LiveScreen: Video Chat Liveness Detection Leveraging Skin Reflection

Hongbo Liu*, Zhihua Li†, Yucheng Xie‡, Ruizhe Jiang‡, Yan Wang§, Xiaonan Guo‡ and Yingying Chen¶
*University of Electronic Science and Technology of China, China
†SUNY at Binghamton, USA
‡Indiana University-Purdue University Indianapolis, USA
§Temple University, USA
¶WINLAB, Rutgers University, USA
Email: hongbo830117@gmail.com, zli191@binghamton.edu, yx11@iupui.edu, ruizjian@iu.edu
y.wang@temple.edu, xg6@iupui.edu, yingche@scarletmail.rutgers.edu

***Abstract*—The rapid advancement of social media and communication technology enables video chat to become an important and convenient way of daily communication. However, such convenience also makes personal video clips easily obtained and exploited by malicious users who launch scam attacks. Existing studies only deal with the attacks that use fabricated facial masks, while the liveness detection that targets the playback attacks using a virtual camera is still elusive. In this work, we develop a novel video chat liveness detection system, LiveScreen, which can track the weak light changes reflected off the skin of a human face leveraging chromatic eigenspace differences. We design an inconspicuous challenge frame with minimal intervention to the video chat and develop a robust anomaly frame detector to verify the liveness of the remote user in the video chat using the response to the challenge frame. Furthermore, we propose resilient defense strategies to defeat both naive and intelligent playback attacks leveraging spatial and temporal verification. We implemented a prototype over both laptop and smartphone platforms and conducted extensive experiments in various realistic scenarios. We show that our system can achieve robust liveness detection with accuracy and false detection rates 97.7% (94.8%) and 1% (1.6%) on smartphones (laptops), respectively.**

I. Introduction

Due to the rapid development of social media and communication technology, recent years have witnessed video chat gradually becoming a convenient and indispensable means for people's daily communication. However, such convenience also makes personal images and videos easily obtained and exploited by malicious users to launch impersonation scam attacks as shown in Figure 1. For example, relatives or friends of international students have been victims of video scam attacks [1], [2] due to their lack of instant means to contact the students living abroad. The attacker usually obtains video footages of an international student from social media or a stolen smartphone and invites the victim (i.e., student's relative or friend) to engage in an appealingly genuine video chat with a muted voice using the stolen video footage. If the victims are convinced, the attackers will claim to run into some financial difficulties or emergencies and ask for money, which would result in irreparable economic damage for the victims. Similarly, there have been online romance scams [3], [4] that reach out to the victims on their social media accounts (e.g., Facebook and WhatsApp) and lure the victims into performing obscene acts in a live video chat while the victims never actually chat with the attacker but a pre-recorded video of someone else. All these video scams are usually premeditated, organized crimes that steal millions, potentially billions, of dollars from vulnerable or lonely people over the internet.

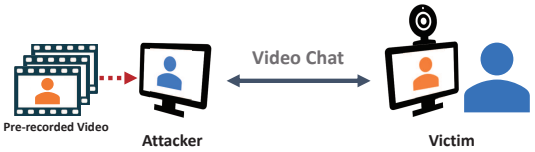


Fig. 1. A video scam attacker uses a pre-recorded video to impersonate a user in a video chat.

Intuitively, video scam attacks may be thwarted by requesting the person in chatting with to respond in accordance with some specific challenges (e.g., blinking, reading words or numbers aloud, head movements, etc.). However, the short video playback used for impersonation attacks may end before the victims are aware of its malicious intent, and the attackers also usually ignore or reject the challenges with reasonable excuses (e.g., broken microphone), which reassures the victims that this is a live video conversation. Existing methods [5]–[9], which benefit from the explosive advancement of image processing and machine learning techniques, can detect media-based facial forgery or impersonation attack leveraging fabricated 2D/3D facial masks [10]–[13]. However, if the attacker impersonates someone by playing a prerecorded video through a virtual camera, existing approaches, even human eyes, are failing to verify the liveness of people appearing in the video chat window. Our paper aims to deal with this challenging problem on liveness detection. Recently, Face Flashing [14] exploits flash frames on the screen to create special reflection light off human faces for user authentication. However, the huge training efforts with respect to each individual is not achievable for common video chat. Moreover, the users have to stay static and have their face very close to screen during the detection process, making it inapplicable for video chat scenarios. Instead, we seek a generic and robust liveness detection solution that can be easily integrated into mobile devices to defend against scam attacks during the video chat.

Towards this end, we propose a low-cost video chat live-

The authors Dr. Hongbo Liu and Zhihua Li have an equal contribution to the paper. Zhihua Li was under the supervision of Dr. Yan Wang at Binghamton University when conducting this work.

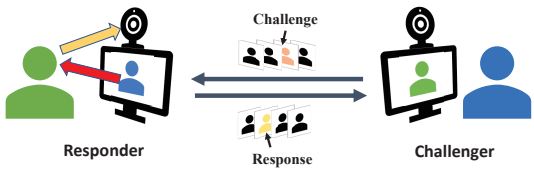


Fig. 2. Challenge-response process of LiveScreen leveraging the inconspicuous light reflected off the human face for video liveness detection.

ness detection system, *LiveScreen*, for various video chat terminals (e.g., smartphones and computers) with different chatting window sizes. Our system is low-cost and easy to integrated into existing video chat terminals because it only requires a screen and a camera, which are essential in the video chat. Unlike existing solutions, LiveScreen leverages the chromatic eigenspace difference to capture the minute changes of the light reflected off the human face, enabling robust video liveness detection under various practical scenarios with complex environmental light conditions, head movements, and non-stationary video background.

The liveness detection process of LiveScreen is shown in Figure 2. The local user customizes some video frames captured by the local camera with a special light pattern. The light pattern works as a *challenge* that will be displayed at the screen of the remote user and projected onto the remote user's face. The reflected light off remote user's face will be captured by the remote camera and sent back to the local user as a *response* along with other normal video chat frames. Thus, LiveScreen can detect the video liveness by examining the change of the light pattern without requiring active participation of the remote user, and thus verify the liveness of the remote user.

To develop such a video liveness detection system, it is critical to detect the response reflected off the human face and determine whether the reflection resulting from the challenge or not. However, the response is usually too weak to be detected, especially under strong ambient light interference and low skin reflectance. Inspired by the remote photoplethysmogram (rPPG) technology [15], we propose to extract chromatic eigenspace difference features from captured video frames to capture the subtle light intensity changes on the human face and achieve accurate response detection in real video chat scenarios. In addition, the challenges should be carefully designed to ensure high signal-to-noise ratio of response while keeping the intervention to the video chat at a minimum. Furthermore, to enable resilient liveness detection, reliable verification strategy is required to defend the system against the naive and intelligent playback attackers.

