Dynamic Facial Presentation Attack Detection for Automated Border Control Systems

David Ortega, Alberto Fernández-Isabel, Isaac Martín de Diego, Cristina Conde, Enrique Cabello
{david.ortega,alberto.fernandez.isabel, isaac.martin, cristina.conde, enrique.cabello}@urjc.es

Rey Juan Carlos University
Face Recognition and Artificial Vision Group
Data Science Laboratory
C/ Tulipán, s/n, 28933, Móstoles, Spain

Abstract

Every day millions of passengers travel, being border crossings one of their most common activities. At these points it is extremely important that the security is completely guaranteed. Nevertheless, the maintenance of the proper security levels is a very demanding issue. This has promoted the development of systems able to provide support to the border authorities automatising some of their tasks. Thus, Automated Border Control (ABC) systems have become a key tool. These systems increase the flow of travellers as they can achieve fast evaluations of individuals through their machine-readable travel documents. However, this has motivated the appearance of attacks that try to avoid the identity detection of individuals by these systems. Presentation Attack Detection (PAD) algorithms have arisen to mitigate such a problem. This paper presents the *On-the-Fly Presentation Attack Detection (FlyPAD)* framework that implements a set of dynamic PAD techniques. It allows detecting multiple attack types while the traveller is approaching to the ABC system, instead of being static in front of cameras. Several experiments have been carried out, both in laboratory and in real environments, obtaining promising results.

Keywords: Border access control, Dynamic detection, Presentation attack detection, Face recognition, Face anti-spoofing

1. Introduction

Every day millions of border crossings happen all over the world. It was estimated that around 3.5 million people daily crossed internal borders between Schengen countries in 2017 [1]. In the case of the United States, the number of border crossings in the same year was roughly 400 million [2], while in the United Kingdom the quantity of 314 million [3] was reached in 2016.

For all these border crossings, security agents have to determine as fast as possible the following: who may or may not enter the country according to immigration policies or national security aspects, whether a traveller is in a watch-list of suspects, update the corresponding databases, or even in some cases, stamp the travellers passport. As a whole, all these operations become a time-consuming task. Thus, it leads to automatising them to make easier the work of border agents, speeding up the transit of people and increasing the traveller comfort.

In particular, in the last years the so-called Automated Border Control (ABC) systems [4] have emerged to assist the border authorities by automating (totally or at least partially) the process. Thus, ABC systems have currently spread to all kinds of borders.

These systems allow increasing the flow of travellers through a border, while keeping control on security issues. They rely on machine-readable travel documents, such as passports, visas, Id cards, or even frequent traveller cards. These documents contain a chip with information such as the travellers personal data, and some biometric traits (e.g. face, fingerprints and iris).

One task performed by the ABC systems is the biometric identification of the traveller. It is usually performed comparing the captured face of the subject in static frontal position with the face image stored in the passport chip. For this configuration (subject stated in front of the camera), systems have included different Presentation Attack Detection (PAD) algorithms (see, for instance, [5]). This has incremented their capability of detecting diverse types of attacks like masks, printed photos or screen videos. In any case, most responsibility of attack detection relies on border guard that supervises the system.

New trends in this technology is the deployment of *On-the-Fly* ABC systems [6]. This approach is much more comfortable for the traveller because all the process can be done while passenger is walking. This case means that the face processing algorithms have to run as soon as the face is detected, without requiring a static pose nor collaboration of the user. In the experimental situations in which *On-the-Fly* ABC systems have been deployed, the focus has been placed on facial verification rates and no report has been found showing PAD results.

The present paper introduces the *On-the-Fly Presentation Attack Detection (FlyPAD)* framework that implements dynamic PAD algorithms for face recognition in five different types of attacks: printed photos, paper masks, paper masks without eyes, screen videos, and 3D masks.

Multiple experiments in a controlled environment (i.e. in a laboratory) and in a real border scenario have been achieved to illustrate the viability of the system. These experiments are focused on evaluating the system performance and the PAD capabilities of the system in static and dynamic situations.

The remainder of this paper is organised as follows. Section 2 introduces the foundations of the proposal. Section 3 presents the developed framework detailing the architecture, while Section 4 is focused on the image database generated for the attack detection. Section 5 addresses the different experiments focusing on the obtained results. Finally, Section 6 concludes and provides the future guidelines.

2. Background

This section describes the foundations of the *FlyPAD* framework. Firstly, it addresses the ABC systems detailing their configurations, possible designs and implantation in real environments. Then, presentation attacks are introduced, establishing a basic classification and describing the most typical instruments used to achieve them. Finally, the PAD systems are presented, illustrating how they work, evaluating the considered technologies and their strengths and weaknesses.

2.1. ABC systems

An ABC system is an automated system with multiple sensors which performs three main specific tasks according to the European Border and Coast Guard Agency (Frontex) [7]. First, it accepts and reads the passengers travel document (e.g. passport or visa) or token with data stored in a chip, and authenticates its validity. Second, it checks that the traveller is the owner of the document, which means it has to acquire some real-time biometric data of the traveller (mainly face and fingerprints) and compares them with those stored in the chip of the document. Third, it submits a query to the border control databases to check whether the traveller has right to cross the border according to administrative or legal rules.

In the case of European border crossings where ABC systems are deployed, it is required checking subject identification against two lists: RTP (Register Traveller Programme) and EES (Entry/Exit System) [8]. The identity of travellers and their suitability to cross the border is verified in the RTP checking according to the information stored in their documentation. Once the travellers are identified, their data are registered. At border crossing time, biometric match between registered information and data captured is achieved by EES.

Depending on the devices in which the RTP and EES processes are performed, there are different ABC systems topologies: *One Step Process* and *Two Step Process* [9]. In the *One Step Process* topology, RTP and EES are merging into a single process in which the identification of travellers is carried out at the same time as travellers cross the border. Devices for this type of topology are usually mantrap e-gates [4], that do not allow crossing until identification has been correctly carried out. In the *Two Step Process* topology, RTP and ESS processes are well differentiated. Travellers are registered and then, their biometric information is matched before allowing crossing. *Integrated* or *Segregated Two-Step* ABC can be considered attending whether RTP and EES are achieved through one or two devices.

Regarding the implantation of the ABC systems and their usability, it is important to mention that almost all airports receiving travellers from non-Schengen countries use them (a complete map of airports with ABC is presented in [10]). However, ABC-equipped seaports are not so frequent.

