# A Spatio-Temporal Residual Network for Deep Fake Anti-Spoofing

Seyed Ali Miraftabzadeh, Arun Das, Paul Rad

***Abstract*—With IoT proliferation, a decentralized biometric Access Control System (ACS) is crucial to ensure identity consistency in cyber physical space. In this paper, a biometric secure modality is proposed based on temporal domain that is often captured by IoT systems. A novel neural network architecture, consisting of ResNet blocks and Long Short-Term Memory (LSTM), is proposed to learn the temporal dynamics of human biometrics, such as face, in uncontrolled environment (typical of many real-world scenarios). The model is designed to support multi device-embedded implementation to keep reference data securely distributed on IoT devices in the form of biometric tokens for decentralized cooperative anti-spoofing, and 94.7% accuracy are achieved on YouTube Faces dataset. A time series analysis of these embeddings using Chebyshev distance is carried out to deter spoofing attacks such as video-replay by finding the dominant facial feature distribution from a ground truth reference. The distribution of data, which are not within the accepted error tolerance would be considered as fake data. The results presented on YouTube Faces dataset, as well as our in-house dataset, are promising to prevent tampering and spoofing unauthorized access for both real-time on-line and off-line identification in decentralized biometrics IoT.**

***Index Terms*—Deep learning, internet of things, face authentication, spoofing detection, temporal embeddings, time series analysis, Chebyshev distance, t-SNE.**

## I. Introduction

As more of the 'things' around us are being connected to the Internet - i.e. Internet of Everything, the need to authenticate ourselves on-the-go will be more pronounced. For example, a smart home can authenticate the occupants based on biometric traits, using an image or motion capturing Internet-connected devices in the home environment. In today's access control systems, users can be authenticated based on some possession, knowledge of some cryptographic materials, and biometrics. Traditional methods such as passwords and RFID cards does not complement a distributed IoT network of devices which requires authentication. Therefore, ensuring proper security using traditional methods will create a bottleneck. Multi-factor authentication has the same fate because it is tied back to at least one of the traditional methods of authentication. A well set-up remote biometric-based authentication system save users from having physical authentication cards or keys. Biometric authentication, due to its potentially higher level of security assurance and user experience is going to witness a rise in cyber-physical authentication.

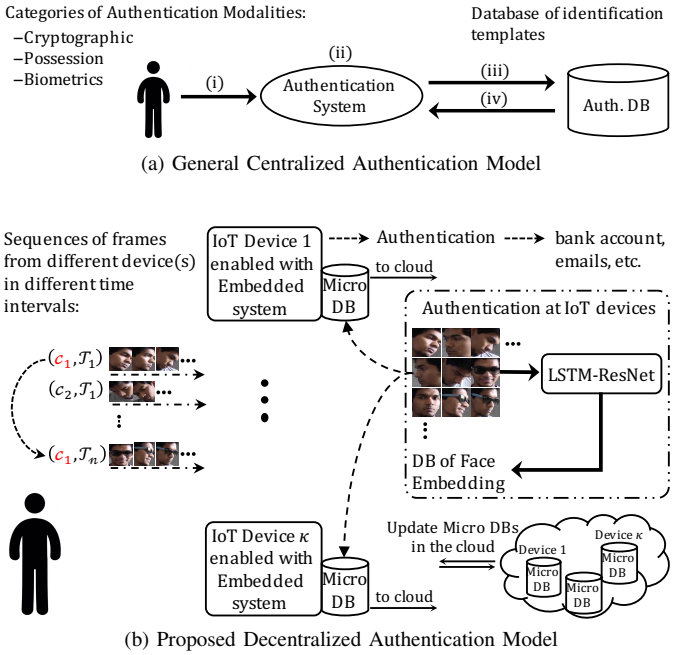



Fig. 1. High level illustration of the centralized authentication and proposed decentralized authentication model. In the proposed decentralized authentication model, sequences of frames are converted to unique temporal biometric token embeddings of the face using proposed ResNet-LSTM deep learning model and are saved in a database. A connected device can then query the database to verify, identify or authenticate a particular individual.

Biometrics can be broadly categorized into two classes, namely:

1) physiological traits: those relating to the shape and size of the body, such as facial features.
2) behavioral traits: those associated with human behavior and any changes over time, such as gait (the way one walks), and rhythm of typing keys.

Authentication schemes can also utilize a combination of both physiological and behavioral biometric traits. In this paper, we present a secure and dynamic biometric modality based on both physiological and behavioral changes of facial images over time. The proposed approach, illustrated in Figure 1, differs from common face recognition models designed for

The first two authors contributed equally to this work.

S. A. Miraftabzadeh and A. Das are with the Department of Electrical and Computer Engineering, University of Texas at San Antonio, San Antonio, Texas, USA 78249 (email: ali.miraftab@utsa.edu; arun.das@utsa.edu)

P. Rad is with the Department of Information Systems and Cyber Security and the Department of Electrical and Computer Engineering, University of Texas at San Antonio, San Antonio, Texas, USA 78249 (e-mail: paul.rad@utsa.edu)

identification and verification tasks in the following ways.

Face recognition can be undertaken in a number of various ways, such as by capturing a facial image using an optical camera (in the visible spectrum) or by processing the facial heat emission captured by an infrared camera [1]. In the visible spectrum, face recognition tasks rely on the modeling of facial features from a tight crop of the center of the face to avoid superficial features such as hair, beard, etc. which might change over time. Current face recognition algorithms are generally designed and trained to provide better identification and verification accuracies in large face datasets. However, the proposed model relies on dynamic physical and behavioral characteristics of an individual's facial images over time. It process the central portion of the individual's facial images in a sequence of frames coming from one or multiple cameras and it considers the face variations in different expressions in an uncontrolled environment.

From a system perspective, a centralized biometric access control is essentially a pattern recognition process, comprising the following steps, reference data set from authorized/unauthorized individuals are acquired, a notable feature set from the reference data set extracted, new extracted feature set is evaluated against the feature sets stored in the database, and a command to execute an action according to the result of the comparison is transmitted by a center [2]–[4]. This similar process is applied for other secure modalities as well, see Figure 1a. To maintain privacy and eliminate the inconvenience of storing sensitive biometric data, the proposed model is implemented as a distributed device-embedded biometric authentication system, such as an AI-embedded smart IoT camera (see Figure 1b). Device-embedded biometric authentication keeps data securely decentralized on local IoT devices for autonomous cooperative pattern recognition. In this paper, we assume IoT devices are dedicated hardware with sufficient computational resources, and not sensors or devices with low computational resources. This allows us to have the required process, feature extraction and comparison software built into the IoT device itself for autonomous pattern recognition.

In the visible spectrum, there are a number of approaches for the modeling of facial images. Examples include Local Feature Analysis, Elastic Graph Theory, Principal Component Analysis, Multi-Resolution Analysis, and Neural Networks. The authentication problem can be translated to a binary classification problem, when given a face image as the input it will respond with either a yes (i.e. successfully authenticated) or a no (i.e. denied). However, an adversary can spoof most of the existing models, including neural networks, by simple replay-based attacks [5] such as replaying a recorded video or photo of the user for authentication. As such, with the improved capabilities today's digital age provides, it is imperative that we find solutions to face anti-spoofing techniques that enables robust facial feature based ACS. Towards that goal, in this paper, a new neural network architecture is proposed to learn temporal features captured from a sequence of frames to classify the dynamic structure of faces. The network learns a global interpretation of a face's temporal evolution over time from one or more image capturing devices. This network consists of two sub-networks, namely: the recurrent neural network (RNN) with Long Short Term Memory (RNN-LSTM) and residual neural network (ResNet).

To learn hidden patterns in a time sequence, the proposed architecture leverages RNN-LSTM because RNN on its own has a drawback of vanishing and exploding gradients (also known as the vanishing gradient problem). In other words, RNN is inefficient when used to discover long-term temporal relationships from the input sequences. Thus, the RNN is implemented by leveraging enabled memory LSTM units to store the information for long-term temporal learning. To extract dynamic facial features, ResNet is used as a discriminative learning neural network which also addresses the vanishing gradient problem in the optimization phase. For each coming frame, one LSTM unit stacks above ResNet as a top layer before the final softmax layer. Therefore, the architecture leverages the local and dense property from the residual operation and learns the long-term temporal structure by supplying information in each LSTM units. The temporal embeddings enables the network to learn face embeddings for different facial profiles, orientations, expressions, etc. of users to be used later for the purpose of authentication. We show, with our time series study, that we can deter video replay attacks by evaluating the temporal nature of embeddings using a proper distance function.

