Domain-Specific Adaptation of CNN for Detecting Face Presentation Attacks in NIR

Ketan Kotwal, Sushil Bhattacharjee, Philip Abbet, Zohreh Mostaani,
Huang Wei, Xu Wenkang, Zhao Yaxi, and Sébastien Marcel

Abstract—For the automotive industry moving towards personalized applications and experiences, the identification of the person inside vehicle is necessary; and it must be carried out in a secure manner. In this paper, we propose a unique face presentation attack detection (PAD) system for operation inside a passenger vehicle. A typical *in-vehicular* face PAD system is required to function with several constraints such as bounded sensing (imaging) capabilities, limited computing resources on embedded devices, real-time inference, and essentially, very high accuracy. In this work, we develop a face PAD system for automotive domain, relying on a single NIR camera, to continually verify whether the driver's face is *bona-fide* or not. Our work has two main contributions: first, a lightweight face PAD framework has been developed using a 9-layer convolutional neural network (CNN). With its compact size and limited set of operators, it can be deployed in a resource constrained embedded device to achieve a near real-time inference. To alleviate the problem of limited training data (face PAD in NIR) for a given system, we develop an efficient mechanism to obtain this CNN through the combination of adaptation of domain-specific layers and task-specific fine-tuning of a base CNN. As the second contribution, we collect a large face PAD dataset with 5800+ videos, acquired in NIR (940 *nm*) illumination, for *in-vehicular* use-cases. This dataset, named VFPAD, captures several real-world variations in terms of environmental settings, illumination, subject's pose, and appearances. Based on the VFPAD dataset, we demonstrate that the proposed face PAD method achieves very high performance (overall accuracy $\approx$ 98.0%), and also outperforms several baseline face PAD methods. The dataset will be shared with the wider scientific community for research purposes.

Index Terms—CNN, Face presentation attack detection (PAD), near-infrared, domain adaptation.

1 Introduction

Face recognition (FR) technology, alongside its technical advancements, is finding new consumer applications beyond the realms of access control. FR systems are now also being introduced into the domain of automotive applications [1]. Current applications of FR technology in cars revolve mainly around personalization (*e.g.*, [2]) and driving safety [3]. The next generation of biometric applications being developed in the automotive domain will include integrated services- that will leverage the identity authentication capabilities built into the vehicle in the form of FR and other biometric technologies. However, robustness and security are two important concerns that must be addressed for such applications to be truly useful. Hence, an efficient mechanism for detection of *spoofing* attempts or presentation attacks (PAs) on FR system is an indispensable component of the aforementioned class of automotive applications.

A classic example of such use-case is the companies offering home delivery services—where the customer receives a doorstep delivery of the goods, cloths, food items, etc.; and makes the payment using cash or card on the receipt. In this whole process where the end transaction takes place at remote locations, it is crucial that the person involved in the delivery as well as payment of each transaction is correctly identified. It helps the companies to ensure safe and trustworthy delivery experience, and also to avoid frauds related to impersonations. In this classic scenario of a distributed PoS (point of sale), each PoS is typically in the form of a vehicle (delivery truck or car). Figure 1 shows a typical attempt of *spoofing* the FR system (mounted on steering wheel or dashboard) where the subject presents another identity for authentication. A secure and robust identity management can be accomplished by installing an FR system in each PoS, coupled with a face PAD system to assure that the FR system is being presented with the genuine (*bona-fide*) identity of the subject.

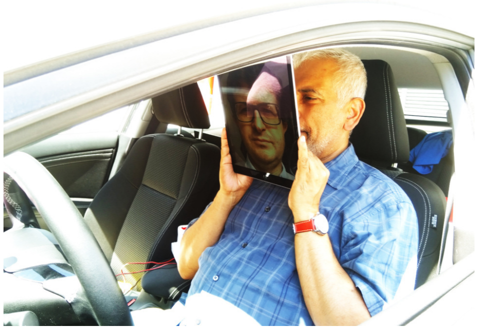


Figure 1. A user attempting to *spoof* the FR system inside the car by presenting a fake identity of another subject. The camera, mounted on the steering wheel, is intended to capture facial region of the person in the driver's seat; while the user presents a tablet replaying a video clip of another person's face.

In this work we propose a face PAD system for deployment in a passenger vehicle. Such a face PAD system is expected to function in a resource-limited environment,

- K. Kotwal, S. Bhattacharjee, P. Abbet, Z. Mostaani, and S. Marcel are with Idiap Research Institute, Martigny, Switzerland.
- H. Wei, X. Wenkang, and Z. Yaxi are with Huawei Technologies Co., Ltd.

similar to that on current mobile devices; while providing the inference in continuous, near real-time manner. Most mobile computing platforms include a Trusted Execution Environment (TEE) that provides a safe area for executing trusted applications (TA). The TEE guarantees the security, confidentiality, and integrity of the code and data loaded into the environment. In mobile computing parlance, the TEE is also referred to as the *Secure World*, in contrast to the *Normal World* (which may include a Rich Execution Environment (REE)). Most mobile applications run in the Normal World, whereas security-sensitive applications, such as biometric authentication systems, run in the Secure World. In comparison to the REE, the TEE offers a limited set of computing operators. These computing constraints influence the design of the appropriate face PAD solution.

Besides accuracy, the face PAD model must be small in terms of memory footprint, and fast in processing or inference. The state-of-the-art methods of face PAD are based on the use of deep convolutional neural networks (CNNs). Several popular face PAD methods employ a well-known architecture as a base CNN (often referred to as backbone) which is then suitably adapted for the task of face PAD. The architectures such as VGG [4] and ResNet [5] are used as backbone networks for state-of-the-art face PAD methods in [6], [7], [8], [9], [10]. Although efficient, these backbone CNNs are large in size and comprise many layers, resulting in slow processing and heavy storage. A 16-layer VGG network consists of as many as 135M parameters. For different variants of ResNet, these numbers are in the range of 11–58M. The number of layers in these CNNs also impact the processing time unless the system is highly optimized. For the face PAD system to function at real-time speeds in the TEE, the PAD CNN model should be implemented using a limited set of operators, and it should be relatively small in terms of number of parameters.

A primary limitation in developing a face PAD system for the aforementioned *in-vehicular* environment is the lack of appropriate datasets. Most of the publicly available PAD datasets are collected in limited laboratory settings; and hence, these are far from being realistic. For example, in the present use-case, the person in driver's seat may wear glasses, sunglasses, or hat– which occlude a part of their face. Due to limited options of installing camera (such as vehicle's dashboard), one may not obtain an absolutely frontal orientation of the person's face. An uneven illumination is a well-known problem for FR and face PAD. Since the delivery vehicle can be at different locations, including indoor ones, the effect of outside illumination (which often impacts from one window of the vehicle) also requires specific attention. The existing face PAD datasets do not cover such multitude of variations, and thus, their applicability towards developing a solution to the real world problem is limited. It should also be noted that majority of existing PAD datasets, and therefore, face PAD methods involve presentations acquired in visual spectra (grayscale or RGB). The extended spectra, such as near infrared (NIR), offer several advantages over visual domain [11], [12]; and thus it can be a better choice for such real applications, provided a low-cost sensor is available.

In this work, we address the real-world problem of PAD for automotive domain through two steps: creation of PAD dataset that covers many varied scenarios of in-vehicle use-case with several *bona-fide* presentations and combinations of different PAs. This dataset is acquired in NIR imaging channel. Secondly, we build a small size, lightweight face PAD CNN using a 9-layer LightCNN [13] for detection of PAs in an automobile. Note that the base CNN is trained for FR task using an FR dataset captured in visual domain (due to availability of large training data); whereas the PAD CNN is required to detect PAs acquired NIR channel. We propose a combination of domain-specific adaptation and task-specific fine-tuning to obtain the CNN for NIR-based Face PAD. We also ensure that the proposed CNN is built using a limited set of CNN operators so that it can be deployed in generic hardware (such as TEE), and can also be easily optimized. The effectiveness of proposed PAD method is evaluated on the newly collected unique VFPAD dataset.

The specific contributions of our work can be summarized as follows:

- A lightweight CNN architecture for continuous (near real-time) face PAD for presentations acquired in near infrared (NIR) imaging channel. The CNN consists of 5.5M parameters in 9 layers. With a limited set of neural network operators, it may be easily optimized further for specific hardware, such as TEE.
- We provide a simple, yet efficient mechanism through the combination of domain adaptation and fine-tuning that can be employed to obtain a lightweight CNN for face PAD in NIR data, whereas the base CNN is pretrained for FR task on RGB presentations.
- A new *in-vehicular* NIR dataset with 5800+ videos captured in a large variety of conditions with respect to illumination, pose, accessories, etc. To the best of our knowledge, this is the first of its kind, publicly available PAD dataset acquired in NIR.
- Through performance evaluation over several baseline PAD methods, we demonstrate that the proposed PAD method yields state-of-the-art results.

After a discussion on published scientific literature relevant to the current work in Section 2, we describe the VFPAD dataset in Section 3. Details of the experimental methodology are described in Section 4. The experimental details and results of PAD are provided in Sections 5 and 6, respectively. Conclusions are discussed in Section 7.