In the case of the configurations of ABC systems, they can have several physical configurations [11]. The most typical use of electronic gates (e-gates) [12]. These devices regulate travellers flow through the border with the use of biometric sensors (e.g. cameras for face recognition [9] and fingerprint readers [13]), travel document readers (e.g. scanners [14] and radio frequency contactless chip readers [15]), as well as physical barriers that let or not the traveller to cross the e-gate [16].

Delving into the design of ABC systems, their capability to recover from problematic situations (i.e. resilience) and to resist against external assaults (i.e. robustness) are their principal security requirements. Most typical attacks are focused on the biometric system. These attacks are called presentation attacks, which consist of an attacker presenting to sensors forged biometric features of another subject for obtaining permission to cross the border. For this reason, ABC systems include some kind of PAD and anti-spoofing module in the process of biometric recognition.

In the case of *FlyPAD*, it has been preliminary tested in laboratory simulating a border crossing. Then, it has been included into an ABC system with e-gates in a real border scenario. Both perspectives have shown the viability of the prototype. Nevertheless, this framework has as a main purpose the dynamic PAD. This leads the system to fit better with the *Segregated Two-Step* topology as the system performs the facial verification having previously registered the traveller information. It is also interesting to remark that the detection of manipulated travel documents is out of the scope of *FlyPAD*.

2.2. Presentation attacks

A presentation attack could be defined as the impersonation of an individual (i.e. the victim) who possesses the desired authorisation. There are several techniques to carry out these type of attacks, but all of them can be organised into biological-based attacks and document-based attacks [17]. The former are mainly focused on three main elements: face [18], fingerprint [19] and eyeprint (i.e. iris recognition) [20]. The latter usually considers the documents used by travellers to identify them (e.g. passport and other travel documents) [21]. It is typical that these attacks can tackle more than one element (e.g. the face and the fingerprint, or the face in the picture of the passport [22]).

Delving into face presentation attacks, several artefacts or Presentation Attack Instruments (PAIs) have been identified in the literature [23]. For instance, the so-called photo attacks consist of presenting a face picture to the system instead of the face itself. This picture can be a standard photograph printed in paper or it can be shown with the help of electronic devices (laptop, tablet, or mobile phone). These devices can enhance the attack showing a video in front of the sensor [24]. This sensor usually only takes into account the normal movement of the head or specific features, such as lips (when reading out a sentence) or eyes (blinking, reaction to light), which makes more difficult to detect the attack. This issue can be addressed detecting the presence of some strange elements such as hands in the acquired image or the edges of the picture. Another well-known type of attack makes use of face masks [25]. The easiest case is printing a face picture into a mask which is worn by the attacker. The eyes area is usually cut to let the eyes of attackers be visible to prevent a possible eye blinking detection module [26]. On the other hand, the advents of cheap 3D printers have paved the way to using 3D realistic masks which mimic the face of other individuals [27]. There are several commercial solutions for creating these masks with a handful of normal photographs (frontal and two profiles). Also, in this point it is important to consider the use of make-up, disguises, wigs, fake beards or moustaches, and even plastic surgery to carry out this kind of presentation attacks [28].

In the case of the *FlyPAD* framework, it is focused on face presentation attacks. It considers the following types of attacks: printed photos, paper masks, paper masks without eyes, screen videos and 3D masks. Thus, the system covers most of the spectrum of the related literature, making it very robust to face presentation attacks.

2.3. Presentation attacks detection

Biometric systems, including those based in face recognition, usually comprise several modules devoted to specific functions, such as data capture (sensor), feature extraction, data storage, score comparison and decision [29]. Depending on which system module is hacked, several vulnerabilities can be identified. In particular, presentation attacks take place at the front end of the system (i.e. at the sensor level). Thus, attackers present to the system spoofed biometric traits (this is, fake or forged). This kind of attacks are simple to commit as they are external to the system, in contrast to others which need the thorough knowledge of how the system works (e.g. to hack the feature extractor, the database, the classification or the decision modules). This issue makes presentation attacks the most likely form of attack for a face recognition system [30].

Concisely, PAD algorithms can be classified into hardware-based or software-based [5]. The hardware-based (or sensor-level) methods rely on intrinsic properties of the body. Notice that, in some cases, a specific or non-conventional sensor is needed to acquire these features. Instances of these properties are: facial textures, electrical resistance, temperature, sweat, colour, skin reflectance for wavelengths other than visible and 3D shape. Some of these properties are involuntary as they are controlled by the nervous system, in particular pulse, ocular saccades, and breathing. In the case of the anti-spoofing systems, they produce a stimulus and try to detect body reactions (challenge-response methods) [31]. A common instance of these methods consists of requesting the user to follow a light with the head, or reading out a sentence. Involuntary reactions can be searched, such as eye blinking or pupil constriction due to dazzling light [32]. Although these tasks can be performed by software, they can be also carried out by dedicated hardware. Finally, the use of multimodal strategies (same trait, multiple sensors) or multibiometrics (multiple traits, same sensor) can increase the robustness of the system against spoofing attacks.

In contrast, software-based (or feature-based) methods are applied by a module located just after the sensor, so they operate over the biometric sample acquired. This provides high accuracy and relatively low cost [5]. Most of these methods only rely on static features, this is, on a single image. These latter can be organised into texture-based static and frequency-based static methods [5]. The first ones can detect the facial features, or even the presence of artefacts due to low printing quality. The second ones make use of the spectral information contained in a face image. Both methods can be combined through hybrid approaches. Nevertheless, some software-based methods are dynamic in the sense that they also take into account timing information. For instance, when the sensor acquires videos instead of snapshots. Thus, some texture-based methods depend on motion features of the incoming data, such as head movement tracking, background motion or optical flow [33].

In the case of *FlyPAD*, the PAD task relies on hardware-based algorithms. These algorithms are able to detect distant individuals, indicating possible presentation attacks while they are in movement (i.e. *On-the-Fly*). This issue differs from the related literature on the domain, being one of the main novelties provided by the system.