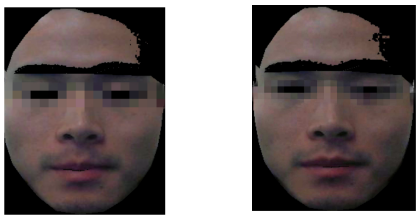
The main contribution of this work is listed as follows:

- We devise a non-invasive, low-cost and lightweight liveness detection system, which can be easily integrated into existing video chat applications without additional devices.
- We extensively explore the light reflected off human skin and design an inconspicuous challenge that can minimize the interference to the users' viewing experience in video chat.
- Our unique chromatic eigenspace difference feature is capable of tracking the light intensity changes regardless of various impact factors.
- We propose resilient defense strategies that leverage the spatial and temporal verification on the light intensity changes in video chat frames to defend our system against types of attacks.
- We build a prototype video chat application integrating LiveScreen. Extensive experiments on laptops and smartphones demonstrate that our system can accurately detect the video scam attacks under practical scenarios (e.g., different chatting window sizes, light conditions, and body movements).

II. Related Work

Many liveness detection methods have been proposed to defend against various types of scam attacks. Some existing liveness detection methods [16]–[21] can identify fabricated face masks/3D head model based on representative facial features. However, no matter how realistic the fabrication of forged faces are, these fabrications either look unnatural in a video chat or incur high cost on materials and manufacturing, making them easily detected by real people. Dynamic attackers can prepare a video beforehand and then either play the video in front of a real webcam or stream the video through a virtual webcam [22]. To detect such attacks, existing solutions rely on texture analysis [23]–[26] and depth-characteristics [7], [27], [28] to detect a forged face displayed on a screen, but the computational cost is usually high. If the prerecorded video is streamed via a virtual webcam by attackers, the prior solutions will fail. Intuitive solutions [29], [30] require explicit real-time interaction among the participants in a video chat (i.e., blinking, reading words or numbers aloud, hands movements, etc.). However, the attackers can ignore or reject the challenges to verify themselves with reasonable excuses, or carefully prepared video playbacks that include required interactions. Some approaches [31]–[33] propose to integrate different biometric traits collected from multiple sensors at the attacker's end for consistency check. For example, Biggio [31] proposed to fuse fingerprint and face recognition techniques to determine the liveness of the remote user; FaceLive [32] defends against the video playback attacks by performing consistency check between built-in inertial sensor readings and the head-pose changes inferred from the video frames. However, the above studies either need to access inertial sensors or require the cooperation of the remote user. Even worse, the attackers can fake the sensor data transmitted along with the video playback, and fail FaceLive.

To overcome the above limitations, researchers recently proposed to achieve liveness detection by leveraging reflection light off human faces. Patrick et al. [34] proposed to perform liveness detection based on the face reflectance resulting from a flashlight, but this approach requires the assistance of a flashlight, which may not be readily available or uncontrollable by local user. Face Flashing [14] exploits dedicated flash frames on the screen instead of an additional flashlight to create special reflection light off human faces and then perform liveness detection leveraging deep learning techniques. However, the success of this method is built upon huge training efforts on the face reflection pattern with respect to individual people to be authenticated, which is unachievable in common video chat scenarios. Moreover, the people to be



(a) Without challenge (b) With challenge

Fig. 3. Comparison of video frames showing no observable light intensity change in the volunteer's face when it is affected by the challenge.

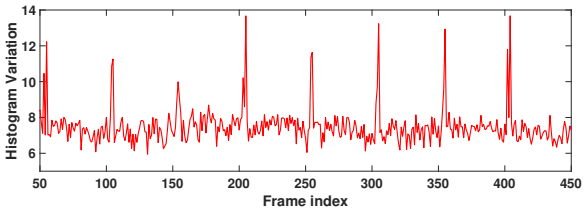


Fig. 4. RGB histogram variance of video frames showing clear changes when the volunteer's face is affected by the challenge.

authenticated have to stay static during the process of liveness detection, making it inapplicable for video chat. Therefore, a more generic and robust liveness detection system without the cooperation of remote user is highly required.

III. Attack Models & Feasibility Study

A. Attack Models

This work focuses on two types of impersonation scam attacks in a video chat: *Naive Playback Attack* and *Intelligent Playback Attack*. In both attacks, the attacker invites a local user to a video chat in the name of a person who is close to the user. During the video chat, the attacker replaces the live video stream with a pre-recorded video clip of the person and communicates with the user using text. Once the user is convinced that this is a live video chat, the attacker cheats the user for money or something else of value. We assume that the attacker does not use voice communication as it exposes the fraudulent nature. The attacker also ignores or rejects the user's challenges (e.g., making a facial expression, blinking, or nodding) with reasonable excuses (e.g., broken microphone or distracting the user by changing the topic of conversation). Neither type of attack has access to the local user's device or software.

Naive Playback Attack (NA). The naive attacker does not have the capability to process the video frames from the user or modify the video frames that are sent to the user. To launch the attack, the attacker can either (1) play the pre-recorded video frames in front of the camera with a smartphone or laptop (denoted as **NA-1**) or (2) stream the pre-recorded video frames instead of the video frames from the real webcam through a virtual camera [22] to emulate a live video chat (denoted as **NA-2**).

Intelligent Playback Attack (IA). Compared to the naive attacker, the intelligent attacker has full knowledge of the proposed system. In addition, the attacker has the capability to process the video frames from the user and modify the video frames that are sent to the user. Therefore, the attacker can detect the challenges embedded in the video frames and synthesize a valid response to the challenges by modifying the pixel intensity, for example, increasing the red-channel intensity of the facial area in the prerecorded video.

B. Feasibility Study

Model of Reflected Light. The image sensors on a digital camera consist of a set of pixels, which capture the reflected light of the object to form an image. Each pixel represents the intensity response of the sensor to the incoming reflected light from a point x in a scene. For simplicity sake, the per-pixel light intensity response can be approximated with a linear diagonal map based on Von Kries model [35] as:

$$I_c(x) = E_c(x) \times R_c(x), c \in \{r, g, b\}, \quad (1)$$

where $E_c(x)$ and $R_c(x)$ are the illuminant spectral power and reflectance of a specific color channel c. Note that the light intensity response E_c of pixel x is a mixture of the light intensity response resulted from multiple illuminant sources.

Considering a typical scene of a live video chat, where a user usually has his/her face in front of a screen and camera, the image of the user's face captured by the camera has the light intensity response for each pixel x as follows:

$$\hat{I}_c(x) = E^s_c(x) \times R_c(x) + E^e_c(x) \times R_c(x), c \in \{r, g, b\}, \quad (2)$$

where E^s_c is the illuminant source from the screen and E^e_c is the mixture of all the environmental illuminant sources excluding the screen. Given two adjacent frames capturing the same scene, there should be little difference in the light intensity due to the transient time interval between them in a video stream. Equation 2 implies that if we could adjust E^s_c to emit a special light pattern that is captured by one of the two adjacent frames, the original light intensity distribution $\hat{I}_c$ will be overrode. Since the skin usually reflects more light from the screen than other objects in the scene do due to its close distance to the screen, it is possible to detect the liveness of a video chat by comparing the intensity distribution of the light reflected off human faces between two adjacent frames.