2 Related Work

For a face PAD in in-vehicular environment, there is no previously published work that may be directly considered as a precedent. In this section, therefore, we discuss some related research works that conform to various processing steps of our specific face PAD problem. In recent years, research in face PAD has seen two significant developments-first, the shift from color (RGB) imagery to other wavelength bands (NIR, SWIR, and thermal imaging); and second, the use of features derived from CNNs instead of hand-crafted features [14], [15]. Accordingly, we discuss the state of research in face PAD based on extended range imagery, followed by brief details of some commonly used face PAD datasets acquired in NIR imaging channel; and provide an overview of recent CNN architectures for face PAD based on NIR data.

2.1 Extended-Range Imagery for Face-PAD

We start by noting that presentation attack instruments (PAI) are usually designed to mimic the appearance of human faces in visible light (*i.e.,* frequency bands roughly in the range from 380 to 750 *nm*, often also referred to as RGB). Such PAIs, however, do not always present the same characteristics as the human face under illumination beyond the visible light range. On the other hand, with improvement in technology, the quality of PAs in visible light is approaching that of *bona-fide* presentations. This realization has led to innovations in face PAD research based on extended range (ER) imagery. In recent years, both active and passive-sensing approaches for face PAD in ER imagery have been explored.

Raghavendra *et al.* [16] have used 7-band multispectral imagery for face PAD, captured using a `SpectroCam`™ multispectral camera. This device captures presentations in narrow bands centered at the wavelengths ranging from 425 *nm*–930 *nm*. They have proposed two face PAD approaches based on image fusion and score fusion. Quantitative results [16] show that the score fusion approach performs significantly better than the image fusion approach.

Bhattacharjee and Marcel [11] have also investigated the use of ER imagery for face PAD. They demonstrate that a large class of 2D attacks, specifically, video replay attacks, can be easily detected using NIR imagery. In live presentations under NIR illumination, the human face is clearly discernible. However, electronic display monitors appear almost uniformly dark under NIR illumination. Therefore, using NIR imagery, simple statistical measures are often sufficient to distinguish between *bona-fide* presentations and PAs. For photo-based PAs, printed on certain class of printers, and for 3D mask-based PAs, the use of NIR towards face PAD is not straightforward, and an advanced machine learning approach is necessary.

Most face PAD studies involving NIR imagery have, infact, explored the combination of RGB and NIR channels. Liu and Kumar [17] demonstrate the superiority of NIR over visible light for detecting 3D-mask based PAs. They consider various CNN configurations for face PAD including a Siamese network. They also show that a combination of RGB and NIR data can further improve the performance of PAD. In [12], Kotwal *et al.* have also demonstrated that combining RGB and NIR image data improves the performance of face PAD, compared to face PAD based on RGB imagery alone. Jiang *et al.* [18] have proposed a generative adversarial network (GAN)-based approach for synthesizing an NIR image from an RGB image, for situations where it is not possible to deploy an NIR camera. Li *et al.* [19] have used a binocular camera that combines an RGB sensor and an NIR sensor along with NIR (850 *nm*) illumination. The RGB and NIR face images captured by the camera are stacked together (after adequate registration) to construct a multi-channel input image for the feature extraction stage. Agarwal *et al.* [20] tackle the problem of detecting obfuscation using 3D flexible masks using multispectral imagery through combining images captured under visible light, NIR, and LWIR wavelength bands. Their experiments, based on a variety of handcrafted local texture descriptors, show that thermal imagery is best suited for reliable detection of masks. Hernandez-Ortega *et al.* have demonstrated the use of remote photo-plethysmography (rPPG) for detection of liveliness [21]. They extracted the rPPG signal from the face region of NIR videos of about 10 *s* duration, followed by an SVM classifier.

The works discussed in this section demonstrate that NIR is a viable imaging channel for face PAD applications. Several of these studies have demonstrated that other wavelength bands, such as SWIR and LWIR, may be even more effective than NIR for face PAD. Capturing SWIR and LWIR data, however, often involves very expensive sensors. NIR sensors today are significantly cheaper, even than low cost thermal sensors of comparable image resolution. These observations give us confidence that using an NIR sensor to design a face PAD system strikes the right balance between cost and efficacy.

Although ER imagery offers significant advantages for detection of face PAs, a majority of real-world face PAD systems acquire only visual spectra data. Very recently, Liu *et al* [22] proposed a cross-modal auxiliary (CMA) framework that uses a generative model to map the acquired RGB presentation to another domain, such as NIR; and then using it along visual domain data towards multi-modal face PAD.

2.2 NIR Face-PAD Datasets in the Public Domain

Many of the publications discussed in Section 2.1 have been accompanied by publicly available face PAD datasets including presentations captured under NIR illumination. Here we present brief descriptions of some such datasets.

MS-Face [23]: This is the first public dataset to explore the use of NIR imagery for face PAD. Specifically, data is collected under two kinds of illumination: visible light and 800 nm (NIR) wavelengths. The dataset contains data captured from 21 subjects. *Bona-fide* presentations in this dataset have been collected under five different conditions. For PAs under visible light, high quality color prints have been used; whereas PAs under NIR illumination have been created using gray level images printed at 600 dpi.

EMSPAD [24]: the Extended Multispectral Presentation Attack Database (EMSPAD) contains images captured using a Pixelteq `SpectroCam`™ camera. The dataset contains seven band multispectral stacks per time instance, that is, for each frame, 7 images have been captured in narrow wavelength bands ranging from 425 *nm*–930 *nm*. *Bona-fide* and attack presentations for 50 subjects comprise this dataset. It includes only one kind of PAI, namely, 2D color-print attacks constructed using a color laser printer and a color inkjet printer.

MLFP [20]: The Multispectral Latex Mask based Video Face Presentation Attack (MLFP) dataset has been prepared for experiments in detecting obfuscation attacks using flexible latex masks. The dataset consists of 150 *bona-fide* and 1200 attack videos, corresponding to 10 subjects. The PAs have been performed using seven latex masks and three paper masks. Data has been collected in both indoor and outdoor environments.

Table 1
Details of publicly available Face PAD Datasets acquired in NIR imaging domain. (* indicates video presentations.)

Dataset	Year	PAI Type	# Files	# Subjects	Collection Environment
MS-Face [23]	2016	2D- print	2352	21	Indoor
EMSPAD [24]	2017	2D- print	10500	50	Indoor
MLFP [20]	2017	3D- latex mask	1350*	10	Indoor+Outdoor
CIGIT-PPM [25]	2019	2D- print; 3D- mask	93358	72	Indoor
WMCA [26]	2019	2D- print, replay; 3D- silicone mask, rigid mask	1941*	72	Indoor
XCSMAD [12]	2019	3D- custom silicone mask	535*	17	Indoor
CASIA-SURF [27]	2019	2D- print, cut	21000*	1000	Indoor
VFPAD (This work)	2021	2D- print, replay; 3D- silicone mask, rigid mask	5836*	40	Indoor+Outdoor

CIGIT-PPM [25]: The CIGIT PPM (paired photo and mask attacks) dataset consists of 93358 paired color and NIR face images corresponding to 72 subjects. Attack presentations consist of print and 3D mask attacks. The diversity of recordings includes variations such as spoofing medium, recording environment, pose, expression, glasses/no glasses, resolution and distance.

WMCA [26]: The Wide Multi-Channel presentation Attack (WMCA) dataset consists of short video recordings from multiple imaging channels that are spatially and temporally aligned. The presentations are recorded in color, depth, thermal, and NIR (860 nm) channels. It contains a variety of 2D and 3D PAs– 2D print and replay attacks, mannequins, paper masks, silicone masks, rigid masks, transparent masks, and non-medical eyeglasses. The dataset consists of 1679 videos, including 347 *bona-fide* videos collected from 72 subjects under various conditions.

XCSMAD [12]: The eXtended Custom Silicone Mask Attack Dataset (XCSMAD) is a multi-channel dataset dedicated to PAs constructed using custom-made 3D silicone masks. The data are captured using color, infrared, and two varieties of thermal imaging devices. It consists of 535 presentations in total, wherein 295 videos are PAs. It also provides *bona-fide* presentations of subjects whose 3D masks were created—which facilitates analysis of vulnerability and impersonation attacks.

CASIA-SURF [27]: This dataset consists of video recordings of 1000 subjects acquired in color, infrared, and depth channels. With 6 types of photo attacks, it consists of 21000 videos in total. The photo attacks also include operations such as cropping, bending the print paper, and stand-off distance.

Table 1 summarizes the details of publicly available PAD datasets acquired in NIR.

2.3 Face-PAD using CNNs

Over the last decade, deep CNNs have emerged as a promising tool for detection of face PAs. With superior performance over PAD methods based on handcrafted features, a majority of NIR-based methods discussed in Section 2.1 employ deep CNNs to perform face PAD. However, the amount of PAD data is often inadequate for training a CNN from scratch. Therefore, most CNN-based face PAD methods function in a two-step approach: a CNN is first trained in an end-to-end fashion for a task of face recognition from a large FR dataset (alternatively, several FR CNN models are publicly available for research purposes). This CNN is then fine-tuned using PAD dataset to distinguish between *bona-fide* presentations and PAs. The fine-tuning often involves utilizing the outputs of intermittent layers of the primary (FR) CNN, and constructing few subsequent layers (and a classifier) for improving the performance of the PAD CNN.