3. FlyPAD system architecture

The *FlyPAD* framework has as a main purpose the dynamic PAD. This is known as *On-the-Fly or On-the-move* detection, and it is able to provide agility in border crossings (see Fig. 1). Therefore, travellers do not need to pose in front of the sensors to be analysed. Notice that the system can also work normally, being able to carry out the PAD task in a static configuration. *FlyPAD* considers five different types of attacks (see Fig.2): printed photos,

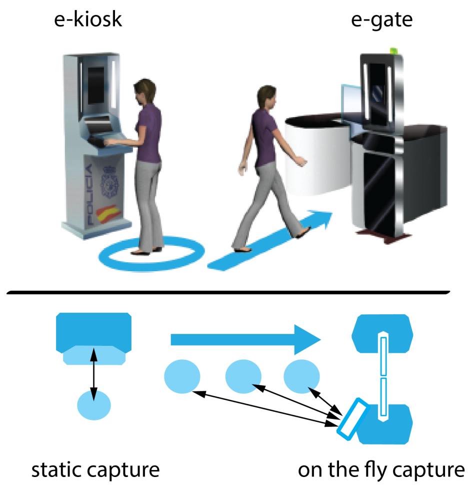


Figure 1: Top schematic view of the e-gate, with the enrolment kiosk (left) and the e-gate (right). Bottom: Face acquisition for static and dynamic capture

paper masks, paper masks without eyes, screen videos and 3D masks.

The current proposal has addressed several challenges that are present in real borders. Parameters such as lighting, pose or camera distance are extremely controlled at current PAD systems for static devices, but it is not easy to control them in a dynamic scenario. This leads to developing techniques able to solve these issues. These techniques have been included in the different components of the architecture of the system.

The architecture of the *FlyPAD* framework comprises four modules: *tracking*, *detection*, *verification*, and *PAD*. They are complemented by the *capture device*, the *models repository* and the *tracking token* (see Fig. 3).

The *tracking* module is the underlying element. It monitors the movement of travellers in their approach to the e-gate. This module locates travellers using the *detection* module, validates their identity through the *verification* module and detects facial biometric attacks with the *PAD* module. Thus, the system must decide with all the captured tracking information whether travellers can cross the e-gate. These modules are detailed in next sections.

Figure 2: Leftmost image: example picture. Left side, upper row: normal image, printed image, paper mask with eyes. Left side, lower row: paper mask with eyes holes, screen video and 3D mask.

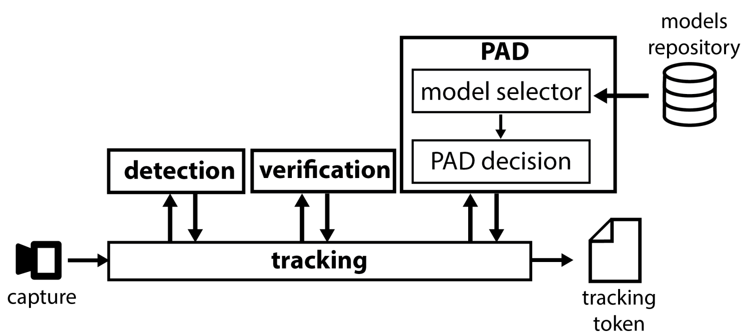


Figure 3: Overview of the architecture of the system.

3.1. Tracking module

The *tracking* module works in consonance with the rest of modules exchanging information with them. Thus, it receives 25 fps (frames per second) from the *capture* device. Each frame consists of a RGB image with $1,920 \times 1,080$ pixel resolution. Only one of the five captured frames is sent to be processed by the *detection* module. If one face is detected, the *detection* module returns the region where the face was located and estimates the distance to the *capture* device. Each face is evaluated by the *verification* module to authenticate the traveller and by the *PAD* module to detect possible attacks.

Considering that border crossing is individually performed, only one subject at a time should be presented in the corridor approaching the e-gate. When a face is located for the first time, the *tracking* module generates a *tracking token* associated with the subject. This *tracking token* is active until the user crosses the border and stores the identity information through all the process. Each time a new face is detected, it is compared with the active *tracking token* to decide if it corresponds to the current identity. In case of a positive matching the *tracking token* information is updated. In case of five consecutive negative matching or acquisition failure, token is discarded and border crossing is not allowed, redirecting the subject to a static e-gate.

In addition to the identity information, the *tracking token* stores other tracking information such as number of acquisition failures, the *verification* module results or the *PAD* module results. When travellers are in front of the ABC system, the *tracking token* information is used to decide if they can cross the border. If the *tracking token* contains more than a certain amount of consecutive authenticated tracking frames (i.e. *bona fide* presentations [5]), then the system allows the traveller to cross. Otherwise, if anomalous tracking frames are detected by the *PAD* module, the system considers the traveller as an attacker. Then, a manual verification by the security agents is required.

3.2. Detection module

This module has as a main purpose to search faces on every received frames. It uses the well-known Viola-Jones algorithm [34] to perform this task. When a face is located, this module considers the camera resolution and the size of the detected region to estimate the distance of the individual to the *capture* device.

Three different distance ranges are considered by the module to optimise the detection performance (see Table 1). The first range considers more than 2 meters to the *capture* device. The second range considers more than 1 meter but less than 2 meters to the *capture* device. The third range contemplates distances less than 1 meter to the *capture* device.

Range	Image Size	Distance
Out of Range	50×50 px	–
Range-1	$\geq$ 50×50 px $-$ $<$ 150×150 px	$\geq$ 2m
Range-2	$\geq$ 150×150 px $-$ $<$ 250×250 px	$<$ 2m $\geq$ 1m
Range-3	$\geq$ 250×250 px	$<$ 1m

Table 1: Region faces size for each selected range.

Regions with less than 50×50 pixel resolution are discarded, as they are too far away to be verified. Regions between 50×50 and 150×150 pixel resolution are considered from the first distance range. Regions between 150×150 and 250×250 pixel resolution are classified into the second distance range, and regions larger than 250×250 pixel resolution from the third distance range. This information is stored in the *tracking token*.

3.3. Verification module

This module has been designed for the *Segregated Two-Step* topology. This means that the RTP process has been previously performed on another device and the information of the travellers is already recorded in the system. Moreover, the EES process is carried out in the e-gate, where the registered biometric information of a traveller and the information captured in situ are compared [7].

To confirm that the captured identity is the same as the registered one, facial verification between them is required. Cognitec facial recognition is used by this module for the face verification task [35]. Congnitec has been specialised on travel documents and its algorithm is in the top-ten performance ranking in NIST FRVT 1:1 test with visa images [36].

In addition to face verification, several security checks in protected databases and systems as VIS (Visa Information System) [37] or SIS (Schengen Information System) [38] are included in this module.

3.4. PAD module

The *PAD* module decides if the detected face is a presentation attack or a *bona fide*, returning this information to the *tracking* module.