Our contributions can be summarized as follows:

1) Proposed biometric secure modality based on an individual's dynamic facial features which change over time, to facilitate user authentication.
2) Proposed neural network architecture, ResNet-LSTM, which is designed to learn the temporal dynamics of human biometrics, such as face, for user authentication.
3) Proposed decentralized face anti-spoofing algorithm using Chebyshev method by measuring dominant facial feature distribution of streaming videos from a ground truth reference data points.

The rest of the paper is structured as follows: after describing some of the related literature in Section II, a detailed explanation of the proposed approach for face recognition using temporal embedding is described in Section III. Section IV lists out the neural network training specifications, hyperparameter values and other details to reproduce the research. Details regarding the experiments and their results are provided in Section V. The gist of the study and future directions are discussed in Section VI.

## II. Related Work

Face identification has attracted renewed attention due to advances in image and audio capturing devices on common consumer technologies such as CCTVs and mobile devices. Face identification also has many applications, such as user authentication.

A number of approaches for face identification have been proposed in the literature, as noted in the surveys by Learned and Miller [6], and Zhang et al. [7]. Most face identification techniques depend on neighborhood facial element recognition and order utilizing factual and geometric models of the human face. Low level investigation initially manages the division of

visual components utilizing picture properties, for example, edges, force, shading, movement, or summed up measures. A number of methodologies depend on format coordinating, where a few relationship layouts are utilized to identify nearby sub features, considered as inflexible in appearance (eigen features) or deformable. Visual elements in eigenfaces, a low dimensional representation of face images in a dataset, describe the general idea of a human face with the facial components and geometry intact.

Changes in the look of the face, perspective, occlusion, etc. have high impact in the representations generated by this method. Also, some key information regarding the individual is lost while creating a genaralized representation of a face. With a specific end goal to deal with troublesome situations where various appearances of changed sizes and postures must be recognized in hard scenes, some picture based example acknowledgment methods have been produced. They keep away from a generalized face model, such as eigenface, and displays exceptional tolerance to changes in the face, orientation, occlusions, etc. We are inspired by these ideas and model our face anti-spoofing architecture to be sensitive to facial orientation and expression changes as illustrated in Figure 2.

Research on privacy preserving facial recognition with a server-client model was explored in Erkin et al. [8]. The authors used an eigenface recognition algorithm on the server side to authenticate user requests from the client side by using an Euclidean distance function. As such, the authors found the distance between a new face vector from client side and the existing face vectors in the server side database and returned the details with minimum distance. Data exchange between the two sides were Encrypted using additive homomorphic encryption (AH-Enc) scheme. Since the client side is presented with details of the matching profile with minimum distance only, this method reduces data and information leakage.

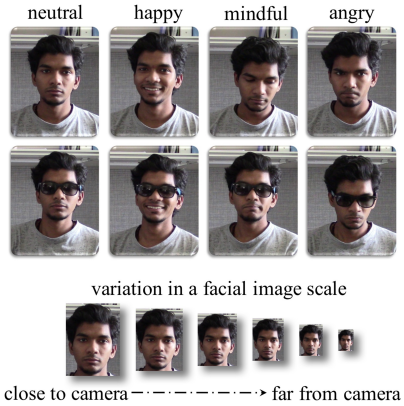
Eye image based biometric authentication protocols were explored in Blanton et al. [9] using iris image encoded using Hamming distance. Similar to [8], a client-server model is used with client receiving information of the comparison result with respect to encodings in a database hosted in the server. The protocol explored the use of garbled circuits and AH-Enc as part of their encryption schemes.

With successful results in prior work, Blanton et al. [10] explored single-server and multi-server outsourcing protocols for encryption of their iris biometric authentication. Deemed as the first published research towards this goal, authors explored non-interactive single-server utilizing a predicated-encryption scheme and more efficient multi-server computations, which is dependent on a minimum of three independent servers. Authors also introduced majority coding algorithms for iris codes and also numerous approximation techniques to reduce complexity and computation.

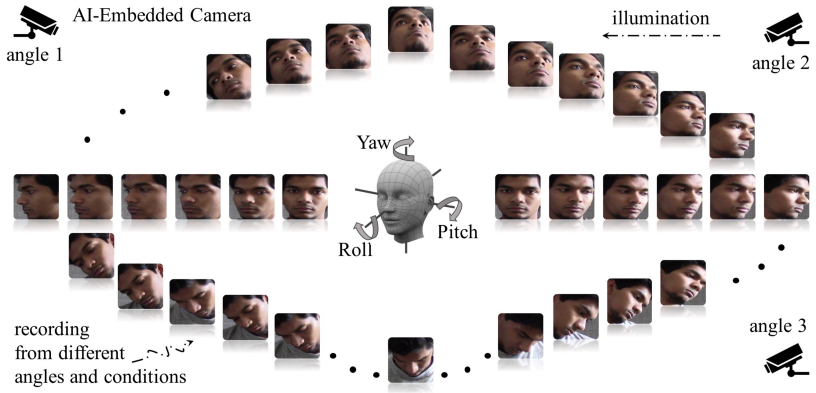
Our previous studies [11], [12] addresses the privacy concerns discussed above. As such, a secure biometric tokenization is explored in our proposed study addressed by generating unique embedding vector for different facial images of an identity. We highlight the ability of our model to uniquely identify person(s) of interest, from a distributed database (microDB), without knowing the true identity of the target. Much like the studies discussed, our proposed model does not disclose any information other than the client request. Details about our implementation is discussed in the following sections.

Thermal face detection is also a widely explored research domain [1] with lesser concerns of privacy. Finding anomalies within the predictions is important as well [13]. As we build applications around face recognition algorithms, for example, to support Alzheimer patients [14] or to help evaluate patient sentiments [15], the need for a secure face recognition algorithm sensitive to spoofing is essential.

Kumar Pandey et al. [16] described a framework for secure face template identification and protection using neural networks. They used a Convolution Neural Network (CNN) to create a mapping from an image to maximum entropy binary




(a) Sample of facial images of an individual in different expressions and scales




(b) Captures images of the same person in different views. The illumination direction is from right to left

Fig. 2. We model our face anti-spoofing deep learning architecture to be sensitive to pose and facial expression variations. As such, we illustrate the possible variations of an individual's facial images for the authentication task. Scale variations, occlusions, and expressions are illustrated in (a), and different Yaw, Pitch and Roll variations are illustrated in (b).

(MEB) code and hash them to a SHA-512 function for future verification. Similar work on template protection has been carried out by Kumar Jindal et al. [17] by training a CNN for one-shot and multi-shot enrollment schemes to learn robust mappings from face images to binary codes, finally hashed using a cryptographic hash function.

Spoofing face recognition algorithms and several anti-spoofing techniques have been studied extensively over the years. Pan et al. [18] introduced an eyeblink-based anti-spoofing face recognition algorithm which works on streaming video from a webcamera. They tapped into the eyeblink behavior in a temporal fashion to detect the liveliness of faces. Bao et al. [19] showed that optical flow based methods are effective against two-dimensional spoofing, for example using a photo-based replay attack. Määttä et al. [20] introduced a texture based spoofing detection method to deter photo replay attack by analyzing the micro-textures using multi-scale local binary patterns. Li et al. [21] introduced a 2D Fourier spectra based approach for liveliness detection from a single face image or face image sequences using the structure and movement information of live face.

Liu et al. [22] described a deep learning based convolution-recurrent neural network (CNN-RNN) model trained to estimate the face depth with pixel-wise supervision, and to estimate a Remote Photoplethysmography (rPPG) signal with sequence-wise supervision. A fusion of these depth and rPPG signals was used to distinguish between live vs. spoof faces.

Schwartz et al. [23] introduced an face anti-spoofing solution based on partial least squares and a set of low-level feature descriptors. The proposed model was capable of distinguishing live and spoof images and videos. Xu et al. [24] introduced a Long Short-Term Memory (LSTM) based deep learning model with a CNN feature extractor for face anti-spoofing. The LSTM-CNN model was trained on CASIA dataset to learn local and dense properties of inputs and temporal structure of input sequences to deter face spoofing.

Curbing face spoofing in mobile devices are also explored. Patel et al. [25] described an anti-spoofing solution for mobile phones using a multi-cue fusion algorithm which is less sensitive to face and facial landmark detection errors. Smith et al. [26] proposed an approach to deter face spoofing in mobile devices by determining the reflections of a sequence of on-screen images on the user's face.