In [12], it was first demonstrated that embeddings extracted from a CNN trained for FR could be directly used to achieve very low misclassification rates for face PAD. This work relies on the 9-layer LightCNN [13]. A multi-channel CNN (MC-CNN) that takes advantage of shared layers of a CNN to obtain unified embeddings from all input channels (where each data channel represents data collected in a single spectral band) has been proposed in [26]. These embeddings are classified by a set of fully connected layers that perform two class classification. Li *et al.* [19] have adapted the MobileNet-v3 [28] for face PAD. They report a half-total error rate (HTER) of 1.1% on a self-collected dataset including RGB and NIR presentations. Unfortunately, this dataset is not publicly available. However, the work is of interest in our context, since they demonstrate that a fairly small CNN (namely, MobileNet-v3) can be used effectively for the task of face PAD. In [29], Kotwal *et al.* have introduced a *patch-pooling* layer in a CNN, which serves to extract texture descriptors of input. Using such patch-pooling layers in a LightCNN, they demonstrate perfect face PAD on the WMCA dataset and state-of-the-art performance on the MLFP dataset.

3 VFPAD DATASET

To develop an efficient face PAD system for an automobile, the foremost crucial requirement is appropriate training dataset. This dataset must encompass various real-world scenarios for the present use-case. In addition to variations in illumination, pose, and common accessories (that cause partial occlusion), the dataset should also include substantial presentations from different 2D and 3D PAIs. Towards this goal, we have created a novel dataset, named VFPAD (in-Vehicular Face Presentation Attack Detection) that consists of 5800+ video presentations from *bona-fide* and attacks acquired in NIR imaging domain.[1]

Each video presentation in the VFPAD dataset, approximately $10\,s$ long, has been recorded inside a car using Entron F001 (with NIR $940\,nm$ filter) camera positioned

[1] VFPAD dataset: https://www.idiap.ch/dataset/vfpad

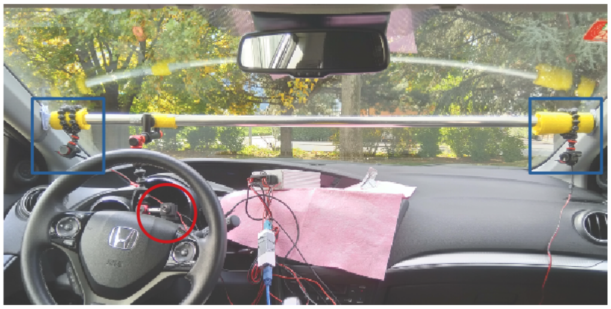


Figure 2. Setup inside the vehicle for VFPAD recordings. The NIR camera, mounted on the steering wheel, is marked by a red circle; and two blue rectangles show the locations of NIR illuminators.

on the steering wheel of the car. Figure 2 shows the data recording setup which also includes two NIR illuminators on each of the front pillars. The dataset consists of 4046 *bona-fide* recordings from 40 subjects (24 males and 16 females), and 1790 attack presentation videos from a total of 89 PAIs. The VFPAD dataset comprise HD recordings (1280 × 720 pixels) with 12-bit resolution.

Here, we provide details of PAIs and acquisition procedure for VFPAD dataset. The experimental protocol used throughout this work has also been subsequently described.

3.1 Description of PAIs

The 2D PAIs used in the VFPAD dataset include: *(a)* A4 sized color printed photographs, and *(b)* digital color images (photographs) and videos replayed on a tablet device (iPad 2). Each PAI (photograph, digital image, or video) represents a face region of the subject. We have used two kinds of color printers to prepare the printed PAIs: laser printer (Develop ineo+ 364e) and inkjet printer (Epson XP 680). As the toner used in the laser printer is NIR-reflective, the facial features in the PAI are clearly visible under NIR illumination. On the other hand, in the photographs printed on inkjet printer, the image (or facial features) are poorly visible or often invisible under NIR illumination. In the digital replay PAIs (iPad 2) as well, the image content is completely invisible under NIR illumination.

To create 3D PAs, we have considered the flexible as well as rigid face-masks for creation of the VFPAD dataset. The flexible masks used here are generic head-and-shoulders masks made of silicone. This type of commercial grade masks, with cost in the range of US$ 600–800, have previously been used for multichannel face PAD [26]. We have used 20 such masks to record PAs from the VFPAD dataset. An example of this category of masks is shown in Figure 3(a).

We have also constructed PAs in VFPAD dataset using 40 rigid masks for real subjects. These masks were manufactured by two sources (20 masks each)- Dig:Ed, a 3D-printing company in Germany; and REAL-f Co. Ltd. based in Japan. To create these custom masks, we captured the following data from 40 subjects: *(a)* a 3D scan of face captured using a RealSense SR300 (RGB-D) camera from Intel, *(b)* several facial photographs captured using a Sony Alpha 7-II camera, and *(c)* physical measurements of facial features (the subjects were asked to hold ruler next to their face during photography). The 3D-printed rigid masks manufactured by Dig:Ed are made of amorphous powder compacted with resin. Real-F uses a proprietary *Three-dimension Photo Form* technique to transfer high resolution photographs onto a synthetic curved surface. Additional facial features such as eyes, eyelashes, etc. are then applied to the mask. Figures 3(b) and (c) show rigid masks created by Dig:Ed and Real-f, respectively.

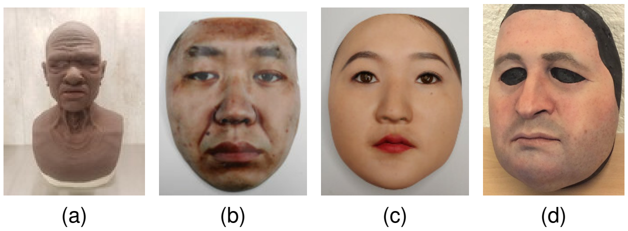


Figure 3. Examples of 3D PAIs used in the VFPAD dataset: (a) generic silicone mask, (b) custom rigid mask (by Dig:Ed), (c) custom rigid mask (by Real-F), and (d) custom silicone mask.

Additionally, we have also used 21 custom silicone masks made by Nimba Creations Ltd. (see [30] for details). An example of custom mask of this category is shown in Figure 3(d). During capturing presentations using silicone masks (which are manufactured with holes in place of eye-sockets), we have used either artificial eyes (made of plastic or glass) or simply eye-cutouts from printed photographs.

3.2 Data Acquisition

For *bona-fide* presentations, each subject sat in the driving seat. Similarly, for 2D PAs, the attacker sat in the driving seat, holding the 2D PAI at the appropriate position. For 3D PAs, the mask was attached to a mannequin or a stand; and the position, height, and angle were adjusted as required.

We introduce several variations in the recordings to construct a challenging dataset that mimics real-world scenarios. As the vehicle may be present in indoor or outdoor environment, we record the presentations in 4 different sessions: *(a)* outdoor in sunny weather, *(b)* outdoor in cloudy weather, *(c)* indoor in dimly lit area, and *(d)* indoor in brightly lit area. The outdoor sessions were recorded in open spaces, while the indoor sessions were conducted in basement parking.

Another form of illumination variation was introduced by controlling the NIR illuminators placed inside the vehicle. When both illuminators were on, the presentations received *uniform* illumination. We recorded *non-uniform* variant of presentations by keeping on only one NIR illuminator (the one near to the subject). For the NIR camera, placed on the steering wheel, to obtain a *frontal* presentation of subject's face, the subject (*bona-fide* or PA) is required to tilt the face at camera. On the other hand, when the subject is in normal position (*i.e.* looking ahead on the road), the NIR camera is placed *below* the subject's face, and is normal to their chin region. We have recorded both variations (frontal and below) of head pose in the VFPAD dataset.

In the real-world scenario, the person in driver's seat is likely to wear an accessory like glasses, sunglasses, or hat. These objects partially occlude the subject's face, and

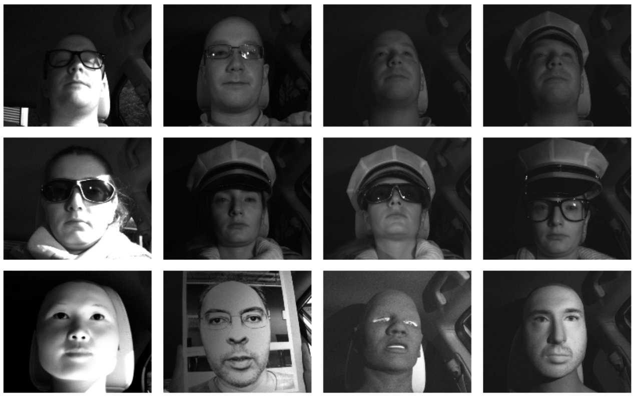


Figure 4. Samples of presentations from VFPAD dataset demonstrating variations in accessories, pose, and illumination, etc. The upper two rows consist of *bona-fide* samples, and the bottom row depicts presentation attacks. The images, captured in 12 bit resolution, have been converted to 8-bit representation by discarding lower 4-bits.

thereby, may deteriorate the performance of FR and PAD systems. Having a training data that consists of such occluded *bona-fide* presentations may be helpful in improving the robustness of face PAD methods. We have created 6–7 *looks* through different combinations of accessories for each real subject as- *(a)* natural, *(b)* medical glasses (wherever applicable), *(c)* artificial (dummy) glasses, *(d)* sunglasses (almost opaque to NIR illumination), *(e)* hat, *(f)* glasses + hat, and *(g)* sunglasses + hat.