The detection task is carried out in two stages (see Fig. 3). In the first stage, the appropriate classifiers stored into the *models repository* module are used to evaluate the capture. This capture is scaled to 100×100 pixel resolution, converted to grayscale and the histograms of the Local Binary Patterns (LBP) [39] is calculated to get a feature vector. Five classifiers are selected (one per attack type) according to the distance of the detection from the device and the predefined three distances ranges. This task is achieved by the *model selector* component. In the second stage, the selected classifiers return the attack probability for the trained PAI. Finally, calculating the average of the response for each classifier it is possible to obtain the probability for a face to be considered a *bona fide* or an attack. This task requires to select one threshold for mean probability. This threshold depends on a desired confidence value (i.e. how many attacks the system is able to accept, or how many times the system produces unnecessary alarms). In the case of the ABC systems, guidelines from Frontex have been considered to fix the parameters values [7].

3.5. Models repository

The *models repository* module stores the different Machine Learning models used by the *FlyPAD* framework to achieve the attack detection tasks. Each classifier must be a lineal bi-class (attack or *bona fide*) and implements the same methods [40].

The module is organised into three different sets of classifiers according to specific distance ranges (Range-1, Range-2 or Range-3). In each one, five models (one per considered PAIs: photo, paper mask, paper mask without eyes, screen video, or 3D mask) are included. This issue is motivated by the fact that multiple specialised classifiers present better results than just one for tracking on route individuals by modifying their distance to a specific origin [41]). Moreover, this decision provides flexibility to the framework, making possible to add new PAIs without retraining the rest of classifiers.

The *models repository* module does not need to store any data of the individuals that have been used for the training of the classifiers. Each model only requires the hyper-plane information that allows it to distinguish between attack or *bona fide* cases.

4. Test scenarios setup

The *FlyPAD* framework performance has been tested in two scenarios. The first called *FRAV-ABC-OnTheFly* contains videos that have been obtained in controlled environment simulating the behaviour of travellers crossing an ABC system and serves as baseline reference. The second one is called *FRAV-ABC-RB-OnTheFly*. It comprises travellers in real border crossing. Both situations have produced two databases that are detailed in the next sections.

4.1. Baseline controlled scenario database

The *FRAV-ABC-OnTheFly* database includes 178 subjects and contains 150 videos. These videos have 25 fps (frames per second) and $1,920 \times 1,080$ pixel resolution (see Fig.4).

Database is formed by 82 women and 96 men. Ages range between 18 and 67 years with approximately 70 percent of the subjects in an age extent between 18 and 28 years. Database subject selection were done according to the border crossing statistics mimicking its distribution. Database was built with voluntary students, teachers and university staff. All subjects keep their privacy and, according to data protection regulation, informed consent were required from all subjects. The *bona fide* acquisition was carried out over a week and later, after constructing the PAIs of all the subjects, the capture of the videos of attacks was carried out in two days.

The videos of the database were captured in the laboratory under controlled illumination conditions. These videos were recorded by using a high resolution standard camera.

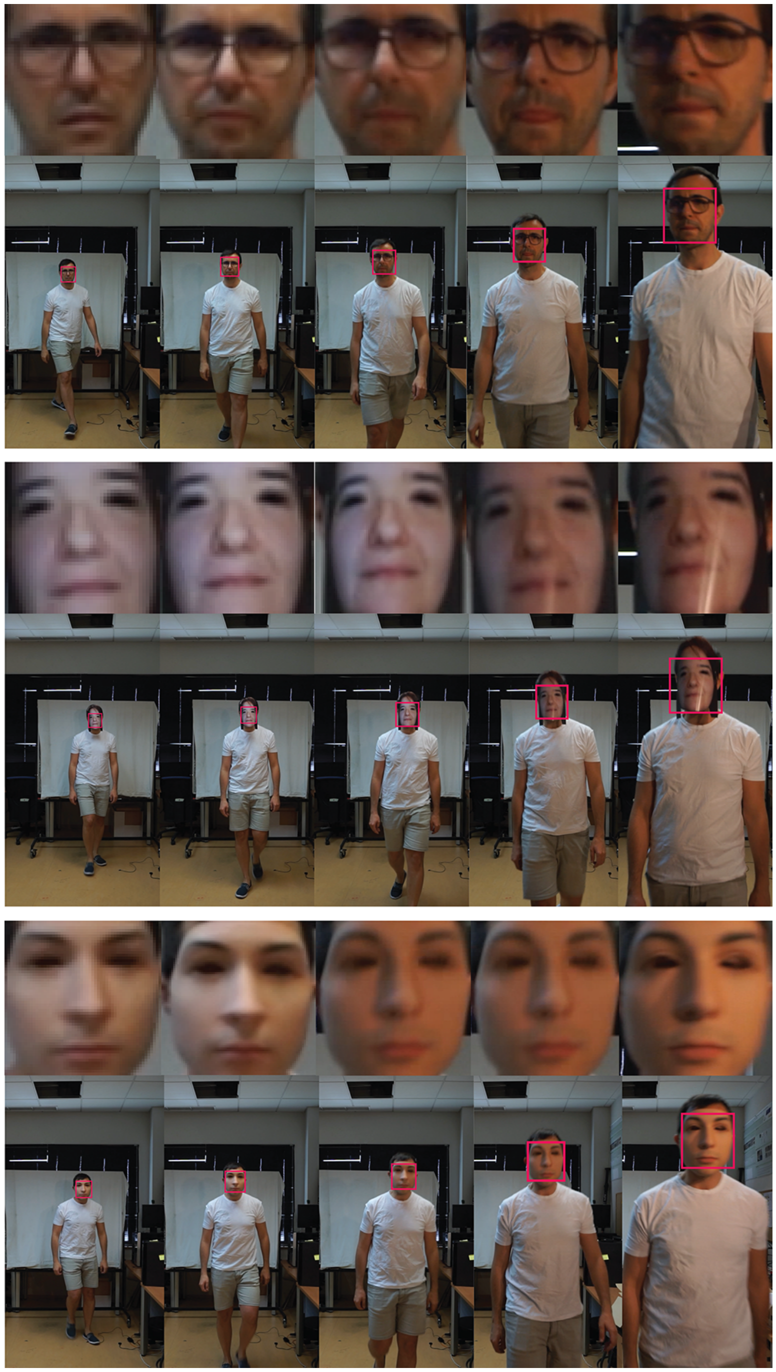


Figure 4: Stored information in the *FRAV-ABC-OnTheFly* database.