Use of heterogenous inputs, such as visual versus near infrared (VIS-NIR), for face recognition is widely explored in recent research. Hernandez-Ortega et al. [27] introduced a rPPG model for presentation attack detection based on time series analysis on video data. Experiments carried on 3D mask attack database and heart rate database showed interesting results on RGB and NIR images.

Song et al. [28] introduced an adversarial discriminative model for heterogeneous face recognition by training a generative adversarial network (GAN) on an adversarial loss and a high-order variance discrepancy loss. A 256 dimensional embedding is generated which is relatively compact comparing with other published research on face recognition.

3D mask based presentation attacks is a much harder spoofing problem to deter. Lui et al. [29] explored numerous

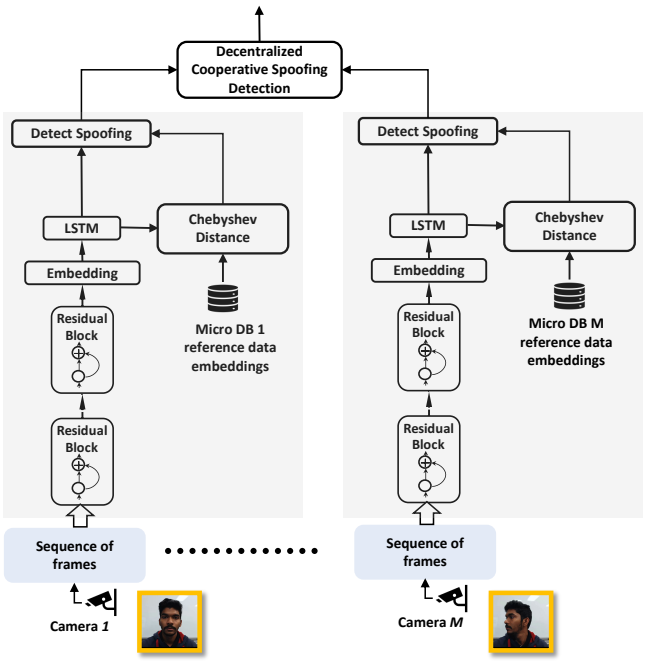



Fig. 3. Proposed ResNet-LSTM architecture for decentralized cooperative spoofing detection is illustrated. Each IoT device, which is self sufficient with computation and storage capabilities for face embeddings, performs face identification and prevents spoofing using proposed temporal anti-spoofing method.

CNN architectures for comparative performance evaluation of images acquired under NIR and visible illumination, towards automatic detection of 3D face masks used for face spoofing. Authors concluded that NIR based imaging of 3D face masks offered higher performance compared to visible light image counterparts. Authors collected images from 13 3D face masked subjects and 9 real subjects, and employed two protocols for performance evaluation.

Li et al. [30] proposed a novel framework leveraging deep learning using 3D CNN and domain generalization for face spoofing. The proposed 3D CNN takes both spacial and temporal information into consideration and improves generalization by minimizing the maximum mean discrepancy distance among the different domains. Various data augmentation methods were presented to generate novel training images to help generalization of the 3 million parameters of the proposed neural network.

## III. Proposed Anti Spoofing Approach

Our research goal is to address the problem of facial spoofing by investigating through on or off line streaming videos. In this section, we present the details of our decentralized anti-spoofing algorithm. To include all the facial image variations such as those presented in Figure 2, a sequence of video frames $(c_1, c_2, ..., c_n)$ is considered as an input and the output of the network is a binary number $y$, authorizing the user for the respective devices (see Figure 3). As illustrated in Figure 3, we propose a residual network combined with LSTM model to extract intra-class similarity and inter-class discriminatory of captured facial images from

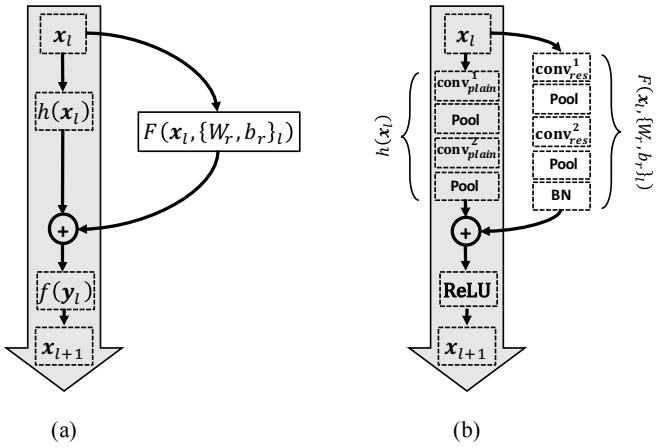



Fig. 4. Left: (a) General form description of the residual unit. Right: (b) illustration of the residual unit including: residual branch consists of two convolution layers each following by pooling and Batch Normalization at the end, and plain branch with two convolution layers follow by the pooling.

different video frames. Thus, the conditional probability of the output, $p(spoofing|(c_1, c_2, ..., c_n))$.

### A. Extract Temporal Features as an Embedding Vector

The temporal feature of a facial image in a frame is presented as an embedding vector. The embedding vector per identity is constructed through the residual network architecture consisting of residual blocks. The general form of each block can be formulated as:

$$\begin{aligned} y_l &= h(x_l) + F(x_l, (W_r, b_r)_l) \\ x_{(l+1)} &= f(y_l), \end{aligned} \tag{1}$$

where $x_l$ and $x_{l+1}$ are the input and output of the $l$th unit, $h$ is a forward function of the plain unit, $F$ is a residual function, $r$ is the number of repeated convolution layer in the residual function, and $f$ is a differentiable threshold function. Figure 4 (b) is an example of the detailed architecture of the residual unit. The initial idea of ResNet is to achieve additive residual function $F$ with respect to $h(x_l)$ and facilitates the minimizing of the loss function. Research presented in [31], [32] emphasize the importance of the identity mapping, $h(x_l) = x_l$, and thus, in the general formula we use $r$ to denote the number of repetition of the convolutional layers in the residual branch and we follow the identity mapping for the plain branch.

In the residual block, we need to consider also the differentiable threshold function. If $f$ is also considered identify mapping, for any deeper unit $L$ and shallower unit $l$:

$$x_L = x_l + \sum_{i=l}^{L-1} F(x_i, (W_r, b_r)_i) \tag{2}$$

This assumption turns the matrix-vector products, say:

$$x_L = \prod_{i=0}^{L-1} W_i x_0, b_i = 0 \tag{3}$$

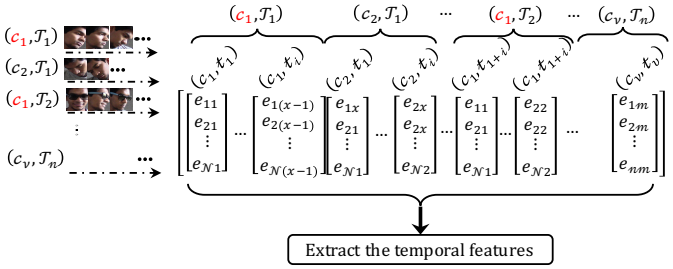



Fig. 5. Illustration of mapping facial images captured from different cameras to the embedding vectors

to the summation of the outputs of all preceding residual functions (plus $x_0$) [31], and consequently we arrive at the following backpropagation formula:

$$\begin{aligned} \frac{\partial E}{\partial x_l} &= \frac{\partial E}{\partial x_L}\frac{\partial x_L}{\partial x_l} \\ &= \frac{\partial E}{\partial x_L}\left[1 + \frac{\partial \sum_{i=l}^{L-1} F(x_i, (W_r, b_r)_i)}{\partial x_L}\right] \end{aligned} \tag{4}$$

An interesting property of Equation 4 in ResNet architecture is the capability to reduce the probability for the gradient to be canceled out. Refer back to the general form of the residual units, there are other residual units with the properties of increasing dimensions and reducing feature map sizes [32], [33] by using the conventional activation function, Rectified Linear Unit (ReLU), as the differentiable threshold function:

$$\frac{\partial E}{\partial x_l} = \frac{\partial E}{\partial x_L}\left[\frac{\partial x_L}{\partial h}\frac{\partial h}{\partial x_l} + \frac{\partial \sum_{i=l}^{L-1} F(x_i, (W_r, b_r)_i)}{\partial x_L}\right] \tag{5}$$

The last residual block maps a facial image into the embedding vector. Figure 5 illustrates such a mapping for different facial images captured from camera(s) in different angles and time windows.