Through combination of aforementioned variations, 96–112 videos presentations for each *bona-fide* subjects have been collected. For each PAI, 16 videos were recorded (no *looks* variation). Figure 4 shows examples of various presentations captured in the VFPAD dataset: where top two rows consist of *bona-fide* presentations, and the bottom row shows some attack presentations.

3.3 `grandtest` Protocol for VFPAD Dataset

To conduct the PAD experiments: training and testing the FPAD CNN, we have designed the *grandtest* protocol with fixed and disjoint partitions. In the grandtest protocol, as summarized in Table 2, the VFPAD dataset is split into three disjoint sets: one for training (`train`); one for development (`dev`) purposes such as tuning the parameters, or validating the performance of CNN; and one for evaluating the performance of the PAD system (`eval`). From each video presentation, 20 frames have been selected through uniform sampling of the video. However, since the number of print attacks were relatively lesser, we have selected 80 frames from presentations involving print attacks. This modification provides a good balance across different types of attacks in the dataset. The `train` set consists of 1503 *bona-fide* and 595 PA videos. The `dev` set contains 1247 *bona-fide* videos and 666 PA videos; while 1296 *bona-fide* and 529 PA videos comprise the `eval` set of the grandtest protocol. It is important to note that the three sets are subject-wise disjoint, that is, all data from a given subject appears only in one of the three sets.

Table 2
The `grandtest` protocol for the VFPAD dataset.

Partition	# Videos	Split Ratio (%)
train-*bona-fide*	1503	37.15
train-attack	595	33.24
dev-*bona-fide*	1247	30.82
dev-attack	666	37.20
eval-*bona-fide*	1296	32.03
eval-attack	529	29.56
Total	5836	

4 Proposed Face PAD CNN

In this section, we provide an overview of the proposed PAD method for detection of 2D and 3D presentation attacks on FR system. These presentations have been acquired in NIR imaging domain in a challenging in-vehicular setup.

An NIR-adapted Face PAD (referred to as **FPAD**) method using a deep CNN is proposed for the aforementioned task. In addition to high accuracy at detection of PAs, the proposed method is also computationally efficient with smaller memory footprint—which facilitates its deployment in resource constrained environment such as TEE. We employ a combination of transfer learning and domain-specific adaptation of few, specific layers of the pretrained FR CNN to accomplish the task of face PAD. Details of different stages of the proposed FPAD framework are described below.

4.1 Preprocessing

The first stage of the proposed FPAD framework is the preparation of input presentations to the specific format required by the deep CNN that functions as a feature extractor. With the availability of several face detectors, the detection of facial landmarks from the RGB presentations is quite straightforward. However, for the NIR presentations, the appearances of genuine human faces and PAs are quite different from the ones in visual spectra. A direct use of face detector pretrained on visual spectra data on the presentations acquired in NIR can result in poor accuracy of the detection and localization of face. The NIR data, therefore, should be explicitly *normalized* (or preprocessed) for better and accurate detection of face.

Additionally, the VFPAD dataset has been captured in various environmental conditions- indoors as well as outdoors. The VFPAD dataset, thus, exhibits a large variation in the illumination conditions. The top row in Figure 4 shows 4 samples from the VFPAD dataset. These samples, that belong to the same subject, have been recorded in different environmental conditions (from left to right: the outdoor sunny, outdoor cloudy, indoor dimly lit, and indoor brightly lit, respectively.) With such a high degree of uneven illumination, the face detection can fail very often. Such failures, considered as *failure to acquire* (FTA), reduce the amount of valid data available to the PAD system. The FTA also limits the applicability of the PAD system for real-life scenarios having similar illumination conditions.

To reduce the effect of uneven illumination, we normalize the input NIR presentation through a region-based adaptive histogram equalization (AHE). However, the vanilla

AHE methods tend to amplify the noise in homogeneous regions. We employ a contrast limited variant of AHE (also known as CLAHE) to each NIR presentation—where the input is divided into small regions (patches); each region is subjected to AHE with parameters determined by region-based statistic; and finally, the equalized regions are blended together. This preprocessing is specific to the detection of face and facial keypoints only. It is **not** applied during preparing the presentation for PAD. We apply MTCNN face detector [31] to detect the face and some facial keypoints. We align the input face presentation such that the eye center and mouth center of a face are aligned to predefined coordinates. The aligned face images are resized to a fixed dimension of 128×128 pixels. The Entron NIR camera used to acquire data generates a 12-bit output. We discard the lower 4-bits of the VFPAD data to obtain an 8-bit representation as required by the subsequent CNN.

4.2 Architecture of FPAD CNN

Since the PAD datasets are often too small to train a deep CNN from scratch, researchers have advocated the use of transfer learning from networks that have been pretrained for a similar, but not the same task. For this purpose, we consider a CNN pretrained for face recognition (FR) due to- *(a)* similarity with PAD with respect to the input feature space, and *(b)* availability of large amounts of training data (although acquired in visual channel). In the subsequent discussions, we refer to this network as 'base network' to distinguish it from the actual PAD network (or model).

Preparation of Base Network:
To build a base network, we have utilized a 9-layer version of LightCNN FR model [13]. This CNN (hereafter, LightCNN-9) is one of the most accurate publicly available FR CNNs with a demonstrated accuracy of $\approx 98.7\%$ on the LFW dataset [32]. With compact architecture, as depicted in Figure 5, the LightCNN-9 achieves this performance with a much smaller set of parameters compared to the other FR CNNs. (For a comparative study of other popular lightweight architectures, see Sec. 6.3.) The smaller size of the network is particularly important for the present FPAD framework, since the final model should be able to function from a low-end device, such as smartphone, with limited resources inside an automobile.

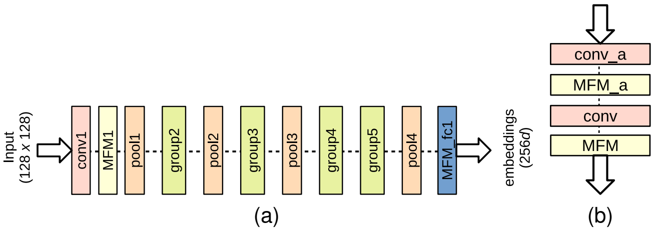


Figure 5. Schematic representation of (a) LightCNN-9 architecture, and (b) its group block.

The base network (FR CNN) is trained from scratch using a grayscale (or RGB) data. We adopt the training procedure as suggested by the creators of LightCNN-9 [13]. In addition to grayscale face presentations with specific alignment, Wu *et al.* suggest a use of data augmentation by random horizontal flipping and random cropping of input presentations to generate fixed size patches. The base network generates an output vector with dimensionality equal to the number of unique identities present in the training dataset. We use the cross entropy loss function to train the base network. For C class classification, the cross entropy loss, $\mathcal{L}_{\text{CE}}$, for i-th input presentation is given by Equation 1.

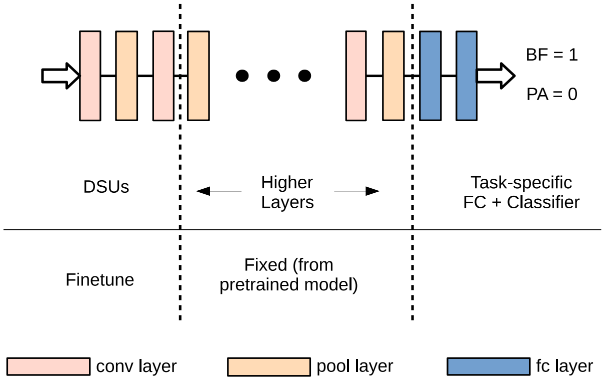


Figure 6. Proposed adaptation mechanism of a generic CNN for face PAD. Different layers of CNN are categorized according to their functioning during adaptation process.

$$\mathcal{L}_{\text{CE}} = -\sum_{c=1}^{C} y_{i,c} \, \log(p_{i,c}), \tag{1}$$

where y_i refers to a boolean (0 or 1) indicating whether the class label c for input i is correct or not. $p_{i,c}$ is predicted probability of input i belonging to class c.