FRAV-ABC-OnTheFly (178 subjects - 1,068 videos - 120,425 faces)						
Range	**Bona fide**	**Photo**	**Video**	**Mask**	**Mask w/e**	**3D Mask**
Range-1	8,799	8,525	8,330	8,203	8,329	412
Range-2	8,837	8,730	8,750	8,335	8,420	441
Range-3	8,843	8,603	8,603	8,208	8,225	435

Table 2: Number of detected faces from the *FRAV-ABC-OnTheFly* database videos.

FRAV-ABC-RB-OnTheFly (10 subjects - 60 videos - 7,200 faces)						
Range	**Bona fide**	**Photo**	**Video**	**Mask**	**Mask w/e**	**3D Mask**
Range-1	440	402	318	350	347	408
Range-2	481	411	421	417	428	423
Range-3	403	383	370	403	408	387

Table 3: Number of detected faces from the *FRAV-ABC-RB-OnTheFly* database videos.

The subjects were taught to simulate the behaviour of travellers in a crossing border. Thus, subjects walk starting at 3 meters from the camera until half a meter. Six different videos following this procedure (one per type of attack and one for the *bona fide*) were produced (see Table 2).

Processing videos is necessary to retrieve training images. Faces are detected in each video frame through the Viola-Jones algorithm. Each located face is labelled with the appropriate distance range (i.e. Range-1, Range-2 and Range-3) according to its size.

Finally, the database information is stored by the three defined ranges. Notice that there are different amount of frames with faces because sometimes a face could not be detected in all the frames.

4.2. Real border scenario database

The *FRAV-ABC-RB-OnTheFly* database uses video data captured during the implemented pilots related to the European Project ABC4EU [42]. The

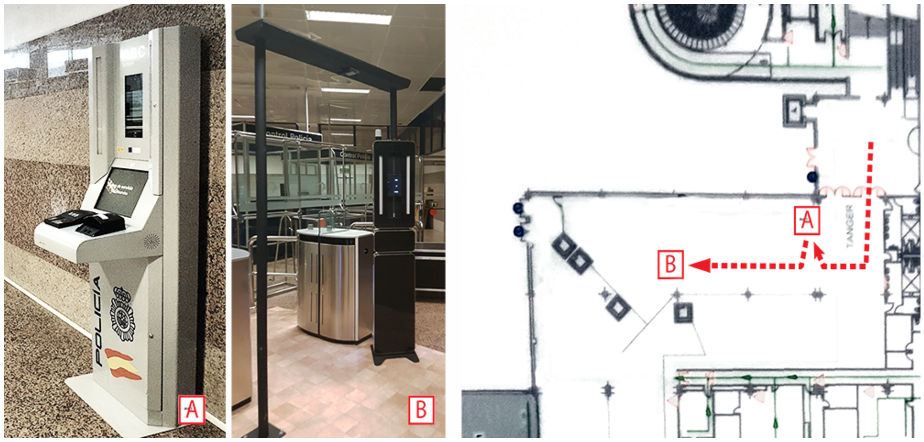


Figure 5: Real scenario with the deployed e-kiosk and e-gate devices.

seaport of Algeciras (Spain) was the selected real scenario. This seaport is a frontier in the Schengen area, being an arrival and departure point for North-African travellers.

The real scenario was the traveller reception hall of the seaport. This scenario consists of two devices: one e-kiosk and one e-gate (see Fig. 5). In the e-kiosk, travellers were able to register showing their documentation. The e-gate completed the deployment being in charge of evaluating the registered crossing users. The e-gate includes the *FlyPAD* framework to achieve the task.

A total of 10 travellers were selected to implement the selected PAIs. They were recorded using a camera with $1,920 \times 1,080$ pixel resolution. Analogously to the *FRAV-ABC-OnTheFly* database, each one of them presents a *bona fide* video and five videos related to the different PAIs at the three different distance ranges (see Table 3).

Subjects of the real frontier database were provided from EU project ABC4EU, and were 5 women and 5 men, aged between 22 and 56 years. As in the controlled environment, data protection regulation has to be applied to protect the personal information of travellers. Also, informed consent should be signed and maintained. Elaborated PAIs such as facial 3D masks were manufactured for these subjects.

5. Experiments

This section presents a set of experiments achieved to illustrate the viability of the *FlyPAD* framework. First, specific metrics to evaluate the *PAD* module are introduced. Then, Support Vector Machines (SVM) [43] models have been included to achieve the classification tasks of the *models repository* module. These classifiers have been trained using the 70% of the videos stored in the *FRAV-ABC-OnTheFly* database (i.e. laboratory environment). Next, the results of the framework with the remaining 30% of videos from these database are presented. Finally, the performance of the system is evaluated with the complete *FRAV-ABC-RB-OnTheFly* database (real environment). In both situations, results with and without the *tracking* module activated are included (i.e. dynamic and static situations).

5.1. Specific metrics to evaluate PAD systems

Standard metrics related to PAD systems [30, 44] have been used to evaluate the *PAD* module of the framework. In particular, the Bona fide Presentation Classification Error Rate (BPCER) is defined as the ratio of *bona fide* presentations misclassified as attacks. It measures the proportion of times that users present their own biometric data to the system in a collaborative way but a presentation attack alarm appears. For a particular experimental scenario, let N_{BF} be the total amount of *bona fide* presentations and Res_i the response of the PAD system (where i is a specific *bona fide* presentation, $1 \leq i \leq N_{BF}$). The value of Res_i is 0 if the i presentation is correctly classified as a *bona fide* presentation, while it is 1 if it is wrongly classified as a presentation attack. As the attacks can be very diverse, they can be grouped according to their PAI. Then, the BPCER for a given PAI species PAIs is computed as follows:

$$BPCER_{PAIs} = \frac{\sum_{i=1}^{N_{BF}} Res_i}{N_{BF}}, \tag{1}$$

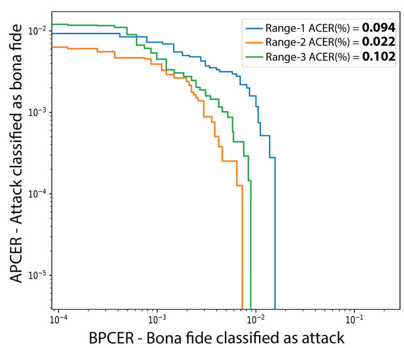
However, if the user tries to deceive the system by providing fake, manipulated or disguised biometric features, this can be considered as spoofing or a presentation attack. In this case, the Attack Presentation Classification Error Rate (APCER) stands for the ratio of presentation attacks misclassified as *bona fide* presentations. This value is key as far as security is concerned, as it represents how many malicious users can break into the system without being detected. Let N_{PAIs} be the number of times a specific type of PAI is used to attack the system. Again,let Res_i be the response of the PAD system (where i is a specific attack presentation, $1 \leq i \leq N_{PAIs}$). It takes the value 0 if the i presentation is wrongly classified as a *bona fide* presentation, while it is 1 if it is correctly classified as a presentation attack. Then, APCER for a given PAI species PAIs is computed as:

$$APCER_{PAIs} = 1 - \left(\frac{1}{N_{PAIs}}\right) \sum_{i=1}^{N_{PAIs}} (Res_i)\,, \tag{2}$$