To validate the feasibility of the proposed idea, we conduct preliminary studies on the light reflected off the human face by varying the light intensity of the video frames displayed on the screen of a laptop. Specifically, a volunteer sits in front of a laptop with a distance of 40cm to the screen and built-in camera. The laptop plays a video clip on the screen containing the frames with modified light intensity (i.e., for every 50 frames, set the light intensity of the red channel to its 150%) to imitate a video chat with challenges. Meanwhile, the built-in camera is recording a video of the volunteer's face. Figure 3 shows that the response is unnoticeable, no obvious change of light intensity on the volunteer's face when comparing the frames with and without challenge. We further manually identify the skin area in the volunteer's face in the recorded video and calculate the variance of RGB histogram [36] based on all the pixels in the skin area. Figure 4 clearly shows that the RGB histogram of video frames has significant changes when a challenge is projected onto the volunteer's face, and provides strong evidence on the feasibility of using light intensity changes on human faces to perform video chat liveness detection.

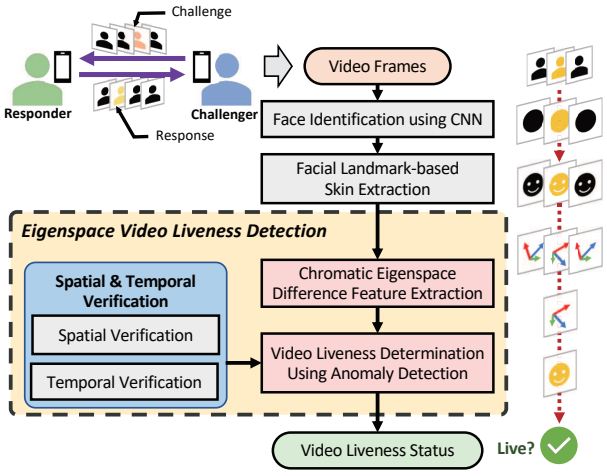


Fig. 5. Overview of LiveScreen.

IV. System Design and Challenges

The goal of this work is to protect users from scam attacks by enabling liveness detection in a video chat. Towards this end, we develop a system that can automatically initiate challenges and detect responses by leveraging the frames in video chat. Specifically, we utilize the frames in video chat as a medium to carry the challenge (i.e., challenge frame), which is designed to create a special light intensity pattern, while keeping the minimum intervention to the video chat. On the remote screen, the challenge frame is projected onto the remote user's face, resulting in a significant change in the intensity of the light reflected off the remote user's face. The remote camera captures the reflected light as the response to the challenge and then sends it back to the user along with the normal video frames in the video chat. Our system can effectively identify the video frames with significant light intensity change caused by the challenge and determine the liveness of the video chat.

The architecture of our system is shown in Figure 5. The system first sends the challenge to the remote chatting end, which plays the challenge on its screen and sends the video frames captured by its camera back to the system. Our system continuously takes the video frames from the remote chatting end as the input. For each frame, the system first performs the *Face Identification using Convolutional Neural Network* to locate the human face in the frame by using a pre-trained convolutional neural network model. The human face is the region of interest (ROI) in the video frame that concentrates most of the response, which would facilitate the robustness of our liveness detection. Then to further boost the detection accuracy, we employ the *Facial-landmark-based Skin Extraction* to exclude the non-skin parts on the identified face area and extract the skin-related pixels. Next, the system performs the *Chromatic Eigenspace Difference Feature Extraction* to derive the chromatic eigenspace difference feature, which utilizes eigenspace distance in the RGB color space to capture the minute light intensity changes caused by the challenge. Last, we conduct *Video Liveness Determination Using Anomaly Detection* to identify valid response based on the time series of the eigenspace difference features and determine the liveness

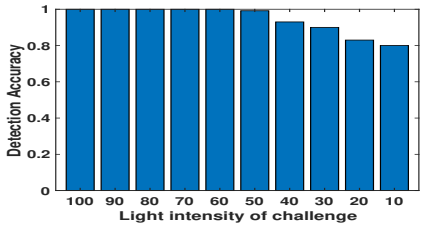


Fig. 6. Response detection accuracy using the challenge with different levels of light intensity.

of a video chat. Furthermore, in order to defend against the attacks launched at the remote end, we also adopt two defense strategies, *Spatial Verification* and *Temporal Verification*. The *Spatial Verification* examines the light intensity distribution on human face and background behind the human face in the received frames to defeat the naive playback attacks. The *Temporal Verification* monitors the round-trip delay time (RTT) between two video chat users and detects the intelligent attacker based on the statistics of the time intervals between consecutive video frames.

V. Challenge Frame Design

Inconspicuous Light Intensity Design. To enable inconspicuous challenge-response-based liveness detection, we need to design the challenge frame to not contain noticeable artifacts but still facilitate reliable liveness detection. In this work, we seek to generate the challenge frame by enhancing the light intensity of the selected video chat frames in the RGB color space. Note that this approach is easy to implement and does not incur extra network overhead. We find that the challenge frame with an enhanced red channel is particularly effective for our liveness detection because human skin generally has higher reflectance to the red light (i.e., light with the wavelength between 630nm-700nm) [37]. Along with this direction, we explore the feasibility of liveness detection using the challenge frames with different light intensities of the red channel and our response detection method introduced in Section VI. Figure 6 presents the percentile of accurately detected responses (i.e., accuracy introduced in Section VIII) when we increase the intensity of the red channel of the challenge frame. We can see that even when the light intensity is increased as low as 10%, the detection accuracy is over 80%. Note that the challenge frame has no obvious difference from the original frame when the light intensity is no more than 50%. The results indicate that our system can detect the liveness of a video chat using inconspicuous challenges. If not mentioned otherwise, we increase the intensity of the original frame's red channel by 50% to generate the challenge frames.

Robustness Design. After the light intensity of the inconspicuous challenge frame is determined, it is essential to add more redundancy of the challenge frame to enhance its robustness due to the security concern and hardware limitations. Specifically, we have multiple challenge-frame transmissions during video chat for reliable liveness detection, and each transmission is allocated at a random time slot, which aims to avoid the arrival time of the challenge frame being predicted by an attacker. Furthermore, due to the limited frame rate of the camera during video chat, the expected

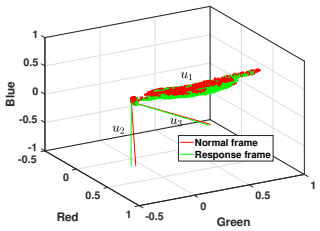


Fig. 7. Illustration of eigenspace representation of adjacent frames.

response frame may not be captured if the challenge frame has a short retention period on the screen at the responder end. To overcome this limitation, we extend the length of challenge frame by covering several consecutive frames to avoid missing the expected response frame.