Adaptation for Face PAD in NIR:
The base network is required to be adapted for a different task (face PAD) on data from a different domain (NIR). The filters learnt by first layer of a CNN are akin to those of Gabor filters or color blobs [33]; while the features computed by the last layer of a CNN are highly specific to the task and dataset [33], [34]. Therefore, several transfer learning methods are related to adaptation of higher layers of the base CNN as per the requirement and specifications of new task. To this end, we fine-tune the fully connected layer (`fc1`) of the base network from the VFPAD dataset to obtain the *embeddings* that are more appropriate for PAD. Also, we construct a regression-based classifier which is fed with the embeddings of FPAD CNN. This classifier can be represented as the final fully connected layer (referred to as `fc2`) having a single output node through sigmoidal activation. The dimensionality of classifier layer for FPAD CNN is 256×1.

Freitas Pereira *et al.* [35] showed that features learnt by higher layers of FR CNN are domain independent, and thus, these can be efficient at encoding face presentations acquired in different imaging domains. They demonstrated a use-case of heterogeneous face recognition (HFR) which involves FR and matching across pairs of various imaging domains. They showed that HFR can be performed by retraining only the lower layers of FR CNN using data from a specific domain, while the set of remaining features, from higher layers of CNN, can be shared among different imaging domains. The features from lower layers, to be adapted for a given domain, are also referred to as Domain Specific Units (DSU). In this work, however, the final objective (or task) remains unchanged across different domains. We apply this concept of domain-specific adaptation of FR CNN for the present task of face PAD. We consider the first two convolutional layers (`conv1` and `group2` as per Figure 5) as the DSUs, and retrain the same using NIR presentations from VFPAD dataset. Our work, thus, applies adaptation of DSUs in a cross-task scenario.

The overall procedure of adaptation and retraining of base network to obtain the FPAD CNN for NIR presentations is summarized in Figure 6. It indicates the role of different layers in the adaptation process for a generic CNN. To initialize the FPAD CNN, model weights from the base network are reused. This prevents plausible over-fitting, that may occur due to limited data from NIR imaging domain. The NIR presentations from VFPAD dataset are used to fine-tune the FPAD CNN. These presentations are preprocessed to obtain cropped, fixed size, and specifically aligned face images. We retrain the specific layers of FPAD CNN for a binary classification task using Binary Cross Entropy (BCE) loss function. If p is the predicted probability of a given presentation being *bona-fide*, then the binary cross entropy loss, $\mathcal{L}_{\text{BCE}}$, is given by Equation 2.

$$\mathcal{L}_{\text{BCE}} = -\left(y \, \log(p) + (1-y) \, \log(1-p)\right). \tag{2}$$

Here, y is a boolean for the label value which is set to 1 for *bona-fide*, and to 0 for PA.

5 Experimental Setup

In this section, we provide implementation details for the CNN framework proposed for face PAD inside automobile.

5.1 Training Base Network

Our base network, LightCNN-9, has been trained from scratch for FR using a visual spectra dataset. For training, we have used a publicly available CASIA WebFace dataset [36] that consists of 494,414 images from 10,575 unique identities. As the base network requires cropped and aligned facial region as the input, we have employed the MTCNN-based [31] face detector on images from the CASIA dataset. The face images, aligned to specific keypoints, were resized to 144×144 pixels—whose random crops of size 128×128 were provided to the base network during training. The images were also randomly flipped around their vertical axis, (*i.e.*, swapping of left and right sides of image) to improve the diversity of training data through augmentation.

We trained the base network for FR using Cross Entropy loss (also referred to as Categorical Cross Entropy) using Stochastic Gradient Descent (SGD) optimizer [37]. For SGD, we set the learning rate to 1×10^{-2}, weight decay to 1×10^{-4}, and momentum was set to 0.90. The base network was trained for 50 epochs with a minibatch of 128 samples.

5.2 Adaptation of FPAD Network

The LightCNN-9 network, pretrained on CASIA WebFace dataset for FR, serves as a base network for FPAD CNN. The first two convolutional layers (referred to as `conv1` and `group2` in [13] and the first fully connected layer (referred to as `fc1`) were adapted for the purpose of NIR-based face PAD. Thus, only these layers were set to *trainable*, while weights of all other layers were frozen (*i.e.*, not to be modified during retraining/fine-tuning process). The prefinal layer, `fc1`, of the FPAD CNN produces a 256-D embedding. We created a subsequent fully connected layer (`fc2`) with dimensionality of 256×1 to provide a final score (or the output) of the PAD network. The output of this layer was passed through sigmoidal activation.

The VFPAD dataset, specifically collected in in-vehicular environment, was used to train and evaluate the FPAD CNN. This dataset consists of 4046 *bona-fide* presentations, and 1790 attack presentations acquired in NIR channel. With `grandtest` protocol as described in Section 3.3, we considered 20 frames from each video for face PAD experiments, except for print attacks where 80 frames from each video were selected. For detection of facial keypoints, the NIR presentations from the VFPAD dataset were normalized using CLAHE method where we have used the regions with $1/8$-th of the input dimensions. The clip limit, as required by CLAHE to limit the over-amplification, was set to 5% for each region. The original presentations were converted to 8-bit format by discarding lower 4-bits, and provided to MTCNN face detector to obtain facial landmarks. After spatial alignment and resizing to 128×128, these presentations were provided to the FPAD CNN for domain-specific adaptation and fine-tuning. To adapt the FPAD CNN for binary classification problem of face PAD, we have used Adam optimizer [38] along with Binary Cross Entropy (BCE) loss. A learning rate of 1×10^{-4} was set for Adam optimizer; and the retraining procedure was run for 20 epochs where each minibatch consisted of randomly shuffled 128 presentations.

The `grandtest` protocol for VFPAD dataset consists of `train`, `dev`, and `eval` sets. To retrain (adaptation of DSUs and `fc` layer) the FPAD CNN, we have used the presentations from `train` set of the dataset only. At every epoch, the performance of FPAD CNN was validated on the `dev` set of the VFPAD dataset, and the best model was tracked.

5.3 Performance Measures

We have used the following evaluation measures for reporting the performance of the PAD system:

- **APCER** (attack presentation classification error rate) is defined as the proportion of presentation attacks (PA) incorrectly classified as *bona-fide*. If N_{PAIs} denotes the number of PAIS, the APCER of a PAD method is given by Equation 3.

$$\text{APCER}_{\text{PAIs}} = 1 - \frac{1}{N_{\text{PAIs}}} \sum_{i=1}^{N_{\text{PAIs}}} \text{score}_i, \tag{3}$$

where the binary variable score_i is set to 0 if the i-th presentation is classified as *bona-fide*, and to 1 otherwise.

When multiple categories (or species) of PAIs are present, the overall APCER is considered as the average APCER across total attack categories. However, in the present case, we have not differentiated within such categories.

- **BPCER** (*bona-fide* presentation classification error rate) is defined as proportion of *bona-fide* presentations that are incorrectly classified. For N_{BF} *bona-fide* presentations, the BPCER can be calculated as per Equation 4.

$$\mathrm{BPCER} = \frac{1}{N_{\mathrm{BF}}} \sum_{i=1}^{N_{\mathrm{BF}}} \mathrm{score}_i. \tag{4}$$

- **ACER** (Average classification error rate) is computed as the average of the above two measures:

$$\mathrm{ACER} = \frac{\mathrm{APCER} + \mathrm{BPCER}}{2}. \tag{5}$$

The EER, used to determine the score threshold on development (dev) set, is essentially the ACER for `dev` set where $\mathrm{APCER}_{\texttt{dev}} \approx \mathrm{BPCER}_{\texttt{dev}}$.

6 Results and Performance Evaluation

Since the VFPAD dataset is the first dataset of its kind (that captures presentations inside an automobile using NIR channel) we also evaluate the performance of some common face PAD methods on the VFPAD dataset to establish baselines. Subsequently, we present results with the proposed FPAD CNN. We also include the performance of FPAD CNNs adapted from two other commonly used light weight backbone CNNs. Finally, we briefly discuss the effect of adaptation of different layers for the proposed FPAD framework, and reasons for incorrect results.

6.1 Baseline Face PAD on VFPAD Dataset

To maintain the consistency across experiments from the baselines and proposed PAD method, we have used the same preprocessing steps for all experiments. Also, the presentations from `train` set of VFPAD are considered for training the classifier (or CNN, wherever applicable); and presentations from the `dev` set are used to select the score thresholds of classifier, and also the best model in case of CNN-based methods.

The first baseline method, proposed by Costa-Pazo *et al.* [39], uses Image Quality Measures (IQM) as features to be classified using a logistic regression (LR) classifier. We construct the first baseline using this PAD method on the VFPAD dataset, and refer to it as **IQM+LR** method. Our second baseline method consists of the features derived from local binary patterns (LBP) (uniform $LBP^{u2}_{8,1}$) to be classfied using LR classifier. We denote this baseline as **LBP+LR** method. Our third baseline, referred to as **CNN+LR**, makes use of the embeddings of a pretrained FR CNN (pretrained in visual spectra, but not adapted for NIR-based PAD) as features to train an LR classifier towards face PAD [12].