A Detection Error Tradeoff (DET) curve can be computed by plotting APCER versus BPCER for a range of threshold values of the classifier. As it is usual in biometrics, it is not possible to minimise both error rates at the same time. The values of BPCER and APCER are interrelated with the former increasing while the latter decreases, and vice versa. Both measures describe the performance of a PAD system, which will be better as these ratios are as low as possible. One way to express the performance of a system with a single value is by means of the so called Average Classification Error Rate (ACER), defined as the mean of the APCER and BPCER values, for a specific type of PAI. A reliable PAD system should have an ACER as low as possible. Thus, ACER as well as the corresponding BPCER and APCER values are presented in the experimental results.

5.2. Baseline controlled scenario results

Figure 6 shows the *PAD* module and *tracking* module results using the *FRAV-ABC-OnTheFly* database. Figure 6b presents the results when processing the videos using the complete system. That is, taking into account the tracking and detection failures, and the detected attacks.



(a) *PAD* module results per ranges

		tracking frames		
		15 frames	30 frames	60 frames
% bonafides frames	10 %	89.5%	83.2%	75.5%
	20 %	79.8%	72.2%	70.1%
	30 %	71.1%	68.8%	60.2%

(b) *Tracking* module accuracy

Figure 6: *PAD* module and *tracking* module results using the *FRAV-ABC-OnTheFly* database videos.

The system precision depends on the number of consecutive frames detected that are considered as a valid tracking, and on the percentage of frames considered as *bona fide* needed to label the complete tracking as *bona fide*. For instance, the system gets an accuracy of 72.2%, when it is established that a tracking must have at least 30 consecutive frames, and *bona fide* tracking is allowed only if 20% of those frames are *bona fide*. However, the accuracy is only 60.2%, when it is established that a tracking must have at least 60 consecutive frames and *bona fide* tracking is allowed only if 30% of those frames are *bona fide*.

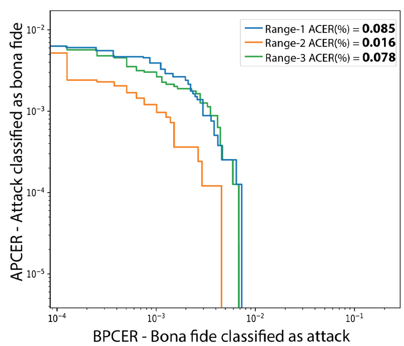
When dealing with ABC systems in which safety prevails, it is convenient to increase the percentage of frames considered as *bona fide*. Regarding the number of frames needed to consider a complete tracking, 15 is a reliable choice for a distance of 3 meters. Under this configuration, the system achieves an accuracy of 71.1%.

Figure 6a presents the system results ignoring the tracking information. Only the *PAD* module is considered. To achieve these results, all the faces in the *FRAV-ABC-OnTheFly* database have been segmented and the range in which the face is found has been estimated. In addition, the classifiers of the *models*

Range	**PAI**	**APCER(%)**	**BPCER(%)**	**ACER(%)**
Range-1	Photo	0.180	0.112	0.146
	Mask	0.425	0.062	0.243
	Mask w/o eyes	0.280	0.152	0.216
	Video	0.075	0.385	0.422
	3D Mask	0.412	0.022	0.217
	All attacks	**0.102**	**0.086**	**0.094**
Range-2	Photo	0.033	0.052	0.042
	Mask	0.044	0.085	0.064
	Mask w/o eyes	0.098	0.015	0.056
	Video	0.093	0.102	0.097
	3D Mask	0.091	0.013	0.052
	All attacks	**0.026**	**0.019**	**0.022**
Range-3	Photo	0.103	0.052	0.077
	Mask	0.345	0.090	0.217
	Mask w/o eyes	0.141	0.285	0.213
	Video	0.258	0.281	0.269
	3D Mask	0.587	0.030	0.308
	All attacks	**0.098**	**0.106**	**0.102**

Table 4: *PAD* module results by range and by PAI using the *FRAV-ABC-OnTheFly* database videos.

repository module corresponding to this range have been selected. The DET curves with the APCER and BPCER errors obtained in each range show that the Range-2 is the one with the lowest ACER error rate, indicating that the distance at which errors are best detected is an intermediate distance greater than 1 meter and lower than 2 meters. Analysing in detail the images of each range, although the error rates are low, it can be seen that the images that are more than 2 meters away have little result for a PAD system based on textures and that the images too close to the capture device are too noisy and have too



(a) *PAD* module results per ranges

% bonafides frames	tracking frames: 15 frames	30 frames	60 frames
10 %	95.2%	88.0%	85.9%
20 %	92.1%	85.2%	78.3%
30 %	76.2%	65.2%	64.9%

(b) *Tracking* module accuracy

Figure 7: *PAD* module and *tracking* module results using the *FRAV-ABC-RB-OnTheFly* database videos.

many artefacts due to such a close capture.

Table 4 shows the APCER and BPCER rates and the ACER of each of the repository classifiers for a given range and with a given PAI. It is confirmed that the lowest error rates are those of the Range-2. Likewise, it can be seen that the best detected attack is the photo attack. Video and 3D mask attacks are the most difficult ones to be detected.

5.3. Real border scenario results

Figure 7 presents the system results by processing the videos from the *FRAV-ABC-RB-OnTheFly* database, which was obtained into the real border crossing scenario.

Although some conditions such as lighting are not controlled, the results are similar to those achieved on the *FRAV-ABC-OnTheFly* database.

As indicated above, the ABC systems require high level of security. Although higher precision values are achieved by considering a lower percentage of frames detected as *bona fide*, it is convenient to make the system more restrictive since false negatives (not allow access to a *bona fide* traveller) can be manually corrected by security agents. As before, a 30% *bona fide* frames of a tracking of at least 15 consecutive frames, is a good choice and guarantees a 76.2% accuracy.