### B. Exploring Temporal Relation in a Sequence of Embeddings

The output of the embedding vector is fed to the LSTM unit, which is the modified version described in [34]. LSTM units [35] have the ability to learn long range dependency from the input sequences. At the time step $t$, the behavior between input $(x_t)$, output $(h_t)$, and internal state is controlled through three gates. For each unit, $c_t$ stores the internal state and three gates are input gate $(i_t)$, output gate $(o_t)$, and forget gate $(f_t)$. Let $W$ and $b$ be model parameters, $\sigma$ be the sigmoid function and $g_t$ is the non-linear transformation of inputs, in Figure 6. To capture the temporal relation from the video frames sequence that is relevant for the identity authentication, outputs and cell memories from last time step are connected to the three gates through the defined dot products in:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \tag{6}$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \tag{7}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \tag{8}$$

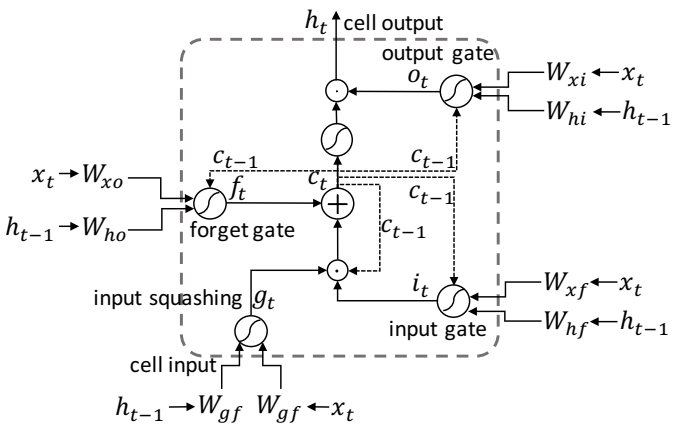



Fig. 6. A single LSTM unit is illustrated in the figure. Input, output, and forget gates are represented as circles with a curve, referring to a non-linear transformation. Element-wise product of respective inputs are carried out in the circle with a dot in the middle. Element-wise addition is done in the circle with a sqaure in the middle. Connection from the last time step is represented by the dashed arrows.

$$g_t = \text{PReLU}(W_{xg}x_t + W_{hg}h_{t-1} + b_g) \tag{9}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{10}$$

$$h_t = o_t \odot \text{PReLU}(c_t) \tag{11}$$

In Equations 9 and 11, PReLU represents Parametric Rectified Linear Unit activation function which is an extension of Leaky ReLU with coefficient of leakage as a learnable parameter instead of fixed hyperparameter. PReLU allows a small, non-zeo gradient to leak through when the unit is not active. Inputs of the three gates consist of the current time step of the input and last time step of the output and internal memory. This allows retaining information about face embeddings from previous frames, and current frame to improve the robustness of the model. The cell memory is updated as a result of the combination of input gate $(i_t)$ and forget gate $(f_t)$ (see Equation 10). The influence of the input in the internal state is controlled by the input gate and the forget gate takes control over the contribution of the last internal state to the current internal state. We call this new architecture ResNet-LSTM.

### C. Temporal Face Anti-spoofing Detection

We focus our attention to face authentication systems and enforce the following rules: 1) a user who wish to authenticate using the secure face recognition system has to do certain head orientation or expressions as requested 2) user will keep looking straight at the camera till user is authenticated. This allows the ResNet-LSTM architecture to generate temporal embeddings of the users and make statistical correlations with existing set of embeddings over time. Our focus for spoofing is centered around video replay-attacks, where an adversary captures a video of the user and shows it in front of the camera on a digital display.

We use Chebyshev distance to find the dominant feature changes in the embeddings over time. Chebyshev distance, also known as the Chessboard distance, is the $L_\infty$-norm of the difference of two vectors. For both ordinal and quantitative variables, it finds the absolute magnitude of differences between coordinates of a pair of objects. Consider two 1D face embedding vectors $X_f$ and $R_f$ representing a new and known reference embedding respectively. Then, the Chebyshev distance between $X_f$ and $R_f$ is given by:

$$\begin{aligned} &D_{chebyshev}(X_f, R_f) : (X_f, R_f) \mapsto \|X_f, R_f\|_\infty \\ &= \lim_{p\to\infty} \left( \sum_{i=1}^{n} |X_{fi} - R_{fi}|^p \right)^{\frac{1}{p}} = \max_i |X_{fi} - R_{fi}| \end{aligned} \tag{12}$$

A spoofing detection algorithm is modeled as a decentralized decision making process across multiple cameras. The high-level flow of the process is described in Algorithm-1. Further experimental validation of the algorithm and discussions on Chebyshev distance as an anti-spoofing method using proposed ResNet-LSTM model is described in subsection V-B.

**Algorithm 1** Face Spoofing Detection Algorithm based upon Chebyshev's Distance Function by Measuring Similarity of Face Feature Embedding Distribution and Ground Truth Reference

**Input:** $S = \{s_1^1, s_1^2, \ldots, s_2^1, s_2^2, \ldots, s_n^m\}$ where $s_n^m$ is $n^{th}$ frame from a stream of images from $m^{th}$ camera.
$W_{size}$ = Window size or number of frames to process from the input stream to limit $S$ to $S_w$.
$R_f$ = Reference point embedding features.
$\epsilon$ = Error tolerance.
**Output:** Face-spoofing detection result

```
Limit S → S_w based on W_size.
for j ∈ 1 → m do                                   ▷ For each camera
    for i ∈ 1 → n do                               ▷ For each frame
        X_i^j ← ResNet-LSTM(s_i^j)
        D_i^j ← ChebyshevDistance(X_i^j, R_f)
    end for
end for
for all X do
    D_min = argmin(D)
    if D_min ≤ ε then
        spoof = 0
    else
        spoof = 1
    end if
end for
```

## IV. Architecture and Training Methodology

Prior works on residual networks done by He et al. [31], [32] shows that a combination of residual architecture and embedding address the over-fitting and degradation problems. It has been shown that augmented LSTM units with PReLU can learn the fine distinction between sequences of nonlinear periodic embeddings patterns without external resets.

The implementation for residual network follows the practice in [31], [32]. To address the different scales of the facial images, a pyramid scale of each sample is crated the face images with a resolution in $\{250\times250, 120\times128, 60\times64, 30\times32\}$ are used and the maximum mini-batch sizes are {32, 64, 128, 128, 324} respectively.

The ResNet-LSTM model is implemented and optimized by using TensorFlow, the open-source software library for

machine intelligence [36]. The residual network and LSTM units are trained separately on a deep learning cluster described in [37] using advanced NVIDIA Tesla P100 GPUs with 5.3 TeraFLOPS built for data center. The test is carried on embedded system, NVIDIA Jetson TX2.

Different residual network architectures are trained from scratch while sharing almost the same optimization strategy and parameters. In the training phase, precise investigation of the effects of different stochastic gradient descent (SGD) algorithms are conduced: Adaptive Subgradient [38], RMSProp [39], Adam [40], and Momentum [41]. It is observed that Adam optimization algorithm outperforms the other methods because of the bias-correction characteristic of the algorithm. The best performance were achieved with $\beta_1 = 0.96$, $\beta_2 = 0.\overline{9}$, and $\epsilon = 91$. For several epoch iterations the learning rate is set to $0.001$ and it is reduced by a factor of 10 to stabilize the optimization. $L2-$norm with a weight decay of 0.1 is applied as an artificial constraint to implicitly reduce the number of free parameters and not to make the network difficult to optimize. Since the network is trained for the specific group of authorized users, it is advised to avoid exploiting dropout as the regularization method.

LSTM units are trained following the practice in [34] using stochastic gradient decent (SGD) and a momentum of 0.9 with the truncated Backpropagation Through Time (BPTT). To enable the LSTM to learn bridging minimal time lags, the gradient is truncated where the performance degradation is observed.