The performance of all baselines on the `grandtest` protocol of VFPAD dataset are provided in Table 3. The image quality-based PAD method, IQM+LR, resulted in classification error rates in the range of 8–12% across different partitions, and also across both classes of the VFPAD dataset. The LBP+LR method, that learns subtle micro-textural patterns, provided the average error rate above 10% on the `eval` set of the VFPAD dataset. Also, we observed that for the same score threshold, it did not produce good results on the `dev` set where nearly one in every eight samples were incorrectly classified. The CNN-based PAD baseline used the LightCNN-9 model pretrained on the CASIA WebFace dataset. Its embeddings were directly used to train the PAD classifier. With this baseline method, the APCER on the `eval` set of VFPAD was as low as 1.85%; however, more than 6% *bona-fide* samples were incorrectly classified as attacks. On VFPAD dataset, the CNN+LR method resulted in the average classification error of 4%. However, it may be noted that its classification accuracy on the `dev` set was nearly $2.5\times$ that on the `eval` set. For all baseline methods, the ACER values for `dev` set of *grandtest* protocol of the VFPAD dataset lie in the range of 10–15%, while these differ from their `eval` set counterparts with a margin of 4–5%.

6.2 Results of the FPAD CNN on VFPAD Dataset

The results of proposed FPAD method on the `grandtest` protocol of VFPAD dataset are provided in Table 4. On the `eval` set of the VFPAD dataset, the FPAD CNN achieved the overall accuracy of 98.53%. The score-threshold was chosen as EER on the `dev` set of the same dataset. With APCER of 1.49%, the proposed method misclassified 195 attack presentations (in terms of frames or images) from 13126 overall PA presentations. On the other side, out of 25565 *bona-fide* presentations (frames/images), 374 were misclassified resulting in the BPCER of 1.46%.

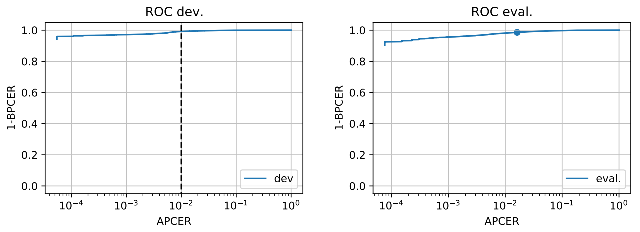
It can be observed that the proposed FPAD method outperforms the baseline methods by large margins in terms of classification error rates from both classes. For baselines using handcrafted features, the FPAD CNN provides an improvement of at least 7% in terms of overall accuracy measured by ACER. With a gain of 4.5+% of BPCER and marginal improvement in APCER, the proposed FPAD CNN proves to be superior to the deep CNN-based PAD baseline as well. On the `dev` set, the FPAD CNN provides at least

Table 3
Performance evaluation of the baseline PAD methods on the VFPAD dataset for grandtest protocol. All measure rates are in %. The numbers in parenthesis indicate the number of incorrectly classified samples for total samples in the given class.

PAD	`dev` set	`eval` set		
Method	ACER	APCER	BPCER	ACER
IQM + LR	11.69	8.53 (1119/13126)	9.09 (2324/25565)	8.81
LBP + LR	15.60	9.01 (1183/13126)	13.60 (3476/25565)	11.30
CNN + LR	10.97	1.85 (243/13126)	6.19 (1582/25565)	4.02

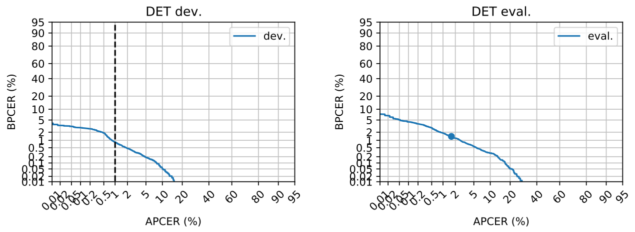
Table 4
Performance evaluation of the proposed FPAD method on the VFPAD dataset for `grandtest` protocol. All measure rates are in %. The numbers in parenthesis indicate the number of incorrectly classified samples for total samples in the given class.

Dataset Partition	FTA	APCER	BPCER	ACER
`dev`	0.27	0.92 (169/18453)	0.91 (221/24190)	0.91
`eval`	0.14	1.49 (195/13126)	1.46 (374/25565)	1.47



(a) `dev` set (b) `eval` set

Figure 7. Receiver Operating Characteristics (ROC) curve for the proposed FPAD CNN using `grandtest` protocol of the VFPAD dataset.



(a) `dev` set (b) `eval` set

Figure 8. Detection Error Tradeoff (DET) curve for the proposed FPAD CNN using `grandtest` protocol of the VFPAD dataset.

10× smaller error rates as compared to any of the baseline methods. Nearly similar error rates across disjoint partitions indicate a better generalization of FPAD CNN. Thus, transfer learning from a pretrained FR CNN, through adaptation of DSUs, is an efficient mechanism to obtain a CNN for the detection of face presentation attacks acquired in infrared domain.

Presentation attacks in the VFPAD dataset are constructed using 4 types of instruments (PAIs): photo prints, digital displays (to replay video), rigid masks, and flexible custom masks. In Table 5, we provide the breakdown of classification error rates for each category of PA. The video-replay attacks were carried out using an iPad. The contents of screen of this tablet were not visible in the NIR channel, and hence, these presentations, indeed, consist of noisy, spurious signals as captured by the sensor of the camera. For most replay attacks, the face detection fails, and therefore, the frame does not get processed. In case of PAs constructed using high quality photo-prints, photographs printed on the inkjet printer were not visible to the sensor of NIR camera. Therefore, only photo-prints obtained from the laser printer have been used to train and test the FPAD CNN. In the `grandtest` protocol, we have considered $4\times$ frames per video presentation for print attacks as compared to the presentations of *bona-fide* entities or those of mask attacks. A higher number of frames were used to balance different types of attacks, and thereby, to obtain the FPAD CNN with a robust response against a variety of presentation attacks. For the `eval` set of VFPAD dataset, 0.1% of print attacks were incorrectly detected as genuine ones. For presentation attacks created using a flexible facial mask, the misclassification rate was as low as 2.35% and 0.7% on the `eval` and `dev` sets, respectively, of the VFPAD dataset. The classification error rates were relatively higher for PAs using rigid 3D masks. We obtained APCER of 2.4–3.4% for rigid masks over different partitions of the VFPAD dataset using the `grandtest` protocol. For every category of PA, the corresponding APCER values are better than those obtained for any of the baseline methods.

Table 5
Performance breakdown across different PAIs for the FPAD CNN on the VFPAD dataset for `grandtest` protocol. The overall EER on the `dev` set was considered as the score threshold. All measure rates are in %.

PA Instrument	APCER (dev)	APCER (eval)
Replay	0.00	0.00
Print	0.00	0.10
Rigid mask	3.35	2.43
Flexible mask	0.69	2.35

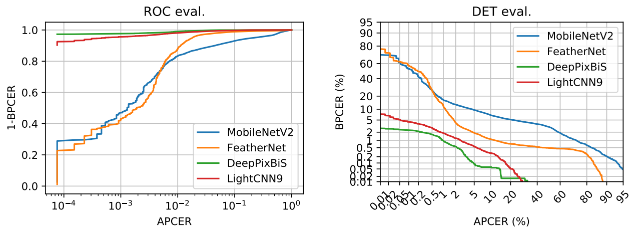
Figure 7 shows the receiver operating characteristics (ROC) curve for the FPAD CNN on `dev` and `eval` sets of the VFPAD dataset. for the proposed FPAD CNN, the ROC is nearly perfect for APCER values above 10^{-2} on both sets of the VFPAD. For extremely small values of APCER (around 10^{-3}), the BPCER deteriorates nominally by 2–3%. It should also be noted that the ROC is similar for both `dev` and `eval` sets, whereas the CNN was trained on the `train` set which has no subjects from the other two sets.

The detection error trade-off (DET) curve from Figure 8 indicates the behavior of the classifier over a range of APCER and BPCER values. As observed from the Figure, the value of BPCER for APCER @ 1.0% is nearly equal to 1.0% which indicates well-balanced performance of the FPAD CNN at low error rates. With this operating point, the performance of FPAD CNN on the `eval` set of VFPAD is marked (small circle) on the DET plot from Figure 8b. This operating point resulted in the APCER and BPCER values of $\approx 1.50\%$– both of which are nearly similar, and slightly higher than the corresponding error rates on `dev` set. It should be noted that the FPAD CNN was adapted only from `train` set; and data from other two sets were completely unused.

6.3 Performance of FPAD CNNs with Different Base Networks

A use of lightweight CNN as the backbone is important for the present FPAD use-case due to limited resource constraints. As a first step, we have conducted PAD experiments on the following 4 different architectures (with some ablation) to identify the most suitable architecture for VFPAD: MobileNets [40], [41], FeatherNet [42], DeepPixBis [43], and LightCNN-9 [13]. While MobileNets are generalized architectures to build lightweight networks; the FeatherNet, DeepPixBiS (based on DenseNet), and LightCNN have specifically been developed for face PAD.