Range	PAI	APCER(%)	BPCER(%)	ACER(%)
Range-1	Photo	0.144	0.100	0.123
	Mask	0.416	0.050	0.234
	Mask w/o eyes	0.252	0.123	0.188
	Video	0.058	0.382	0.220
	3D Mask	0.403	0.000	0.202
	all attacks	**0.092**	**0.070**	**0.085**
Range-2	Photo	0.024	0.037	0.031
	Mask	0.037	0.063	0.050
	Mask w/o eyes	0.063	0.024	0.043
	Video	0.116	0.074	0.095
	3D Mask	0.050	0.000	0.025
	all attacks	**0.018**	**0.020**	**0.016**
Range-3	Photo	0.096	0.037	0.067
	Mask	0.340	0.100	0.221
	Mask w/o eyes	0.126	0.296	0.211
	Video	0.290	0.234	0.262
	3D Mask	1.209	0.000	0.605
	all attacks	**0.080**	**0.078**	**0.078**

Table 5: *PAD* module results by range and by PAI using the *FRAV-ABC-RB-OnTheFly* database videos.

Figure 7a shows the curve with the APCER and BPCER error rates by range in the real scenario. Although error rates are low in all three ranges, the range with the lowest detection error is the Range-2. As in the first case, the loss of quality of images too far away (and too close) from the capture device penalises the performance of the *PAD* module. This is also clear from the results presented in Table 5 where the performance of each classifier in the *models repository* with the segmented faces in the real environment is detailed.

These experiments show the viability of the *FlyPAD* framework. The system achieves accuracy values very similar to other systems which work in real scenarios in static situations [45]. It could be set that the proposal successfully works both in static and dynamic situations. This issue leads to thinking in a future implantation of the prototype in cross borders, being able to simplify the flow of travellers (i.e. more effective crossing-times). This is directly related to its ability to perform its detection tasks using less intrusive biometric procedures.

6. Conclusions

This paper has presented the *FlyPAD* framework. It is a system able to carry out PAD dynamically while the individuals are moving. It covers five different types of attack related to face detection: printed photos, paper masks, paper masks without eyes, screen videos, and 3D masks.

The system comprises four modules: the main one is the *tracking* module which generates a token with the information of a tracked individual. It is supported with the *detection* module, the *verification* module, the *PAD* module. This latter uses different Machine Learning models previously trained for three different acquisition distances to perform the face attack detection.

Several experiments in a controlled environment (i.e. the laboratory) and in a real environment (i.e. an ABC system in a border crossing) have been developed in order to test the proposal. The obtained results allow concluding that *FlyPAD* framework is able to detect *On-the-fly* (i.e. dynamically) possible presentation attacks. This detection can be configured according to a threshold in order to reduce the number of false positives, increasing the robustness of the system in a real environment. Regarding the three acquisition distances, the worse results were for Range-3, the closest to the detector. In this case, the images have a larger resolution and yield a higher ACER compared to the other intervals. This is related to the textures computed in the LBP algorithm, which can vary too much for such a scale.

As a general conclusion, the results obtained in the real environment are better than those obtained in the laboratory, probably because the laboratory tests have been more exhaustive and also because more samples were available. Moreover, it has been detected that certain attacks directly lose their effectiveness in *On-the-Fly* approach since at certain distances the faces of the individuals are not detected. This issue disables these situations as possible attacks.

The system is a functional prototype which has been successfully tested. Nevertheless, some future guidelines are interesting to validate and also enhance its capabilities. For instance, it can be included attacks with silicone masks [25] [27] and also implement other PAD algorithms that consider temporal information and spatial features. Long Short-Term Memory (LSTM) networks [46] could be interesting at this point.

Acknowledgments

This work was partially supported by: Spanish Ministry of Economy research project BIOINPAD (ref TIN2016-80644-P) and European Commission 7FP project ABC4EU (grant agreement 312797). We specially thanks the support of the Spanish General Directorate of Police.

References

[1] E. Commission, Managing migration in all its aspects: Progress under the european agenda on migration, Report 798, European Commission, [Online: accessed 14-Oct-2019] (2018).

[2] U. S. Customs, B. Protection, Cbp snapshot of operations, Report, United States Customs and Border Protection, [Online: accessed 14-Oct-2019] (2018).

[3] A. Morse, The uk border, Report, National Audit Office, [Online: accessed 14-Oct-2019] (2017).

[4] J. S. del Rio, D. Moctezuma, C. Conde, I. M. de Diego, E. Cabello, Automated border control e-gates and facial recognition systems, computers & security 62 (2016) 49–72.

[5] R. Ramachandra, C. Busch, Presentation attack detection methods for face recognition systems: A comprehensive survey, ACM Computing Surveys (CSUR) 50 (1) (2017) 8.

[6] R. Raghavendra, B. Yang, C. Busch, Robust on-the-fly person identification using sparse representation, in: 2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), IEEE, 2013, pp. 1–4.

[7] R. FRONTEX, Best practice operational guidelines for automated border control (abc) systems, European Agency for the Management of Operational Cooperation, Research and Development Unit 9 (5) (2012) 2013.

[8] B. Didier, E. Guild, The EU Counter-Terrorism Policy Responses to the Attacks in Paris. Towards an EU Security and Liberty Agenda, Brussel: CEPS, 2015.

[9] R. D. Labati, A. Genovese, E. Muñoz, V. Piuri, F. Scotti, G. Sforza, Biometric recognition in automated border control: a survey, ACM Computing Surveys (CSUR) 49 (2) (2016) 24.

[10] IATA, Automated border control implementation, `https://www.iata.org/whatwedo/passenger/Pages/automated-border-control-maps.aspx`, [Online: accessed 14-Oct-2019] (2018).

[11] C. Morosan, Information disclosure to biometric e-gates: The roles of perceived security, benefits, and emotions, Journal of Travel Research 57 (5) (2018) 644–657.

[12] R. Donida Labati, A. Genovese, E. Muñoz Ballester, V. Piuri, F. Scotti, G. Sforza, Emerging biometric technologies for automated border control gates, in: PRIP, IAPR, 2016, pp. 1–6.

[13] A. Anand, R. D. Labati, A. Genovese, E. Munoz, V. Piuri, F. Scotti, G. Sforza, Enhancing fingerprint biometrics in automated border control with adaptive cohorts, in: 2016 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, 2016, pp. 1–8.

[14] R. Nguon, J. Lundberg, E. R. Swanson, System and method of border detection on a document and for producing an image of the document, uS Patent 9,729,744 (Aug. 8 2017).