Impact of Network Condition. We note that network condition also has a significant impact on our liveness detection. When a live video chat application detects that the network condition is poor (e.g., low bandwidth or long latency), it usually switches to high compression ratio with lossy video compression algorithm or low frame resolution. In either case, it will result in low-quality video frames with approximated pixel values, which lead to significantly reduced light intensity in the response and lower liveness detection accuracy. Therefore, we design our system to keep monitoring the network conditions by using python psutil tools [38]. If a poor network condition is detected, the system will automatically suspend the challenge-frame transmission until the network condition becomes better.

VI. Eigenspace Liveness Detection

Given the received frames, our system first identifies the human face with a pre-trained Convolutional Neural Network (CNN) model based on Labelled Faces in the Wild (LFW) dataset [39], and then extracts the skin area with face landmarking method [36] based on iBUG 300-W dataset [40] to remove the ambient light interference. Next, we introduce a novel chromatic eigenspace difference feature and response detection method that can capture minute light intensity changes caused by the challenges and detect the video chat liveness in practical environments, respectively.

A. Chromatic Eigenspace Difference Feature Extraction

After the face identification and skin extraction, we need to determine whether the light reflected off the human face is affected by the challenge or not. This is a nontrivial task because the light reflected off human faces is affected by various factors, such as ambient light, head orientations, and skin colors. Therefore, a simple comparison on the light intensity (e.g., using histogram) of skin-related pixels between adjacent frames is not effective and robust enough to detect the valid response to the challenge.

To overcome the above impacts, we propose to use a new feature, named chromatic eigenspace difference, extracted from the skin-related pixels between adjacent video frames. The proposed chromatic eigenspace difference feature is well fitted to our problem because (1) it is robust for different skin tones and light interference; and (2) it utilizes skin-related pixels without any averaging operation, and each pixel contributes

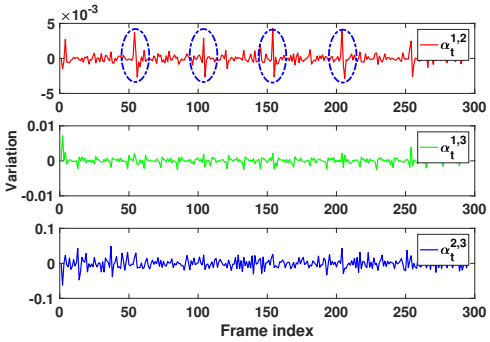


Fig. 8. Effectiveness study of the eigenspace distances $\alpha_t^{1,2}$, $\alpha_t^{1,3}$, and $\alpha_t^{2,3}$.

to the extracted feature. The intuition behind using this feature for liveness detection is that the colors of skin-related pixels gather into certain clusters in the RGB color space due to the similarity of skin-related pixels. We can decompose the RGB color of the skin-related pixels into three primary eigenvectors, which represent the most significant characteristics of the light reflected off the human face. A similar feature has been used to extract the remote photoplethysmogram (rPPG) signals [15] from human faces under various scenarios (e.g., head motions and skin colors), suggesting its effectiveness in extracting target light signals reflected off human faces regardless of various impact factors.

Specifically, the chromatic eigenspace difference feature is obtained through measuring the distance between the eigenspaces of two adjacent video frames, which is derived from the light intensity of skin-related pixels. We first calculate the auto-correlation of the skin-related pixels in a video frame as $C = S^T S/p$, where p is the total number of skin-related pixels, S is a $p \times 3$ matrix vectorized from RGB channels of the skin-related pixels, and T denotes the transpose operation on matrix. Then, we obtain the eigenvectors of skin-related pixels through the eigen decomposition of C as shown below:

$$C \cdot U = \Lambda \cdot U \quad s.t. \quad |C - \Lambda \cdot I| = 0, \tag{3}$$

where U and Λ denote the eigenvectors and eigenvalues, respectively, I is an identity matrix, and $|\cdot|$ denotes the matrix determinant. The eigenvectors in U are orthogonal to each other and are used to construct the eigenspace of the skin-related pixels. Intuitively, the frames with different color distributions have a set of eigenvectors with different orientations, resulting in different eigenspaces. Figure 7 shows the eigenspaces of two video frames (i.e., Frame 2 contains the valid response to the challenge while Frame 1 does not). The red and green marks and lines in the figure correspond to the skin-related pixel intensities in RGB color space and corresponding eigenvectors U of the two frames. We can clearly observe the differences in orientation between the two sets of eigenvectors, suggesting that we can detect the valid response by comparing the eigenspaces between two adjacent frames.

We next derive the chromatic eigenspace difference feature in a time series of video frames. Given two adjacent frames at time t and t', the corresponding eigenspaces are $U_t = \left[u_t^1, u_t^2, u_t^3\right]$ and $U_{t'} = \left[u_{t'}^1, u_{t'}^2, u_{t'}^3\right]$, each entry in the eigenspace corresponding to one color channel in RGB

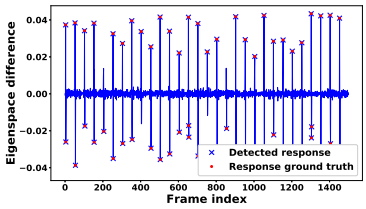
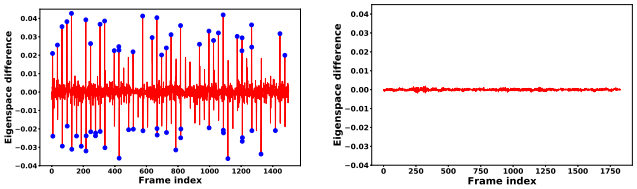


Fig. 9. Illustration of the effectiveness of anomaly frame detection.



(a) NA-1 (b) NA-2

Fig. 10. Feasibility of using the spatial verification to detect naive playback attacks.

color space. $\alpha_t^{i,j} = cos(\theta^{i,j})$ is defined as the eigenspace distance measuring the difference between two adjacent frames, where $\theta^{i,j}$ represents the angle between $u_{t'}^i$ in $U_{t'}$ and u_t^j in $U_t, i = 1, 2, 3$. When there is minute light intensity difference between two adjacent frames, it will cause subtle angular changes (e.g., δ) in the eigenspace. Because $u_{t'}^i$ and u_t^i are linearly correlated to each other in eigenspace (i.e., $\theta \approx 0°$), the eigenspace distance $\alpha_t^{i,i}$ only slightly varies around 1 with small angular changes due to the gentle changes of $cos(\cdot)$ function when θ is around $0°$. Since the response embedded in the received frame is usually weak, $\alpha_t^{i,i}$ is not suitable for response detection. Then we resort to the eigenspace distance between the orthogonal eigenvectors of two adjacent frames, $\alpha_t^{i,j\neq i}$, where θ is around $90°$. Comparing to $\alpha_t^{i,i}$, subtle angular changes will result in significant variations on $\alpha_t^{i,j\neq i}$ due to the steep changes of $cos(\cdot)$ function at $90°$. Our preliminary study as shown in Figure 8 finds that $\alpha_t^{1,2}$, comparing to $\alpha_t^{1,3}$ and $\alpha_t^{2,3}$, has the most significant difference when a valid response is contained in one of the two adjacent frames. Thus, we choose $\alpha_t = \alpha_t^{1,2}$ as the chromatic eigenspace difference feature based on any two adjacent frames in a time series of video frames for response detection.