To train the base (or backbone) networks for each architecture (except the DeepPixBiS), we have used the CASIA WebFace dataset, and followed the same procedure as described in Sec. 5.1 for training an FR network. For DeepPixBiS, we have followed the procedure devised by its original authors where the base network is trained on ImageNet. The model (or checkpoint) with the least loss on the training set was selected for subsequent adaptation



(a) ROC on `eval` set

(b) DET on `eval` set

Figure 9. Plots of ROC and DET of the FPAD CNN using different backbone architectures. The plots are obtained on the `grandtest` protocol of the VFPAD dataset.

for FPAD using VFPAD dataset. The domain-specific adaptation, as depicted in Fig. 6, requires finetuning a subset of DSUs and one or more fully connected layers at higher depth. With multiple blocks, layers, and their parameters, the MobileNets, FeatherNet, and DenseNet present a large number of combinations of layers that may be adapted or finetuned for the task of FPAD. A detailed study of the domain-specific adaptation for these deep networks is beyond the scope of this paper. Thus, we restrict our experiments to a specific subset of layers for each network while keeping all remaining layers fixed. The procedure for adaptation remains the same as detailed in Sec. 5.2. (Although for DeepPixBiS, the entire network has been finetuned as suggested by its authors.) Table 6 provides the details of layers finetuned for each network and the corresponding results obtained on the `dev` and `eval` sets of VFPAD.

Figure 9 shows the ROC and DET plots for `eval` set of the VFPAD dataset for each of the backbone architectures. We obtained less than 3% average error for the FPAD CNNs adapted from FeatherNet and LightCNN9, while the MobileNetV2-based network could not produce a comparable performance. However, it may be observed from the ROC and DET plots that the performance of FeatherNet drops significantly as one attempts to vary the operating point (*i.e.*, the score threshold). For the APCER of 1%, the FeatherNet-based FPAD CNN resulted in the BPCER above 10%. On the other hand, for the LightCNN9-based FPAD CNN, the BPCER value remained around 2% for the exactly same APCER setting. Following these observations, we decided to develop the final model based on the LightCNN9 as the backbone—although other two networks are much smaller in size. The DeepPixBiS architecture provided the best performance with average error of 1.24%; however, it finetunes all layers of the network using VFPAD dataset (while other architectures consider only a small fraction of the base network towards finetuning). Additionally, it uses a different dataset for training the base network. Therefore, these results may not be readily comparable, but they rather demonstrate a trade-off between the training complexity (or timing/ parameters) versus the possible performance of the resultant network. It may be noted that these results are based on our limited experimentation, and adapting more layers and/or changing training procedure may lead to different inferences.

Table 6
Details of layers/blocks of various base networks considered for the domain-specific adaptation; and the performance of corresponding FPAD CNN on the VFPAD dataset for `grandtest` protocol. The overall EER on the `dev` set was considered as the score threshold. All measure rates are in %. The names of layers are consistent with their original publications as cited.

Architecture	**Adapted Layers/ Blocks**	**ACER (dev)**	**ACER (eval)**
MobileNetV2 [41]	conv2d, bottleneck1/ exp1, bottleneck2/ exp (1 of 6), bottleneck7/ exp (6 of 6), conv2d, classifier	15.16	8.88
FeatherNet [42]	conv2d, BlockB, BlockB (1 of 6), BlockA (2 of 2), Streaming (DW), classifier	3.03	3.02
DeepPixBiS [43]	all layers	0.43	1.24
LightCNN9 [13]	conv1, group2, fc1, classifier	0.91	1.47

Table 7
Details of ablation study with different combinations of layers considered for the adaptation; and the performance of corresponding FPAD CNN on the VFPAD dataset for `grandtest` protocol.

Config	**Layers to Adapt**				**ACER**	**ACER**
	`conv1`	`group2`	`fc1`	`fc2`	**(dev)**	**(eval)**
1	×	×	✓	✓	0.62	10.14
2	✓	×	×	✓	3.36	3.32
3	✓	×	✓	✓	1.98	1.51
4	✓	✓	✓	✓	0.91	1.47

6.4 Performance of FPAD CNN with Different Adapted Layers

The proposed FPAD CNN is an outcome of domain-specific adaptation of a base network (trained for FR from visual spectra data). The adaptation procedure operates on a subset of domain-specific lower (or input-side) layers, and a subset of task-specific higher (or output-side) layers. To understand the effect of each subset, and also to find a suitable combination of layers on each side, we conducted the ablation study for adapting different combinations of layers of the FPAD CNN on the VFPAD dataset. For this task, we have considered only the LightCNN9-based FPAD CNN.

In [12], it was demonstrated that the FR CNN pretrained on the visual spectrum data can be effectively used towards face PAD of NIR presentations. Thus, in the first ablation, we do not adapt any DSUs of the base FR CNN, but consider prefinal fully connected layer and subsequent classifier for the purpose of adaptation. These layers are highly specific to the present task. In the complementary study, only the first convolutional layer (`conv1`) has been considered for adaptation, in addition to the final binary classifier. Here, a single DSU has been adapted to the input NIR data; however, the final embeddings (input to the final classifier) are obtained without any finetuning of task-specific higher layers. The next combination consists of adaptation of one layer from DSU (`conv1`) and one layer from task-specific layers (`fc1`) to perform face PAD on VFPAD dataset. It can be observed from Table 7 that this adaptation scheme results in a much better performance as compared to using only domain- or task–specific layers for the present task. In the final combination, we considered two layers from DSU and the task-specific fully connected layer for domain-specific adaptation of FPAD CNN. This combination resulted in the

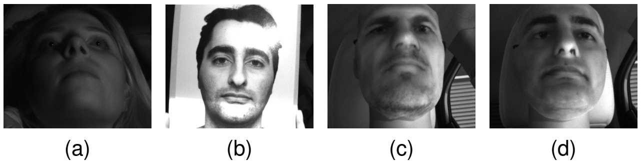


Figure 10. Examples of misclassifications by the FPAD CNN. First example on left is *bona-fide* presentation; and the other examples are presentation attacks. All examples are acquired in NIR spectrum. For 8-bit display, the lower 4-bits of the original data are discarded.

best set of results with APCER and BPCER values around 1%, with more layers (or parameters) being subjected to the adaptation.

Table 7 summarizes the combinations considered for adaptation and corresponding results obtained on the VF-PAD dataset. The learning rates were varied in the steps of $\{10^{-3}, 10^{-4}, 10^{-5}\}$, and training minibatches consisted of either 64 or 128 randomly sampled presentations.

6.5 Discussion

Misclassified Presentations: Figure 10 provides some examples of such presentations from VFPAD dataset. A fraction of misclassified samples appeared to be over- or under-saturated. For these samples, belonging to both classes- *bona-fide* and PA, a large region was not usable due to saturation which could have resulted in the incorrect classification by the CNN. The VFPAD dataset consists of 50% presentations captured from the camera placed at an oblique angle to the subject's face. In these presentations, the orientation of the subject's face or mask is highly non-frontal, which has been a reason for some misclassification. While the face detector and preprocessor are capable of working with minor variations in the size and orientations; for large deviations, the FPAD CNN may generate incorrect results. Finally, certain high-quality mask attacks appear highly similar to that of a genuine face when captured by the given NIR camera in in-vehicular setup. Their classification scores from FPAD CNN were quite close to those of *bona-fide* samples; and hence, resulting in misclassification.

Other Failures to Acquire Input: If the face detection fails, the corresponding presentation cannot be processed further. Such failures can mostly be attributed to *(a)* severely over- or under-exposed presentations, *(b)* too much angled (non-frontal) orientation of subject's face with reference to the axis of camera, and *(c)* non-visibility of a face (or any content) from some PAs, especially digital display ones, in the NIR wavelength bands of the capturing device. Figure 11 shows some examples of presentations where preprocessing module could not detect a human face. These samples are regarded as failure to acquire (**FTA**). Note that FTA values are same for the baseline experiments as well; and hence, the results are comparable.

7 Conclusions

In this paper we have developed a novel face PAD framework for *in-vehicular* environment. This framework, based on a compact 9-layer CNN, can be deployed inside a passenger vehicle for continuous verification of liveliness of the driver's face. It uses a full-HD, single channel NIR (940 nm in this case) image as the input to exploit advantages offered by extended range imagery. However, lack of sufficient training data acquired in these imaging channels (NIR) often limits the applicability of corresponding face PAD models. We provide a fine-tuning method coupled with adaptation of domain-specific layers of the base CNN– to obtain the face PAD CNN for NIR video presentations. The base CNN, in this case, has been trained on RGB data for the FR task. The aforementioned method modifies only 3 layers of the base CNN while the weights from other layers (as learnt for FR in RGB) are shared.

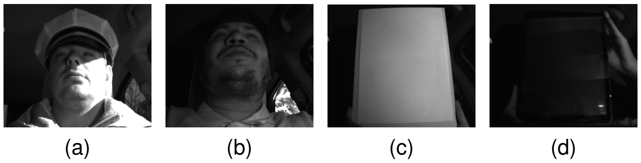


Figure 11. Examples of presentations from VFPAD resulting in FTA due to failure at detecting the face. First two examples are *bona-fide* presentations; while the last two examples are the presentation attacks constructed using a photo print, and a digital display, respectively. All examples are acquired in NIR spectrum. For 8-bit display, the lower 4-bits of the original data are discarded.