[15] N. M. Abdal-Ghafour, A. A. Abdel-Hamid, M. E. Nasr, S. A. Khamis, Authentication enhancement techniques for bac in 2g e-passport, in: 2016 12th International Conference on Innovations in Information Technology (IIT), IEEE, 2016, pp. 1–6.

[16] R. Nieto-Gomez, Walls, sensors and drones: Technology and surveillance on the us-mexico border, Borders, Fences and Walls: State of Insecurity (2014) 191–210.

[17] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, J. Ortega-Garcia, Presentation attacks in signature biometrics: Types and introduction to attack detection, in: Handbook of Biometric Anti-Spoofing, Springer, 2019, pp. 439–453.

[18] U. Scherhag, R. Raghavendra, K. B. Raja, M. Gomez-Barrero, C. Rathgeb, C. Busch, On the vulnerability of face recognition systems towards morphed face attacks, in: 2017 5th International Workshop on Biometrics and Forensics (IWBF), IEEE, 2017, pp. 1–6.

[19] C. Sousedik, C. Busch, Presentation attack detection methods for fingerprint recognition systems: a survey, Iet Biometrics 3 (4) (2014) 219–233.

[20] K. B. Raja, R. Raghavendra, C. Busch, Color adaptive quantized patterns for presentation attack detection in ocular biometric systems, in: Proceedings of the 9th International Conference on Security of Information and Networks, ACM, 2016, pp. 9–15.

[21] C. Kraetzer, A. Makrushin, T. Neubert, M. Hildebrandt, J. Dittmann, Modeling attacks on photo-id documents and applying media forensics for the detection of facial morphing, in: Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security, ACM, 2017, pp. 21–32.

[22] U. Scherhag, C. Rathgeb, C. Busch, Towards detection of morphed face images in electronic travel documents, in: 2018 13th IAPR International Workshop on Document Analysis Systems (DAS), IEEE, 2018, pp. 187–192.

[23] J. Hernandez-Ortega, J. Fierrez, A. Morales, J. Galbally, Introduction to face presentation attack detection, in: Handbook of Biometric Anti-Spoofing, Springer, 2019, pp. 187–206.

[24] X. Li, J. Komulainen, G. Zhao, P.-C. Yuen, M. Pietikäinen, Generalized face anti-spoofing by detecting pulse from face videos, in: 2016 23rd International Conference on Pattern Recognition (ICPR), IEEE, 2016, pp. 4244–4249.

[25] I. Manjani, S. Tariyal, M. Vatsa, R. Singh, A. Majumdar, Detecting silicone mask-based presentation attack via deep dictionary learning, IEEE Transactions on Information Forensics and Security 12 (7) (2017) 1713–1723.

[26] S. Bhattacharjee, A. Mohammadi, S. Marcel, Spoofing deep face recognition with custom silicone masks, in: 2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS), IEEE, 2018, pp. 1–7.

[27] A. Agarwal, D. Yadav, N. Kohli, R. Singh, M. Vatsa, A. Noore, Face presentation attack with latex masks in multispectral videos, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2017, pp. 81–89.

[28] C. Chen, A. Dantcheva, T. Swearingen, A. Ross, Spoofing faces using makeup: An investigative study, in: 2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA), IEEE, 2017, pp. 1–8.

[29] I. O. for Standardization, ISO/IEC FDIS 30107-3:2017: Information technology -Biometric presentation attack detection Part 1: Framework, Switzerland, Geneva, 2017, [Online: accessed 14-Oct-2019].

[30] S. Marcel, M. S. Nixon, S. Z. Li, Handbook of biometric anti-spoofing, Vol. 1, Springer, 2014.

[31] Y. Shoukry, P. Martin, Y. Yona, S. Diggavi, M. Srivastava, Pycra: Physical challenge-response authentication for active sensors under spoofing attacks, in: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, ACM, 2015, pp. 1004–1015.

[32] I. Rigas, O. V. Komogortsev, Eye movement-driven defense against iris print-attacks, Pattern Recognition Letters 68 (2015) 316–326.

[33] D. Tiwari, V. Tyagi, Dynamic texture recognition: a review, in: Information Systems Design and Intelligent Applications, Springer, 2016, pp. 365–373.

[34] M. Jones, P. Viola, Fast multi-view face detection, Mitsubishi Electric Research Lab TR-20003-96 3 (14) (2003) 2.

[35] C. Systems, cognitec, `https://www.cognitec.com/`, [Online: accessed 14-Oct-2019] (2019).

[36] J.-C. Chen, V. M. Patel, R. Chellappa, Unconstrained face verification using deep cnn features, in: 2016 IEEE winter conference on applications of computer vision (WACV), IEEE, 2016, pp. 1–9.

[37] F. W. Brom, M. Besters, greedyinformation technology: The digitalization of the european migration policy, European Journal of Migration and Law 12 (4) (2010) 455–470.

[38] S. K. Karanja, Transparency and proportionality in the Schengen information system and border control co-operation, Vol. 32, Martinus Nijhoff Publishers, 2008.

[39] T. Ojala, M. Pietikäinen, T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Transactions on Pattern Analysis & Machine Intelligence (2002) 971–987.

[40] I. Chingovska, A. Anjos, S. Marcel, On the effectiveness of local binary patterns in face anti-spoofing, in: 2012 BIOSIG-proceedings of the international conference of biometrics special interest group (BIOSIG), IEEE, 2012, pp. 1–7.

[41] K. Goto, K. Kidono, Y. Kimura, T. Naito, Pedestrian detection and direction estimation by cascade detector with multi-classifiers utilizing feature interaction descriptor, in: 2011 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2011, pp. 224–229.

[42] ABC4EU, Abc4eu fp7 research projec, `http://abc4eu.com`, [Online: accessed 14-Oct-2019] (2014).

[43] Y. Ma, G. Guo, Support vector machines applications, Springer, 2014.

[44] I. O. for Standardization, ISO/IEC FDIS 30107-3:2017: Information technology -Biometric presentation attack detection Part 3: Testing and reporting, Switzerland, Geneva, 2017, [Online: accessed 14-Oct-2019].

[45] J. Komulainen, Z. Boulkenafet, Z. Akhtar, Review of face presentation attack detection competitions, in: Handbook of Biometric Anti-Spoofing, Springer, 2019, pp. 291–317.

[46] Z. Xu, S. Li, W. Deng, Learning temporal features using lstm-cnn architecture for face anti-spoofing, in: 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR), IEEE, 2015, pp. 141–145.