B. Response Detection

Next, we detect the response based on the time series of extracted chromatic eigenspace differences. Specifically, we adopt Hodrick-Prescott filter [41] to remove the cyclical component and ambient interferences for a smoothed-curve representation of the time series, and then use Median Absolute Deviation (MAD) test to detect the response frames.

Let $\alpha_t = \tau_t + c_t$ for $t = 1, 2, \cdots, N$, denote the time series of chromatic eigenspace differences consisting of a trend component τ_t and a cyclical component c_t, where τ_t can be used to locate the abnormal changes in time series, and c_t reflects the irrelevant scene variation. The trend component is obtained by solving the following minimization problem:

$$\min_{\tau} \left(\sum_{t=1}^{N} (\alpha_t - \tau_t)^2 + \lambda \sum_{t=2}^{N-1} (\tau_{t+1} - 2\tau_t + \tau_{t-1})^2 \right), \quad (4)$$

where the first term is the sum of the squared deviations of α_t from the trend and the second term, which is the sum of squared second differences in the trend, is a penalty for changes in the trend's growth rate. The larger the value of the positive parameter λ, the greater the penalty and the smoother the resulting trend will be.

Given the filtered chromatic eigenspace difference measurements τ, we adopt Median Absolute Deviation (MAD) test [42] to detect response. Our empirical study finds that the chromatic eigenspace difference extracted from video frames does not follow Gaussian distribution due to the complex environmental impacts. Thus, MAD, as a robust measure of variability for Non-Gaussian signals, is a more suitable detector with less impact from anomaly measurements. Additionally, since the round-trip time starting from challenge emission to response reception is usually very short, so the detection process is restricted within a short time window W after the challenge frame is sent. Figure 9 shows the detection results on a short video footage containing the valid responses, indicating our method accurately detect the responses based on the time series of chromatic eigenspace differences.

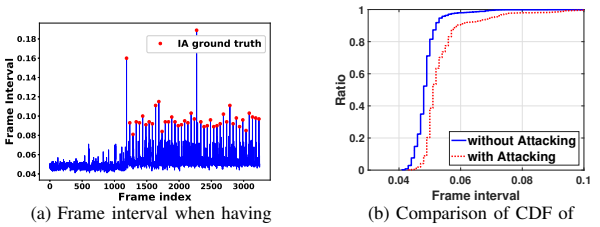
VII. Spatial & Temporal Verification

We design the spatial and temporal verification methods based on the spatial and temporal distribution of valid response to defend against the playback attacks.

Spatial Verification. Since users usually face to their cameras during a video chat, the valid response should only appear in the human face and no other area (e.g., the background) in the video frame should have the similar response to the challenge. Therefore, with the presence of naive playback attacks, if the attacker utilizes virtual camera to transmit the prerecorded video, no valid response should appear in the received video frame, including the human face; if the attacker utilizes a playback device (e.g., a tablet or smartphone) to play the pre-recorded video in front of the camera, the entire frame (i.e., including the human face and the background) should contain the response because the flat screen of the playback device has the same distance to the video chat screen and camera. Thus, we seek to examine the spatial distribution of light intensity in the face area and non-face area to detect the naive playback attacks. In particular, let Z_t^S and $Z_t^{\overline{S}}$ denote the modified Z-score measurements with respect to the face area and non-face area in the received frames, respectively. Thus, a successful detection of the naive playback attacks (NA-1 and NA-2) should satisfy the following conditions:

$$\begin{aligned} &\textbf{NA-1}: Z_t^{\overline{S}} \geq \gamma, \exists t \in (T, T+W], \\ &\textbf{NA-2}: Z_t^{S} < \gamma, \forall t \in (T, T+W], \end{aligned} \quad (5)$$

where T is the timestamp when the challenge is sent, W is the window size for expected valid response and γ is the empirical threshold for anomaly frame detection in Section VI-B. The system tries to detect the response in both facial and non-facial (i.e., background) areas in each frame. If the response is detected in the non-facial area, the system determines there is a naive attack.



(a) Frame interval when having intelligent playback attacks

(b) Comparison of CDF of frame interval

Fig. 11. Feasibility of using the temporal verification to detect intelligent playback attacks.

Temporal Verification. In order to defend against the intelligent playback attacks, we develop a temporal verification scheme, which determines whether the response is legitimate or not based on the time delay between consecutive frames. Intuitively, when there is no intelligent playback attack, the response is naturally captured by the camera at the responder and then streamed to the challenger without any obvious delay. Therefore, the challenger should observe stable intervals between every two adjacent frames. When the intelligent playback attack is launched, the attacker inevitably needs to perform the following operations: 1) detect the challenge frame, 2) generate the synthetic responses, and 3) encode the synthetic responses into the video stream sent to the challenger. Such operations introduce non-negligible extra processing time and temporarily increase the interval between the synthetic response frame and its following frame. Inspired by the above analysis, we develop the temporal verification method to detect the intelligent playback attack by examining the variation of the time interval between frames. Specifically, we detect the intelligent playback attacks (IA) if the following condition is satisfied:

$$\textbf{IA}: t \geq T + W, \text{if} \quad Z_t^S \geq \gamma. \tag{6}$$

Figure 11 shows the frame intervals before and after the responder starts the intelligent playback attacks. We can clearly observe that the frame intervals have significant large peaks after the 1200th frame when the responder begins attacking, each peak corresponds to the occurrence of the intelligent playback attacks. Figure 11(b) shows there are significant differences between the CDFs of the frame intervals affected by the intelligent playback attacks and those obtained from a typical on-campus WiFi network (i.e., 72 hours), confirming that we can detect the intelligent playback attacks using the proposed temporal verification.

Note that LiveScreen continuously performs the spatial and temporal verification on every received frame in a separated thread. The user will be notified when the system detects a scam attack with a very short delay (i.e., less than 200ms) after sending a challenge frame, which is negligible compared to the time that the attacker needs to cheat the user (ie., more than ten seconds).

VIII. Performance Evaluation

A. Experimental Setup

Prototype. To evaluate the effectiveness and robustness of our LiveScreen system, we build a prototype system on both laptop and smartphone platforms with Python. Specifically, a video chat application is developed to incorporate our challenge-response process between two mobile devices. The connection between devices is established through the built-in Python socket interface on wired or wireless local area network. The liveness detection process is implemented by leveraging Python image processing and machine learning libraries (i.e., OpenCV, dlib, etc.).