TABLE I
DIFFERENT ARCHITECTURE FOR APPLIED RESIDUAL NETWORK FOR RESNET-LSTM. BUILDING BLOCKS ARE SHOWN IN BRACKETS, WITH THE NUMBERS OF BLOCKS STACKED.

| Layer name | ResNet-A | ResNet-B | ResNet-C |
|---|---|---|---|
| conv_0 | $7 \times 7, 64$ stride 2 | | |
| conv_1 | $7 \times 7, 128$ stride 1 | | |
| block_0 | $\begin{bmatrix} 1\times1,32 \\ 3\times3,32 \\ 1\times1,32 \end{bmatrix} \times 3$ <br> $\begin{bmatrix} 1\times1,32 \end{bmatrix} \times 1$ | $\begin{bmatrix} 1\times1,32 \\ 3\times3,32 \\ 1\times1,32 \end{bmatrix} \times 3$ <br> $\begin{bmatrix} 1\times1,32 \end{bmatrix} \times 1$ | $\begin{bmatrix} 1\times1,32 \\ 3\times3,32 \\ 1\times1,32 \end{bmatrix} \times 3$ <br> $\begin{bmatrix} 1\times1,32 \end{bmatrix} \times 1$ |
| block_1 | $\begin{bmatrix} 1\times1,32 \\ 3\times3,32 \\ 1\times1,32 \end{bmatrix} \times 3$ <br> $\begin{bmatrix} 1\times1,32 \end{bmatrix} \times 1$ | $\begin{bmatrix} 1\times1,32 \\ 3\times3,32 \\ 1\times1,32 \end{bmatrix} \times 3$ <br> $\begin{bmatrix} 1\times1,32 \end{bmatrix} \times 1$ | $\begin{bmatrix} 1\times1,32 \\ 3\times3,32 \\ 1\times1,32 \end{bmatrix} \times 3$ <br> $\begin{bmatrix} 1\times1,32 \end{bmatrix} \times 1$ |
| block_2 | $\begin{bmatrix} 1\times1,32 \\ 3\times3,32 \\ 1\times1,32 \end{bmatrix} \times 3$ <br> $\begin{bmatrix} 1\times1,32 \end{bmatrix} \times 1$ | $\begin{bmatrix} 1\times1,32 \\ 3\times3,32 \\ 1\times1,32 \end{bmatrix} \times 3$ <br> $\begin{bmatrix} 1\times1,32 \end{bmatrix} \times 1$ | – |
| block_3 | $\begin{bmatrix} 1\times1,32 \\ 3\times3,32 \\ 1\times1,32 \end{bmatrix} \times 3$ | – | – |
| Embedding | Average pool, $128-d$ fc, softmax | | |

## V. EXPERIMENT

The proposed residual network is trained using FaceScrub dataset [42] which is one of the most accurate face datasets in terms of duplicated, mislabeled, and morphed faces. It comprises a total of 106,863 facial images of 530 celebrities (male and female), with about 200 images per person. As such, it is one of the largest public face databases, with an average of 2.15 images per person. Images are collected from the Internet and are taken in uncontrolled conditions (real-world situations). To maximize the usage of the dataset, it is split into seven folds; five folds for training, one fold for validation, and two folds for testing. Note that there is no common subjects in the folds. To prevent the unbalance problem, an even ratio of image per person in each fold is maintained.

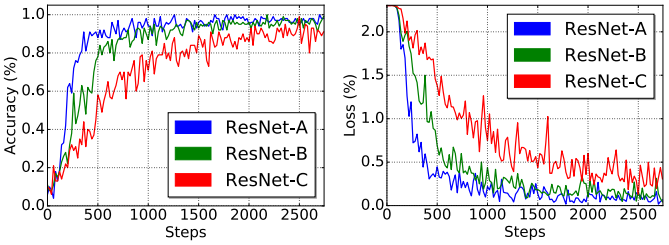
The LSTM units of the ResNet-LSTM model are trained using YTF dataset [43], with no people overlapping with FaceScrub. YTF dataset consists of large variations in pose, expression and illuminations for 1,595 recorded different identities in 3,425 videos (in average 2.15 videos per person). The dataset is made up of diverse video duration, 48 frames to 6,070 (in average 181.3 frames per video).

The facial data in the images and the videos varies in size and length, and contains a lot of background noises. Face detection and alignment are the primary steps to make them ready for training the ResNet-LSTM model. To detect and localize faces, multitask convolutional neural network proposed in [44] is applied on both datasets. Cases in which the algorithm fail to detect face locations are eliminated from the experiment process. For data augmentation in the training phase, after cropping face locations, images are randomly horizontally flipped, rotated $\{3^\circ, 5^\circ, 7^\circ, \text{or } 10^\circ\}$, and brightness and contrast of the images are manipulated.

### A. Model Details and Results

Three types of residual network architecture are explored, see Table I. Their practical differences lie in the variation in the number of residual blocks and consequently FLOPS. Depending on the application, the best model may be different. A model running at the embedded system as the edge device can have many parameters and handle more computation per second, where as the model running on a mobile phone has restrictions in a term of memory capacity and device battery life. Feature extraction is performed by adding $1 \times 1 \times d$ convolutional layers rather than playing with different filter sizes as suggested in [45]. Dimension matching in connections is performed by the sole convolutional layer at the end of each block, except the last one connected to the embedding.

On the FaceScrub dataset, the residual networks are trained to learn a feature mapping from facial images to a feature vectors in a compact euclidean space in which $L2-$norm distances present the similarity of faces [46]. To address hard samples problem [46], the residual networks supervised by center loss are trained and optimized by standard stochastic gradient descent instead of the original triplet loss function for such an embedding system. All residual architectures are trained with previously described folds, the average loss of a validation fold in multiple experiments is used to tune the models parameters. The best parameter is exploited to retrain models on the whole training folds and then the final result is reported on the fold test.




(a) Training accuracy (b) Training loss

Fig. 7. Classification results for training different ResNet architectures on FaceScrub. We see an increase in the learning performance for deeper networks.

TABLE II
ERROR RATES (%) OF SINGLE-MODEL RESULTS ON THE FACESCRUB VALIDATION FOLD AND TEST FOLD.

| Model | Validation Fold | | Test Fold | |
|---|---|---|---|---|
| | top-1 error | top-5 error | top-1 error | top-5 error |
| ResNet-C | 17.81 | 5.22 | 18.03 | 3.01 |
| ResNet-B | 15.37 | 3.12 | 16.36 | 1.88 |
| ResNet-A | 15.13 | 1.97 | 16.12 | 1.23 |

TABLE III
PERFORMANCE OF DIFFERENT METHODS ON YTF DATASETS.

| Method | Protocol | Acc. on YTF |
|---|---|---|
| DeepFace [47] | Large | 91.4% |
| FaceNet [46] | Large | 95.1% |
| Deep FR [48] | Large | 97.3% |
| ResNet-A | Small | 84.4% |
| ResNet-LSTM-A | Small | 94.7% |

Figure 7a shows the classification accuracy obtained from the training on different permutations of the five training folds. The deeper residual architecture, ResNet-A, was training much faster, see Figure 7b. The proposed network is explored and examined with multiple probes to monitor the bias-variance trade-off of the weight vectors. The experiment illustrates the deeper network has better training accuracy. To increase the algorithm performance, the hyper-parameters are fine-tuned to achieve a balance between bias and variance. In turn, the algorithm utilize all variables to learn fair assumptions about the form of the target function while suggesting small changes to the estimate of the target function. This makes the ResNet-LSTM architecture highly sensitive to facial orientation and expression changes.

Table II presents some comparisons between various versions of residual networks for top-1 error and top-5 error on validation fold and test fold. These major observations affirm ResNet-A reduces the top-1 error by $2.68\%$, which is a result from the successfully reduced learning error in the training phase, see Figure 7b. This comparison verifies the effectiveness of residual learning on extremely deep systems. Figure 8 illustrates a classification result of the $128-d$ embedding for ResNet-A architecture after applying $2D$ principal component analysis.

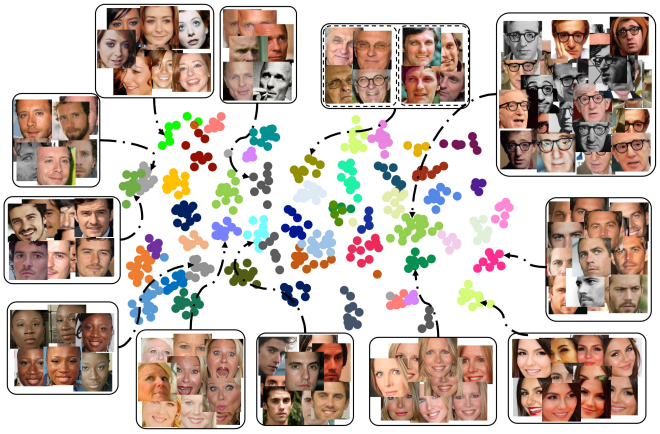


Fig. 8. 128-dimensional embedding for ResNet-A architecture after applying 2D principal component analysis (PCA). The different face clusters shows the generalization of ResNet-A architecture and the learning performance to represent different users in a 128-dimension vector.

