G. Fang, H. Zhangqin, W. Shulong, and J. Xinrong, "Optimized parallel implementation of face detection based on embedded heterogeneous many-core architecture," *Int J Pattern Recognit Artif Intell.*, vol. 31, no. 7, p. 1756011, 2017. https://doi.org/10.1142/S0218001417560110

[7] J. L. Fernandez-Alemain, I. C. Seaor, P. O. Lozoya, and A. Toval, "Security and privacy in electronic health records: A systematic literature review," *Journal of Biomedical Informatics*, vol. 46, no. 3, pp. 541–562, 2013. https://doi.org/10.1016/j.jbi.2012.12.003

[8] G. Ishita, P. Varsha, and K. Shreya, "Face detection and recognition using Raspberry Pi," in *IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)*, 2016.

[9] J. H. Shah, M. Sharif, M. Raza, and A. Azeem, "Face recognition across pose variation and 3S problem," *TÜBİTAK Academic Journals*, 2012.

[10] Y. Wu and Q. Ji, "Facial landmark detection: A literature survey," *International Journal on Computer Vision*, vol. 127, no. 2, pp. 115–142, 2018. https://doi.org/10.1007/s11263-018-1097-z

[11] A. Lanitis, C. J. Taylor, and T. F. Cootes, "Toward automatic simulation of aging effects on face images," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, pp. 442–455. https://doi.org/10.1109/34.993553

[12] A. Lanitis, C. J. Taylor, and T. F. Cootes, "Toward automatic simulation of aging effects on face images," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, pp. 442–455. https://doi.org/10.1109/34.993553

[13] L. Yanrong, L. Lixiang, P. Haipeng, and Y. Yixian, "An enhanced biometricbased authentication scheme for telecare medicine information systems using Elliptic curve cryptosystem," *Journal of Medical Systems*, vol. 39, no. 3, p. 32, 2015. https://doi.org/10.1007/s10916-015-0221-7

[14] M. Sharif, S. Mohsin, M. J. Jamal, and M. Raza, "Illumination normalization preprocessing for face recognition," in *IEEE International Conference on Environmental Science and Information Application Technology (ESIAT)*, 2010, pp. 44–47.

[15] D. P. Mirembe, "Design of a secure framework for the implementation of telemedicine, eHealth, and wellness services," Radboud University, Nijmegen, the Netherlands, p. 8, 2006.

[16] I. Nanayakkara and G. U. Ganegoda, "Biometric traits in enhancing security of healthcare," in *2018 3rd International Conference for Convergence in Technology (I2CT)*, 2018, pp. 1–6. https://doi.org/10.1109/I2CT.2018.8529442

[17] P. Irgens, C. Bader, T. Le, D. Saxena, and C. Ababei, "An efficient and costeffective FPGA based implementation of the Viola-Jones face detection algorithm," *HardwareX*, vol. 1, pp. 68–75, 2017. https://doi.org/10.1016/j.ohx.2017.03.002

[18] F. Rezaeibagha, K. T. Win, and W. Susilo, "A systematic literature review on security and privacy of electronic health record systems: Technical perspectives," *Health Information Management Journal*, vol. 44, no. 3, pp. 23–38, 2015. https://doi.org/10.1177/183335831504400304

[19] Umm-e-Laila, M. A. Khan, M. K. Shaikh, S. A. bin Mazhar, and K. Mehboob, "Comparative analysis for a real-time face recognition system using Raspberry Pi," in *IEEE 4th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA)*, 2017. https://doi.org/10.1109/ICSIMA.2017.8311984

[20] S. Wang and J. Liu, "Biometrics on mobile phone," in *Recent Application in Biometrics*, Jucheng Yang and Norman Poh, Eds., IntechOpen, 2011, chapter 1, p. 10. https://doi.org/10.5772/17151

[21] W. K. Than, S. Willy, and M. Yi, "Personal health record systems and their security protection," *Journal of Medical Systems*, vol. 30, no. 4, pp. 309–315, 2006. https://doi.org/10.1007/s10916-006-9019-y

[22] X. Zhao, X. Liang, C. Zhao, M. Tang, and J. Wang, "Real-time multi-scale face detector on embedded devices," *Sensors*, vol. 19, p. 2158, 2019. https://doi.org/10.3390/s19092158

[23] K. Zhang, K. Yang, X. Liang, Z. Su, X. Shen, and H. H. Luo, "Security and privacy for mobile healthcare networks: From a quality of protection perspective, *IEEE Wireless Communications*, vol. 22, no. 4, pp. 104–112, 2015. https://doi.org/10.1109/MWC.2015.7224734

[24] Z. Zhou, A. Wagner, H. Mobahi, J. Wright, and Y. Ma, "Face recognition with contiguous occlusion using Markov random fields," in *IEEE 12th International Conference on Computer Vision*, 2009.