Fig. 12. Illustration of the real-life experimental environments.

Hardware. Our experiments involve two laptops and three smartphones as the responder, which include *Laptops*: a Lenovo Thinkpad E430 (14" screen, 3MP camera), a Dell Latitude E6430 (14" screen, 1.3MP camera) and a cobra CDR 840 5MP external camera; *Smartphones*: a Nexus 6 (5.96" screen, 2MP camera), a VIVO XI+ (6.2" screen and 16MP camera), and a Sony Xperia XA2 (5.2" screen, 8MP camera). We use another laptop (i.e., Dell Latitude E6430) as the challenger to send a challenge for every 50 frames to the responder during a video chat. The responder is set to record at 20FPS on laptops and 30FPS on smartphones, respectively.

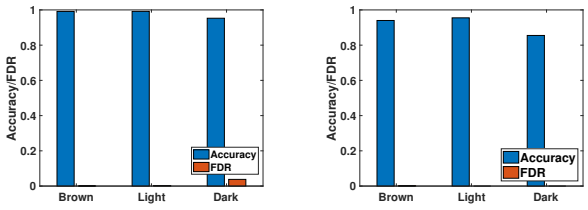
Participants and Scenarios. We recruit 30 volunteers with different ages (i.e., 20 to 40 years) and skin colors, including 21 brown, 4 white, 5 dark skin individuals. The experiments are carried out in both static and dynamic scenarios. In the *Static Scenarios*, the responder device is fixed on a desk and volunteers are asked to sit still in front of the responder with a default distance of 40cm and 20cm to a laptop and a smartphone, respectively. In the *Dynamic Scenarios*, we consider both head and device dynamics. For head dynamics study, the volunteers sit in front of a fixed responder and turn their heads $\pm 30^\circ$ horizontally at moderate speed (i.e., 2s per round) and fast speed (i.e., 1s per round) to mimic the movements of looking around during a video chat. For device dynamics study, the volunteers hold a smartphone and walk around while keeping their faces at the default distance to the smartphone. We also evaluate the system under different real-life environments as shown in Figure 12. For all the scenarios, we record the video from the responder for about 1min/person, and in total over 780min of video data are collected.

B. Evaluation Metrics

We use *Accuracy* and *False Detection Rate (FDR)* to evaluate our system performance. Accuracy is defined as the ratio between the number of correctly detected responses and the total number of challenge frames. FDR is defined as the ratio between the number of incorrectly detected responses and the total number of challenge frames.

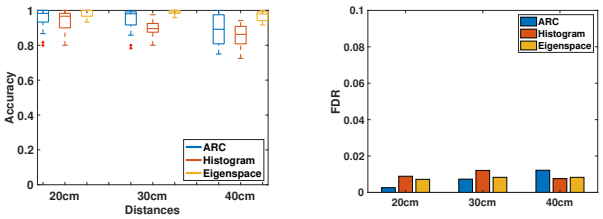
C. Static Scenario Results

Impact of Skin Colors. Since different colored skins have different reflectance, we study the impact of skin colors on the performance of LiveScreen by focusing on three different



(a) Smartphone (b) Laptop

Fig. 13. Detection performance with different skin colors.



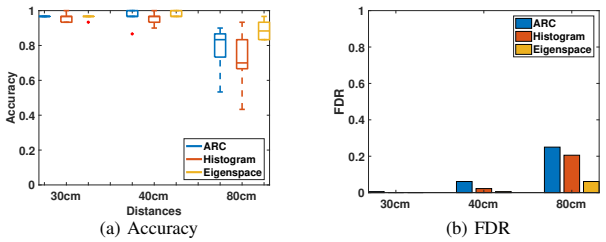
(a) Accuracy (b) FDR

Fig. 14. Smartphone performance at different distances.

colors, namely *brown*, *light*, and *dark*. Figure 13 presents the average accuracy and FDR for the liveness detection results on the three skin colors using smartphones and laptops. We find that the overall performance on brown and light skin is better than that of the dark skin. This is because dark skin has stronger spectral absorption in the visible light spectrum, resulting in reduced reflection of light. Overall, the above results indicate that LiveScreen is effective and robust for different skin colors.

Impact of Face-to-Screen Distances. To study the impact of the distance between the face and screen, we set the distances as: Laptops - 30cm, 40cm and 80cm; Smartphones - 20cm, 30cm and 40cm, which are inline with the normal distances in most daily video chats. We compare the performance of using chromatic eigenspace difference feature (denoted as *Eigenspace*) in our system with two other features, the histogram of RGB channels (denoted as *Histogram*) and the average intensity of red channel (denoted as *ARC*). As shown in Figure 14, on the smartphone platform, our eigenspace feature can achieve over 98% median accuracy with lower than 2% false detection rate at different distances, while the median accuracy of histogram-based decreases from 96.7% to 86.25% as the distance increases. Moreover, the interquartile ranges for our eigenspace feature are 0.035, 0.018 and 0.052 when the distance is 20cm, 30cm and 40cm, respectively, which indicates the high stability of our system under different distance settings. A similar observation is also found on laptops at different distances in Figure 15, indicating that eigenspace-based method is more robust to ambient light interference than the other two methods.

Impact of Ambient Light. To study the impact of different ambient light intensities, we place a LED light (Philips Energy Light HF3418) in front of the user in a video chat and set 3 light intensity levels (i.e., low, medium and high) to emulate different signal-to-noise-ratios. As the light intensity increases, we can observe a decreasing trend on the detection accuracy for eigenspace (i.e., from 95% to 90%), histogram-based method (i.e., from 90% to 80%) and ARC method(i.e., from 90% to 80%) in Figure 16. But eigenspace method



(a) Accuracy (b) FDR

Fig. 15. Laptop performance at different distances.

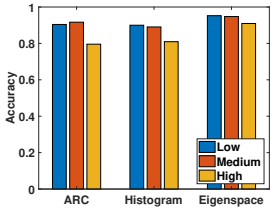


Fig. 16. Impact of ambient light intensity.

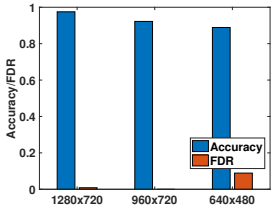


Fig. 17. Impact of video frame dimensions.

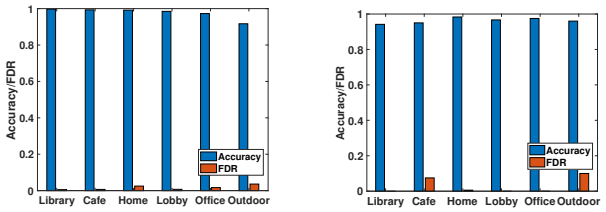
always outperforms histogram-based method and ARC method under different ambient light intensities. The aforementioned observations confirm that eigenspace method is more accurate and robust in the scenarios with lower signal-to-noise-ratio.