The basic result of the ResNet-LSTM architecture on YTF dataset is summarized in table III. The result indicates that ResNet-LSTM architecture can learn a discriminant representation of the input facial images over a sequence of frames. Two protocols are introduced in table III, large protocol which indicates that the related method is trained using large datasets in sophisticated efforts, and small protocol indicating that small training dataset is used for the related method. Note that the proposed approach focuses on small protocol rather than the large one. The methods which are trained in large datasets show decent results in face recognition task on YTF. Significantly, the proposed ResNet-LSTM method exhibits analogous results. Note that face recognition methods try to recognize an individual on an image correctly, but ResNet-LSTM tries to recognize an individual on a sequence of frames recorded the person. The best obtained accuracy of ResNet-A architecture by itself is $84.4\%$, and $94.7\%$ accuracy for ResNet-LSTM-A architecture, which is a notable improvement compared to the ResNet-A. By learning temporal features from input sequences, the ResNet-LSTM-A model performs superior than general residual neural network and analogously to sophisticated architectures trained on large datasets.

### *B. Temporal Biometric Token Embeddings for Face Anti-Spoofing*

Face spoofing is one of the most prominent cyber attacks of modern digital age. As face recognition systems are being used to verify and authenticate users in border crossings, banks, personal device unlocking, mobile payments, and much more, the need for secure and spoofing-proof face recognition methods is important than ever. As described briefly in Section II, face anti-spoofing methods are being developed to mitigate real-world spoofing commonly done using masks, high-definition photographs, replaying videos, and such. In our study, we make use of the proposed ResNet-LSTM architecture that can learn a discriminant representation of the input facial images over a sequence of frames to promote anti-spoofing.

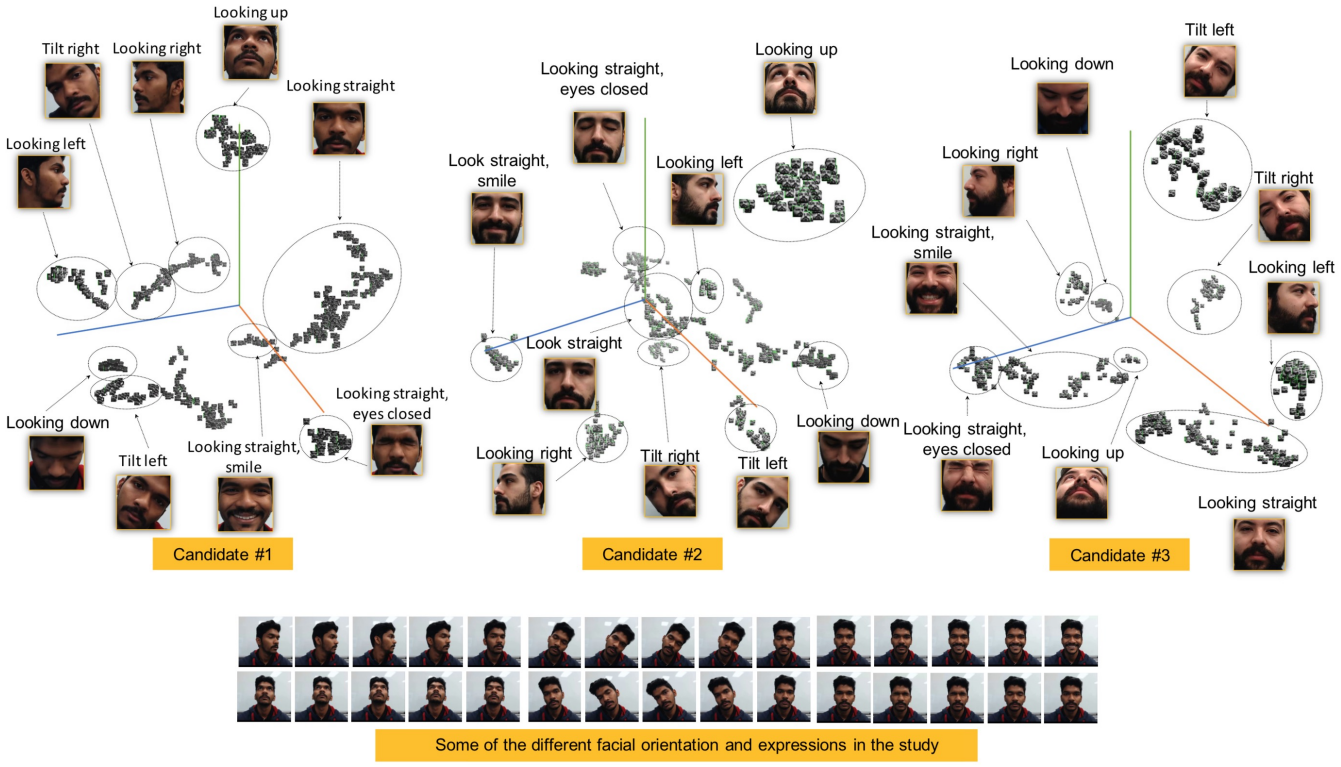



Fig. 9. Comparison of 3D t-SNE representations of the embeddings. Figure illustrates spatially segregated clusters of different candidates for different facial orientations and expression. A few sample facial orientation changes are also illustrated. t-SNE study shows the high sensitivity of ResNet-LSTM towards facial orientation and expression changes.

In [49], Wattenberg et al. described the effective use of the t-SNE plots. t-SNE is a non-linear algorithm which creates a lower-dimensional representation of a high-dimensional input space. It performs different non-linear transformations on different regions of the input distribution based on a tunable parameters named perplexity and a learning rate, epsilon. Perplexity is tuned to balance the attention between local and global aspects of input data distribution, which in-turn helps to find local and global neighbors of each input point. We chose a value of 30 for perplexity and epsilon of 10 in our t-SNE study to cluster face embeddings extracted from the ResNet-LSTM model. Each data point thus correspond to the different face orientations and expressions a user is asked to carry out. For example, in our studies, for the purpose of authentication, a user is asked to look straight at the camera, then followed by looking left, then right, up and then down. Afterwards, user closes the eye, smiles, tilt head left and right. These events are recorded and face embeddings are generated using the ResNet-LSTM model. Figure 9 illustrates the 3D t-SNE visualizations for three candidates. We see creation of clusters for each facial expression and orientation considered for our study, suggesting the high performance of proposed ResNet-LSTM deep learning model.

A time series analysis of the embedding, as illustrated in Figure 10, reveals patterns in the embeddings for facial movements and expressions of the user when compared to a known embedding. By using the Chebyshev distance, as discussed in subsection III-C, we find events in time where there is a sharp difference in the dominant facial features [50]. The ground truth curve in Figure 10 illustrates the temporal feature changes of the original video for different facial orientation (pose) variations. Here, the positive and negative slopes illustrate the changes in facial orientations, which in our case is $candidate - 1$'s head movement from $straight \rightarrow left \rightarrow straight \rightarrow right \rightarrow straight \rightarrow up \rightarrow straight \rightarrow down$ in this specific order.

Replay-based attacks are one of the most widely exploited means of face-spoofing partly due to its easiness. Reproducing a photo, video, high-definition masks, and more, of a user in front of an authentication system with the intent of spoofing the system comes under the umbrella of replay attacks. Out of the many, photo based attacks are tackled using histogram, quantization, local binary pattern and other methods. Since photo's are essentially static, our ResNet-LSTM model quickly identifies the noisy changes in the distance of embeddings between the frames. Hence, we present the tests carried out against video spoofing, which is a much harder spoofing method to deter.