[25] I. Zriqat and A. M. Altamimi, "Security and privacy issues in ehealthcare systems: Towards trusted services," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 9, 2016. https://doi.org/10.14569/IJACSA.2016.070933

[26] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, "Conceptual understanding of convolutional neural network – A deep learning approach," *Procedia Computer Science*, vol. 132, pp. 679–688, 2018. https://doi.org/10.1016/j.procs.2018.05.069

[27] D. Dandan, K. Lingyi, C. Guangyao, L. Zoe, and F. Yong, "A switchable deep learning approach for in-loop filtering in video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 7, pp. 1871–1887, 2020.

[28] B. Nataliya, B. Oleg, and S. Nataliya, "Performance evaluation and comparison of software for face recognition, based on Dlib and OpenCV Library," in *2018 IEEE Second International Conference on Data Stream Mining Processing (DSMP)*, 2018, pp. 478–482.

[29] T. Isabelle, "Apprentissage automatique pour le TAL: Préface," *Revue TAL : traitement automatique des langues*, vol. 50, no. 3, pp. 7–21, 2009.

[30] D. Abhilash, C. Chandrashekar, and S. Shalini, "Economical, energy efficient and portable home security system based on Raspberry Pi 3 using the concepts of OpenCV and MIME," in *2017 International Conference on Circuits, Controls, and Communications (CCUBE)*, 2017, pp. 60–64. https://doi.org/10.1109/CCUBE.2017.8394155

[31] B. Vamsi, A. Sagheer, G. Ilya, K. Vinod, R. Vidya, and W. Ralph, "UltraScale+ MPSoC and FPGA families," in *2015 IEEE Hot Chips 27 Symposium (HCS)*, 2015, pp. 1–37.

[32] P. Karen, W. Qianwen, A. Sos, R. Srijith, K. Shreyas, R. Rahul, R. S. Paramathma, K. Aleksandra, T. A. Holly, S. Arash, and Y. Xin, "A comprehensive database for benchmarking imaging systems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 3, pp. 509–520, 2020. https://doi.org/10.1109/TPAMI.2018.2884458

8 AUTHORS

Hedi Choura was born in Sfax (Tunisia) in 1995. He is a PhD student in Computer Science from the National Engineering School of Sfax (ENIS), Tunisia, since December 2019. He received the engineering and MS degree from the National Engineering School of Sfax (ENIS), Tunisia, in 2019. He is a research member of Computer Embedded Systems Laboratory (CES-Lab), (ENIS), Sfax (Tunisia). His research interests include smart embedded systems design, FPGA prototyping, IoT, Security systems, Blockchain and e-health (E-mail: hedi.choura@enis.tn).

Faten Chaabane graduated from the National School of Engineers of Sfax in 2006, obtained her Master's degree in Communication Systems from the National School of Engineers of Tunis in 2012 and her PhD in Computer Systems Engineering in March 2017. Currently, she is an Associate Professor in the Department of Computer Science and Multimedia at ISLAIB Beja, University of Jendouba. Her research work mainly focuses on cybersecurity, watermarking, multimedia document tracing, and since 2020, she has been interested in securing data with Blockchain architectures.

Mouna Baklouti was born in Sfax (Tunisia) in 1983. She is currently an Associate Professor in Computer Science at the University of Sfax. She received the engineering and MS degrees from the Tunisian Polytechnic School, Tunisia in 2006 and 2007, respectively. She received a PhD in Computer Science from the National Engineering School of Sfax (ENIS), Tunisia and University of Lille 1, France in December 2010. She received a HdR in electrical engineering from ENIS School in June 2016. Since 2012, she has been a co-founder and coordinator of the research master in embedded systems, at the ENIS School in cooperation with the University of Chemnitz, Germany. She is a research member of Computer Embedded Systems Laboratory (CES-Lab), (ENIS), Sfax (Tunisia). Her research interests include smart embedded systems design, IoT applications, Blockchain and e-health.

Tarek Frikha was born in Sfax (Tunisia) in 1982. He is an assistant professor in National Engineering School of Sfax. He received the engineering degree in electronic engineering from National School of Engineers of Sfax, in 2006 and the Master Diploma from polytech Sophia Antipolis in France. He received PhD in science and technology of information and communication from the University of South Brittany, France, and the National Engineering School of Sfax, Tunisia. His research interests include multiprocessor architecture optimization for multimedia domains, hardware/software codesign, approximate computing, Blockchain for multimedia applications, medical and paramedical data and agricultural applications.