Impact of Video Frame Dimensions. The video frame dimension is also a key factor affecting the liveness detection capability of LiveScreen, as the smaller video frame dimension that we use, the lower light intensity is introduced by the challenge on the screen. We examine the performance with different video frame dimensions on a laptop (i.e., 1280x720, 960x720 and 640x480). As shown in Figure 17, our liveness detection performance improves as the video frame dimension increases. In particular, as the video frame changes from 640x480 to 1280x720, the average accuracy increases from 88.9% to 97.5% and FDR decreases from 8.9% to 0.8%, respectively. The results demonstrate that our system is capable to capture subtle face reflections and robust to different video frame dimensions.

Real-Life Environments Study. To validate the scalability of LiveScreen, we carry out the experiments under six common real-life environments (i.e., library, coffee store, home, lobby, home, outdoor) and compare the results in Figure 18. For all the indoor environments, our system always achieves high detection accuracy of 94.5% on both smartphone and laptop platforms with less than 2.5% FDR. For outdoor environments, our method still maintains over 90% detection accuracy but relative high FDR of 4% and 10% on smartphones and laptops, respectively. We notice that the higher FDR happens when there is strong sunlight projected on the human face, which creates strong interference on detecting valid response. Since people usually do not have video chat under strong sunlight, our system still achieve high effectiveness and scalability on liveness detection in most real-life scenarios.

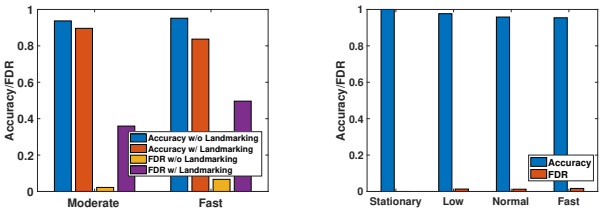
D. Dynamic Scenario Results

Impact of Head Movement. For dynamic status, we first study how head movement affects the detection performance on laptop platform. The reflection pattern on human face



(a) Smartphone (b) Laptop

Fig. 18. Detection performance in real-life scenarios.



(a) Head dynamics (b) Device dynamics

Fig. 19. Performance in dynamic scenarios.

changes as head moves, thus undesired light intensity variations will be involved in the captured frames and bring about the ambiguity on detecting the response. Note that our system may lose track of the skin due to the incomplete facial landmarks when user's face turns to one side. So we also compare the detection performance with and without facial landmarking. As shown in Figure 19 (a), our system performs better without facial landmarking than with landmarking (i.e., 94% vs 89% for moderate movement and 95% vs 84% for fast movement) and has lower FDR (i.e., 2% vs 40% for moderate movement and 7% vs 50% for fast movement). Such observation indicates that poor skin extraction with facial landmarking poses a negative impact on system performance, but our system can still gain high accuracy regardless of head movement during video chatting by automatically switching between using facial landmarking or not.

Impact of User Motion. It is common that people may use their smartphones for video chat while in motion, which will result in video frame jitter and thereby affect the detection performance. Therefore, it is also critical to study how user motion affects the detection performance of our system. Specifically, we conduct the experiments under four motion status (i.e., stationary, low-speed, normal-speed and fast-speed walking) while keeping default distance between smartphone screen and user's face in Figure 19(b). Specifically, when the walking speed is relatively slow, our system maintains a high accuracy of 97.67%, which is only a little bit worse than that of stationary status. For normal speed, although frame jitter is more obvious, our method stills perform well with an accuracy of 95.83%. Even under fast-speed walking, an accuracy above 95% still holds. The encouraging results indicate that our method is robust under various motion status.

E. Performance on Attack Detection

Finally, we evaluate the performance of LiveScreen's defense mechanism under the naive and intelligent playback attacks. To facilitate the evaluation, we define the *Attack Detection Rate (ADR)* as the ratio between the number of accurately detected attacks and the total number of effective attacks (i.e., the total number of challenges frames), and *Miss Detection Rate (MDR)* as the ratio between the number of incorrectly detected attacks and the total number of effective attacks. We conduct the experiments with each of the three attackers (i.e., NA-1, NA-2, and IA) performing attacks on 200 challenges sent in a video chat protected by LiveScreen. Note that we use a smartphone to playback a victim's pre-recorded video in front of the camera to launch the NA-1.

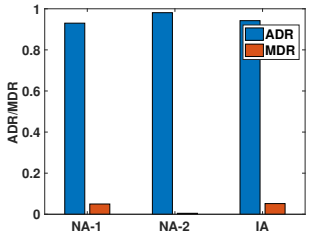


Fig. 20. Performance of attack detection.

As shown in Figure 20, our system can achieve high accuracy and low miss rate on detecting NA-1, NA-2, and IA. In particular, the ADR for detecting the three attackers are 93%, 98%, and 94%, and the MDR for detecting the three attackers are 5%, below 1%, and 5%, respectively. The detection accuracy for NA-1 is a bit worse than that of NA-2. This is because the smartphone screen that we used to perform NA-1 may create the mirror-like reflection not pointing to the camera, which results in weak reflected light in the entire frame, including the face area and non-face area. However, in such cases, our system still detects the attacks but considers them as NA-2 attack as there is no response detected in the face area. Overall, the results confirm the effectiveness of our defense strategy levering spatial & temporal verifications.

IX. Conclusions and Future Work

In this paper, we addressed the problem of liveness detection in a video chat with the presence of video playback attacks. Specifically, we propose a novel video chat liveness detection system, LiveScreen, to protect the users from impersonation scams. The proposed system can track the inconspicuous light changes reflected off the skin of a human face leveraging chromatic eigenspace difference features and determine the liveness of video chat with a robust anomaly detector. We also propose inconspicuous challenge design with minimal intervention to the video chat. Furthermore, a resilient defense strategy is developed to defeat both naive and intelligent playback attacks leveraging spatial and temporal verification. We implement a prototype video chat application to integrate LiveScreen on both laptop and smartphone platforms. Extensive experiments involving 30 volunteers show that LiveScreen achieves high detection accuracy with low false detection rate in various real scenarios. In addition, a comprehensive study of different impacts (e.g., distance, skin color, user motion, etc.) further confirms the robustness of the proposed system.

X. Acknowledgments

This work was partially supported by the National Science Foundation Grants CNS-1566455, CNS-1815908, CNS-1717356, CNS-1814590, CNS-1820624, CNS-1826647 and ARO Grant W911NF-18-1-0221.

REFERENCES

[1] "Virtual kidnapping," https://www.fbi.gov/news/stories/virtual-kidnapping.

[2] "Beware of virtual kidnapping scams," https://www.fs.fed.us/inside-fs/beware-virtual-kidnapping-scams.

[3] "Romance scams," https://www.fbi.gov/news/stories/romance-scams.