To simulate replay attacks, the original video is replayed in a 2K display screen and the same camera used to record the ground truth videos is used as the input source to the ResNet-LSTM algorithm. The face feature embeddings are extracted and distance between the embeddings and a known true embedding is found out and plotted, as illustrated in the replay attack curve in Figure 10. Here, Figures 10 $(a)$ and $(d)$ are in a well lit room. Figures 10 $(b)$ and $(c)$ are in darker rooms, with $(b)$ slightly lit from above and $(c)$ lit from below

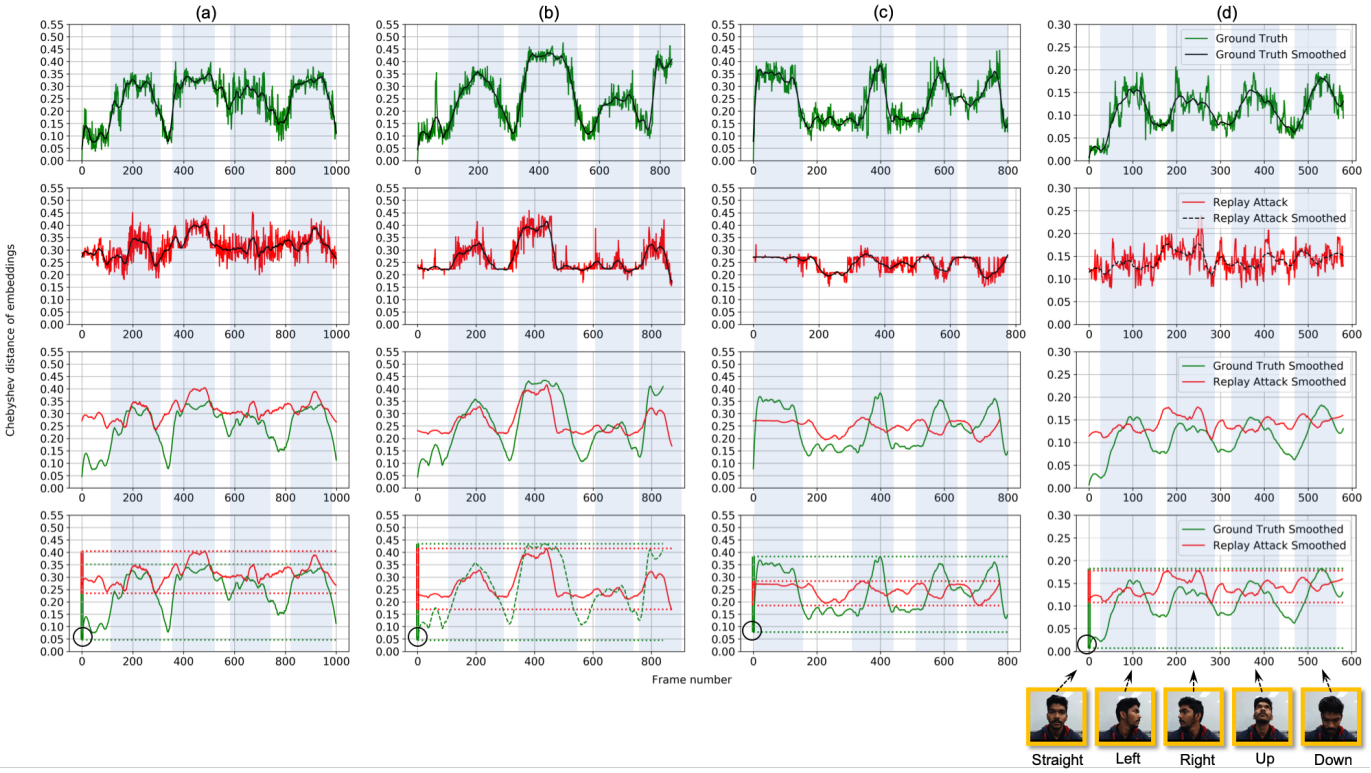



Fig. 10. Change in distance of ResNet-LSTM embeddings, with respect to a known pre-recorded safe embedding, of ground truth video and a replay attack is illustrated. Replay attack is done by showing a high definition video of the user in a 2K digital display. User was asked to turn head left, right, up, down, from the baseline view of looking straight at the camera. Each column from $a$ through $d$ illustrates the study on different videos. Videos for figures ($a$) and ($d$) were taken in well lit rooms while videos for figures ($b$) and ($c$) were taken in darker rooms, with ($b$) lit slightly from above and ($c$) lit from below the face.

the face.

In the ground truth curve, as mentioned earlier, a clear distinction for facial orientation changes are present, with change in slope as changes in orientation from straight to left, then back to straight, then to right, etc. However, from the replay attack plots, we can see abrupt and highly noisy changes from the known reference. Also, even for straight face orientation during frames 0-50, 150-200, etc., the distance from known embedding is always above the known average. This, in a temporal perspective, illustrates the anomalies in the input video stream. This is primarily due to the loss in texture, and increase of contrast and brightness in the replay videos. As such, we look for these anomalous, random and noisy

TABLE IV
COMPARISON OF OUR PROPOSED METHOD WITH PUBLISHED RESEARCH ON FACE ANTI-SPOOFING

| Study | Techniques Used | Summary |
|---|---|---|
| Our proposed study | ResNet-LSTM based embeddings evaluated using Chebyshev distance to do face recognition and face spoofing detection at once, using temporal queues in the face embeddings. Each VIoT AI-camera is standalone and enables a scalable decentralized authentication model only for enrolled users in a microDB. | Realtime face recognition and anti-spoofing against photo and video replay attacks. High scalability in IoT domain. Requires only one encoding per user for secure face recognition. |
| Liu et al. [22] | Deep learning based CNN-RNN model trained to estimate the face depth with pixel-wise supervision, and to estimate a rPPG signal with sequence-wise supervision. A fusion of these depth and rPPG signals was used to distinguish between live vs. spoof faces. | Current state-of-the-art in publicly available datasets for face anti-spoofing on the wild. Face recognition is not available. |
| Bao et al. [19] | Optical flow field of streaming video is analyzed to detect liveliness of the face. | Photo based replay attacks can be mitigated accurately. Since the method relies on precise calculation of optical flow field, illumination changes affects the reliability. Face recognition is not available. |
| Pan et al. [18] | Eyeblink behavior modelled to predict liveliness of the face. | Real-time face anti-spoofing. Inability to deter video replay attacks. Face recognition is not available. |
| Schwartz et al. [23] | Face anti-spoofing solution based on partial least squares and a set of low-level feature descriptors. | Realtime face anti-spoofing on videos and images. Face recognition is not available. |
| Lui et al. [29] | Automatic detection of 3D face masks used for face spoofing. | Comparative performance evaluation of images acquired under NIR and visible illumination for face anti-spoofing. Face recognition is not available. |

behaviors in the streaming video for understanding liveliness factors to deter face spoofing as described systematically in Algorithm 1. Our studies indicate that the biometric traits in the temporal embedding is substantial enough to detect replay-based spoofing attacks. Also, we see that video replay attacks are considerably harder to crack with proper lighting. For example, in Figure 10 (*b*), video recorded in a darker room with light shining from above the face actually allowed for a better spoofing distance pattern while in Figure 10 (*c*), light from below the face allowed better detection of spoofing attacks.

We plan to delve deep into the temporal face feature embeddings in our future research. As such, a comparison of our proposed methods with published research on face anti-spoofing is provided in Table IV. Our decentralized method does both face recognition and face anti-spoofing in the same forward pass of inference on user images. User unenrolled in the microDB of a particular AI embedded IoT camera is not provided access. Users with access to the IoT device is monitored over time using the ResNet-LSTM embeddings and Chebyshev distance function to detect spoofing in a temporal fashion. This provides a decentralized access control system which is highly secure for authenticating a controlled group of individuals.

## VI. Conclusion

In this paper, a temporal face embedding model is proposed and developed to recognize and authenticate individuals. A face anti-spoofing method is described by using the proposed neural network based ResNet-LSTM architecture which allows face embeddings to learn from the changes of the face attributes, orientations, and much more. We train the cascaded residual blocks to extract facial features and map them to a 128 dimensional embedding vector which is fed to the LSTM to learn long-term temporal structure from a sequence of frames captured by a camera(s), and then perform the recognition task. A Chebyshev distance based face anti-spoofing temporal algorithm is developed towards robust real-time face recognition. The ResNet-LSTM model which is designed for on-device or embedded biometric authentication systems is hosted on a decentralized authentication architecture, keeping data securely on local devices and safe from hackers thereby preserving user privacy. Experimental results on YTF dataset demostrates the effectiveness of the proposed ResNet-LSTM model. Also, the face anti-spoofing study on video based replay attack on an in-house replay dataset shows the robustness of the proposed model against adversaries.