To evaluate the performance of the proposed face PAD method, we have collected a large dataset (VFPAD) consisting of 5800+ videos. It includes *bona-fide* presentations from 40 subjects and attack presentations from 89 PAIs (2D PAI: printed photographs and digital replay; and 3D PAI: rigid and flexible masks). These presentations have been captured in a car using a single channel NIR camera mounted on the steering wheel. The dataset provides four variations in external environmental conditions, and two variations of pose and NIR illumination each. Additionally, for *bona-fide* subjects, we have incorporated 6–7 variations through combinations of eyewear and hat. Experimental results over the `grandtest` protocol of VFPAD dataset show that the proposed method achieves excellent performance with approximately 1.5% of the APCER as well as BPCER. These results also outperform several classical and deep learning-based face PAD methods by a reasonable margin.

Acknowledgments

The authors would like to express their gratitude for the support from the algorithm team of Huawei IAS BU Smart Cockpit Product Department.

References

[1] D. Rotar and H. Popa Andreescu, “Face Recognition in Automotive Applications,” in *20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, 2018, pp. 355–359.

[2] J. Kang, D. Anderson, and M. Hayes, “Face recognition for vehicle personalization with near infrared frame differencing,” *IEEE Transactions on Consumer Electronics*, vol. 62, no. 3, pp. 316–324, 2016.

[3] N. Nagendhiran and A. Kolhe, “Security and Safety with Facial Recognition Feature for Next Generation Automobiles,” *International Journal of Recent Technology and Engineering*, vol. 7, pp. 289–294, 01 2018.

[4] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[5] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[6] Y. A. U. Rehman, L. M. Po, and M. Liu, “Livenet: Improving features generalization for face liveness detection using convolution neural networks,” *Expert Systems with Applications*, vol. 108, pp. 159–169, 2018.

[7] R. Shao, X. Lan, and P. C. Yuen, “Deep convolutional dynamic texture learning with adaptive channel-discriminability for 3d mask face anti-spoofing,” in *Proceedings of the IEEE International Joint Conference on Biometrics (IJCB)*, 2017, pp. 748–755.

[8] D. Perez-Cabo, D. Jimenez-Cabello, A. Costa-Pazo, and R. J. Lopez-Sastre, “Deep anomaly detection for generalized face anti-spoofing,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 06 2019.

[9] A. Liu *et al.*, “Multi-modal face anti-spoofing attack detection challenge at cvpr2019,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.

[10] A. Parkin and O. Grinchuk, “Recognizing multi-modal face spoofing with face recognition networks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 06 2019.

[11] S. Bhattacharjee and S. Marcel, “What you can’t see can help you – extended-range imaging for 3d-mask presentation attack detection,” in *Proceedings of the 16th International Conference on Biometrics Special Interest Group.* Gesellschaft fuer Informatik e.V. (GI), 2017.

[12] K. Kotwal, S. Bhattacharjee, and S. Marcel, “Multispectral Deep Embeddings as a Countermeasure to Custom Silicone Mask Presentation Attacks,” *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 1, no. 4, pp. 238–251, 10 2019.

[13] X. Wu, R. He, Z. Sun, and T. Tan, “A Light CNN for Deep Face Representation With Noisy Labels,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2884–2896, 11 2018.

[14] S. Marcel, M. Nixon, and S. Li, Eds., *Handbook of Biometric Anti-Spoofing - Trusted Biometrics under Spoofing Attacks*, ser. Advances in Computer Vision and Pattern Recognition. Springer, 2014.

[15] S. Marcel, M. Nixon, J. Fiérrez, and N. Evans, Eds., *Handbook of Biometric Anti-Spoofing - Presentation Attack Detection, Second Edition*, ser. Advances in Computer Vision and Pattern Recognition. Springer, 2019.

[16] R. Ramachandra, K. Raja, S. Venkatesh, and C. Büsch, “Extended Multispectral Face Presentation Attack Detection: An Approach Based on Fusing Information From Individual Spectral Bands,” in *Proceedings of 20th International Conference on Information Fusion (Fusion)*, 07 2017.

[17] J. Liu and A. Kumar, “Detecting presentation attacks from 3d face masks under multispectral imaging,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 06 2018.

[18] F. Jiang, P. Liu, X. Shao, and X. Zhou, “Face anti-spoofing with generated near-infrared images,” *Multimedia Tools and Applications*, vol. 79, pp. 21 299–21 323, 2020.

[19] L. Li, Z. Gao, L. Huang, H. Zhang, and M. Lin, “A Dual-Modal Face Anti-Spoofing Method via Light-Weight Networks,” in *2019 IEEE 13th International Conference on Anti-counterfeiting, Security, and Identification (ASID)*, 2019, pp. 70–74.

[20] Agarwal, A. and Yadav, D. and Kohli, N. and Singh, R. and Vatsa, M. and Noore, A., “Face presentation attack with latex masks in multispectral videos,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 07 2017.

[21] J. Hernandez-Ortega, J. Fierrez, A. Morales, and P. Tome, “Time analysis of pulse-based face anti-spoofing in visible and NIR,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 544–552.

[22] A. Liu, Z. Tan, J. Wan, Y. Liang, Z. Lei, G. Guo, and S. Z. Li, “Face anti-spoofing via adversarial cross-modality translation,” *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 2759–2772, 2021.

[23] I. Chingovska, N. Erdogmus, A. Anjos, and S. Marcel, “Face Recognition Systems Under Spoofing Attacks,” in *Face Recognition Across the Imaging Spectrum*, T. Bourlai, Ed. Springer, 2016, pp. 165–194.

[24] R. Ramachandra, K. Raja, S. Venkatesh, F. Cheikh, and C. Büsch, “On the Vulnerability of Extended Multispectral Face Recognition Systems Towards Presentation Attacks,” in *Proceedings of IEEE International Conference on Identity, Security and Behavior Analysis (ISBA)*, 02 2017, pp. 1–8.

[25] Jiang, F. and Liu, P. and Zhou, X, “Multilevel Fusing Paired Visible Light and Near-infrared Spectral Images for Face Anti-spoofing,” *Pattern Recognition Letters*, 2019.

[26] A. George, Z. Mostaani, D. Geissenbuhler, O. Nikisins, A. Anjos, and S. Marcel, “Biometric Face Presentation Attack Detection with Multi-Channel Convolutional Neural Network,” *IEEE Transactions on Information Forensics and Security*, vol. 15, no. 1, pp. 42–55, 2019.

[27] S. Zhang *et al.*, “A Dataset and Benchmark for Large-Scale Multi-Modal Face Anti-Spoofing,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 06 2019.

[28] A. Howard *et al.*, “Searching for mobilenetv3,” *CoRR*, vol. abs/1905.02244, 2019.

[29] K. Kotwal and S. Marcel, “CNN Patch Pooling for Detecting 3D Mask Presentation Attacks in NIR,” in *Proceedings of IEEE International Conference on Image Processing*, 10 2020, pp. 1336–1340.

[30] S. Bhattacharjee, A. Mohammadi, and S. Marcel, “Spoofing Deep Face Recognition With Custom Silicone Masks,” in *Proceedings of the IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, Los Angeles, USA, 10 2018.

[31] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, “Joint face detection and alignment using multitask cascaded convolutional networks,” *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 10 2016.

[32] G. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, “Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments,” University of Massachusetts, Amherst (MA), USA, Technical Report 07-49, 10 2007.

[33] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, “How transferable are features in deep neural networks?” in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 3320–3328.

[34] H. Li, H. Lu, Z. Lin, X. Shen, and B. Price, “Lcnn: Low-level feature embedded cnn for salient object detection,” *arXiv preprint arXiv:1508.03928*, 2015.

[35] T. de Freitas Pereira, A. Anjos, and S. Marcel, “Heterogeneous face recognition using domain specific units,” *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 7, pp. 1803–1816, 2019.

[36] D. Yi, Z. Lei, S. Liao, and S. Li, “Learning face representation from scratch,” *arXiv preprint arXiv:1411.7923*, 2014.

[37] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, “On the importance of initialization and momentum in deep learning,” in *Proceedings of the International Conference on International Conference on Machine Learning*, 2013, pp. III–1139–III–1147.

[38] D. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” 2017.

[39] A. Costa-Pazo, S. Bhattacharjee, E. Vazquez-Fernandez, and S. Marcel, “The Replay-Mobile Face Presentation-Attack Database,” in *Proceedings of International Conference of the Biometrics Special Interest Group (BIOSIG)*, 09 2016.

[40] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *arXiv preprint arXiv:1704.04861*, 2017.

[41] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[42] P. Zhang, F. Zou, Z. Wu, N. Dai, S. Mark, M. Fu, J. Zhao, and K. Li, “Feathernets: Convolutional neural networks as light as feather for face anti-spoofing,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 06 2019.

[43] A. George and S. Marcel, “Deep pixel-wise binary supervision for face presentation attack detection,” in *Proceedings of International Conference on Biometrics*, 2019, pp. 1–8.