[4] "Romance scams will cost you," https://www.consumer.ftc.gov/blog/2019/02/romance-scams-will-cost-you.

[5] K. Sobottka and I. Pitas, "A novel method for automatic face segmentation, facial feature extraction and tracking," *Signal processing: Image communication*, vol. 12, no. 3, pp. 263–281, 1998.

[6] B. G. Bhatt and Z. H. Shah, "Face feature extraction techniques: a survey," in *National conference on recent trends in engineering & technology*, vol. 14, 2011.

[7] 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*, 2009, pp. 233–236.

[8] X. Tan, Y. Li, J. Liu, and L. Jiang, "Face liveness detection from a single image with sparse low rank bilinear discriminative model," in *European Conference on Computer Vision*, 2010, pp. 504–517.

[9] M. Sajjad, S. Khan, T. Hussain, K. Muhammad, A. K. Sangaiah, A. Castiglione, C. Esposito, and S. W. Baik, "Cnn-based anti-spoofing two-tier multi-factor authentication system," *Pattern Recognition Letters*, 2018.

[10] Y. Xu, T. Price, J.-M. Frahm, and F. Monrose, "Virtual u: Defeating face liveness detection by building virtual models from your public photos," in *USENIX Security Symposium*, 2016.

[11] G. Kim, S. Eum, J. K. Suhr, D. Kim, K. R. Park, and J. Kim, "Face liveness detection based on texture and frequency analyses," *Sinternational conference on biometrics*, pp. 67–72, 2012.

[12] T. Campbell, C. Williams, O. Ivanova, and B. Garrett, "Could 3d printing change the world? technologies, potential, and implications of additive manufacturing," *Strategic Foresight Report*, 2011.

[13] J. Coopersmith, "Fraud and froth: Free-riding the 3d printing wave," *3D Printing*, pp. 137–172, 2016.

[14] D. Tang, Z. Zhou, Y. Zhang, and K. Zhang, "Face flashing: a secure liveness detection protocol based on light reflections," *arXiv preprint arXiv:1801.01949*, 2018.

[15] W. Wang, S. Stuijk, and G. De Haan, "A novel algorithm for remote photoplethysmography: Spatial subspace rotation," *IEEE transactions on biomedical engineering*, vol. 63, no. 9, pp. 1974–1984, 2016.

[16] G. Pan, L. Sun, Z. Wu, and S. Lao, "Eyeblink-based anti-spoofing in face recognition from a generic webcamera," in *2007 IEEE 11th International Conference on Computer Vision*, 2007, pp. 1–8.

[17] G. Pan, L. Sun, Z. Wu, and Y. Wang, "Monocular camera-based face liveness detection by combining eyeblink and scene context," *Telecommunication Systems*, vol. 47, no. 3-4, pp. 215–225, 2011.

[18] A. Lagorio, M. Tistarelli, M. Cadoni, C. Fookes, and S. Sridharan, "Liveness detection based on 3d face shape analysis," in *2013 International Workshop on Biometrics and Forensics (IWBF)*, 2013, pp. 1–4.

[19] T. Wang, J. Yang, Z. Lei, S. Liao, and S. Z. Li, "Face liveness detection using 3d structure recovered from a single camera," in *2013 International Conference on Biometrics (ICB)*, 2013, pp. 1–6.

[20] J. Booth, A. Roussos, S. Zafeiriou, A. Ponniah, and D. Dunaway, "A 3d morphable model learnt from 10,000 faces," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5543–5552.

[21] P. Huber, G. Hu, R. Tena, P. Mortazavian, P. Koppen, W. J. Christmas, M. Ratsch, and J. Kittler, "A multiresolution 3d morphable face model and fitting framework," in *Proceedings of the 11th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 2016.

[22] "Fake webcam," http://www.fakewebcam.com/.

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

[24] M. M. Chakka, A. Anjos, S. Marcel, R. Tronci, D. Muntoni, G. Fadda, M. Pili, N. Sirena, G. Murgia, M. Ristori *et al.*, "Competition on counter measures to 2-d facial spoofing attacks," in *2011 International Joint Conference on Biometrics (IJCB)*, 2011, pp. 1–6.

[25] A. Benlamoudi, D. Samai, A. Ouafi, A. Taleb-Ahmed, S. E. Bekhouche, and A. Hadid, "Face spoofing detection from single images using active shape models with stasm and lbp," in *Proceeding of the Troisime conference internationale sur la vision artificielle CVA*, vol. 2015, 2015, p. 31.

[26] D. Wen, H. Han, and A. K. Jain, "Face spoof detection with image distortion analysis," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 746–761, 2015.

[27] K. Kollreider, H. Fronthaler, and J. Bigun, "Non-intrusive liveness detection by face images," *Image and Vision Computing*, vol. 27, no. 3, pp. 233–244, 2009.

[28] O. Kähm and N. Damer, "2d face liveness detection: An overview," in *2012 BIOSIG-Proceedings of the International Conference of Biometrics Special Interest Group (BIOSIG)*, 2012, pp. 1–12.

[29] Lenovo, "Veriface," https://veriface.software.informer.com/.

[30] G. Chetty and M. Wagner, "Multi-level liveness verification for face-voice biometric authentication," in *2006 Biometrics Symposium: Special Session on Research at the Biometric Consortium Conference*, 2006, pp. 1–6.

[31] B. Biggio, Z. Akhtar, G. Fumera, G. L. Marcialis, and F. Roli, "Security evaluation of biometric authentication systems under real spoofing attacks," *IET biometrics*, vol. 1, no. 1, pp. 11–24, 2012.

[32] Y. Li, Y. Li, Q. Yan, H. Kong, and R. H. Deng, "Seeing your face is not enough: An inertial sensor-based liveness detection for face authentication," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015, pp. 1558–1569.

[33] J. M. Saragih, "Deformable face alignment via local measurements and global constraints," in *Deformation Models*, 2013, pp. 187–207.

[34] P. P. Chan, W. Liu, D. Chen, D. S. Yeung, F. Zhang, X. Wang, and C.-C. Hsu, "Face liveness detection using a flash against 2d spoofing attack," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 2, pp. 521–534, 2018.

[35] M. E. Celebi and B. Smolka, *Advances in Low-Level Color Image Processing*, 2014.

[36] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1867–1874.

[37] E. Angelopoulou, "The reflectance spectrum of human skin," *Technical Reports (CIS)*, p. 584, 1999.

[38] G. Rodola, "psutil," https://pypi.org/project/psutil/.

[39] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008.

[40] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: Database and results," *Image and vision computing*, vol. 47, pp. 3–18, 2016.

[41] T. M. Pedersen, "The hodrick–prescott filter, the slutzky effect, and the distortionary effect of filters," *Journal of economic dynamics and control*, vol. 25, no. 8, pp. 1081–1101, 2001.

[42] R. Falk, *Understanding probability and statistics: a book of problems*, 1993.