## Acknowledgment

This work was supported, in part, by Open Cloud Institute at University of Texas at San Antonio, Texas, USA and by Grant number FA8750-15-2-0116 from Air Force Research Laboratory and OSD, USA. The authors gratefully acknowledge use of the services of Chameleon cloud and Jetstream cloud, funded by NSF awards 1419165 and 1445604 respectively. The authors thank Samuel Silva and Mehrad Jaloli, Ph.D. students at Secure AI and Autonomy Lab, UTSA, for being candidates in the face spoofing studies.

## References

[1] A. Kwaśniewska, J. Rumiński, and P. Rad, "Deep features class activation map for thermal face detection and tracking," in *Human System Interactions (HSI), 2017 10th International Conference on*. IEEE, 2017, pp. 41–47.

[2] P. Rad, M. Muppidi, S. Agaian, and M. Jamshidi, "Systems and methods for secure file transmission and cloud storage," Jun. 14 2018, uS Patent App. 15/574,935.

[3] S. A. Miraftabzadeh, P. Rad, and M. Jamshidi, "Distributed algorithm with inherent intelligence for multi-cloud resource provisioning," in *Intelligent Decision Support Systems for Sustainable Computing*. Springer, 2017, pp. 77–99.

[4] S. A. Miraftabzadeh and P. Najafirad, "Systems and methods for scheduling of workload-aware jobs on multi-clouds," Dec. 14 2017, uS Patent App. 15/620,345.

[5] R. Ramachandra and C. Busch, "Presentation attack detection methods for face recognition systems: a comprehensive survey," *ACM Computing Surveys (CSUR)*, vol. 50, no. 1, p. 8, 2017.

[6] E. Learned-Miller, G. B. Huang, A. RoyChowdhury, H. Li, and G. Hua, "Labeled faces in the wild: A survey," in *Advances in face detection and facial image analysis*. Springer, 2016, pp. 189–248.

[7] S. Zafeiriou, C. Zhang, and Z. Zhang, "A survey on face detection in the wild: past, present and future," *Computer Vision and Image Understanding*, vol. 138, pp. 1–24, 2015.

[8] Z. Erkin, M. Franz, J. Guajardo, S. Katzenbeisser, I. Lagendijk, and T. Toft, "Privacy-preserving face recognition," in *International Symposium on Privacy Enhancing Technologies Symposium*. Springer, 2009, pp. 235–253.

[9] M. Blanton and P. Gasti, "Secure and efficient protocols for iris and fingerprint identification," in *European Symposium on Research in Computer Security*. Springer, 2011, pp. 190–209.

[10] M. Blanton and M. Aliasgari, "Secure outsourced computation of iris matching," *Journal of Computer Security*, vol. 20, no. 2-3, pp. 259–305, 2012.

[11] S. A. Miraftabzadeh, P. Rad, K.-K. R. Choo, and M. Jamshidi, "A privacy-aware architecture at the edge for autonomous real-time identity re-identification in crowds," *IEEE Internet of Things Journal*, 2017.

[12] S. A. Miraftabzadeh, "Real-time adaptive data-driven perception for anomaly priority scoring at scale," Ph.D. dissertation, The University of Texas at San Antonio, 2017.

[13] M. Roopaei and P. Najafirad, *Applied Cloud Deep Semantic Recognition: Advanced Anomaly Detection*. CRC Press, 2018.

[14] M. Roopaei, P. Rad, and J. J. Prevost, "A wearable iot with complex artificial perception embedding for alzheimer patients," in *2018 World Automation Congress (WAC)*. IEEE, 2018, pp. 1–6.

[15] A. D. Torres, H. Yan, A. H. Aboutalebi, A. Das, L. Duan, and P. Rad, "Patient facial emotion recognition and sentiment analysis using secure cloud with hardware acceleration," in *Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications*. Elsevier, 2018, pp. 61–89.

[16] R. Kumar Pandey, Y. Zhou, B. Urala Kota, and V. Govindaraju, "Deep secure encoding for face template protection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 9–15.

[17] A. Kumar Jindal, S. Chalamala, and S. Kumar Jami, "Face template protection using deep convolutional neural network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 462–470.

[18] G. Pan, L. Sun, Z. Wu, and S. Lao, "Eyeblink-based anti-spoofing in face recognition from a generic webcamera," *IEEE 11th International Conference on Computer Vision*, 2007.

[19] W. Bao, H. Li, N. Li, and W. Jiang, "A liveness detection method for face recognition based on optical flow field," in *Image Analysis and Signal Processing, 2009. IASP 2009. International Conference on*. IEEE, 2009, pp. 233–236.

[20] J. Määttä, A. Hadid, and M. Pietikäinen, "Face spoofing detection from single images using micro-texture analysis," in *Biometrics (IJCB), 2011 international joint conference on*. IEEE, 2011, pp. 1–7.

[21] J. Li, Y. Wang, T. Tan, and A. K. Jain, "Live face detection based on the analysis of fourier spectra," in *Biometric Technology for Human Identification*, vol. 5404. International Society for Optics and Photonics, 2004, pp. 296–304.

[22] Y. Liu, A. Jourabloo, and X. Liu, "Learning deep models for face anti-spoofing: Binary or auxiliary supervision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 389–398.

[23] W. R. Schwartz, A. Rocha, and H. Pedrini, "Face spoofing detection through partial least squares and low-level descriptors," in *Biometrics (IJCB), 2011 International Joint Conference on*. IEEE, 2011, pp. 1–8.

[24] Z. Xu, S. Li, and W. Deng, "Learning temporal features using lstm-cnn architecture for face anti-spoofing," in *Pattern recognition (acpr), 2015 3rd IAPR asian conference on*. IEEE, 2015, pp. 141–145.

[25] K. Patel, H. Han, and A. K. Jain, "Secure face unlock: Spoof detection on smartphones," *IEEE transactions on information forensics and security*, vol. 11, no. 10, pp. 2268–2283, 2016.

[26] D. F. Smith, A. Wiliem, and B. C. Lovell, "Face recognition on consumer devices: Reflections on replay attacks," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 736–745, 2015.

[27] 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.

[28] L. Song, M. Zhang, X. Wu, and R. He, "Adversarial discriminative heterogeneous face recognition," *arXiv preprint arXiv:1709.03675*, 2017.

[29] J. Liu and A. Kumar, "Detecting presentation attacks from 3d face masks under multispectral imaging," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 47–52.

[30] H. Li, P. He, S. Wang, A. Rocha, X. Jiang, and A. C. Kot, "Learning generalized deep feature representation for face anti-spoofing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 10, pp. 2639–2652, 2018.

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 630–645.

[32] ——, "Deep residual learning for image recognition," in *Proceedings of the IEEE international conference on computer vision*, 2016, pp. 770–778.

[33] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," *arXiv preprint arXiv:1602.07261*, 2016.

[34] F. A. Gers, N. N. Schraudolph, and J. Schmidhuber, "Learning precise timing with lstm recurrent networks," *Journal of machine learning research*, vol. 3, no. Aug, pp. 115–143, 2002.

[35] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[36] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[37] A. Das, W.-M. Lin, and P. Rad, "A distributed secure machine-learning cloud architecture for semantic analysis," in *Applied Cloud Deep Semantic Recognition*. Auerbach Publications, 2018, pp. 145–174.

[38] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2121–2159, 2011.

[39] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, 2012.

[40] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[41] I. Sutskever, J. Martens, G. E. Dahl, and G. E. Hinton, "On the importance of initialization and momentum in deep learning." *ICML (3)*, vol. 28, pp. 1139–1147, 2013.

[42] H.-W. Ng and S. Winkler, "A data-driven approach to cleaning large face datasets," in *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 343–347.

[43] L. Wolf, T. Hassner, and I. Maoz, "Face recognition in unconstrained videos with matched background similarity," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 529–534.

[44] 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, 2016.

[45] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv preprint arXiv:1312.4400*, 2013.

[46] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 815–823.

[47] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1701–1708.

[48] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition." in *BMVC*, vol. 1, no. 3, 2015, p. 6.

[49] M. Wattenberg, F. Vigas, and I. Johnson, "How to use t-sne effectively," *Distill*, 2016. [Online]. Available: http://distill.pub/2016/misread-tsne

[50] B. G. Amidan, T. A. Ferryman, and S. K. Cooley, "Data outlier detection using the chebyshev theorem," in *Aerospace Conference, 2005 IEEE*. IEEE, 2005, pp. 3814–3819.