GOTCHA: Real-Time Video Deepfake Detection via Challenge-Response

Govind Mittal, Chinmay Hegde, Nasir Memon
Dept. of Computer Science and Engineering, Tandon School of Engineering, New York University, USA
{mittal, chinmay.h, memon}@nyu.edu

***Abstract*—With the rise of AI-enabled Real-Time Deepfakes (RTDFs), the integrity of online video interactions has become a growing concern. RTDFs have now made it feasible to replace an imposter's face with their victim in live video interactions. Such advancement in deepfakes also coaxes detection to rise to the same standard. However, existing deepfake detection techniques are asynchronous and hence ill-suited for RTDFs. To bridge this gap, we propose a challenge-response approach that establishes authenticity in live settings. We focus on talking-head style video interaction and present a taxonomy of challenges that specifically target inherent limitations of RTDF generation pipelines. We evaluate representative examples from the taxonomy by collecting a unique dataset comprising eight challenges, which consistently and visibly degrades the quality of state-of-the-art deepfake generators. These results are corroborated both by humans and a new automated scoring function, leading to 88.6% and 80.1% AUC, respectively. The findings underscore the promising potential of challenge-response systems for explainable and scalable real-time deepfake detection in practical scenarios. We provide access to data and code at https://github.com/mittalgovind/GOTCHA-Deepfakes.**

1. Introduction

In an increasingly interconnected world, the adoption of live, online video interactions has become widespread. Projections for 2023 indicate that 86% of companies are conducting interviews online, while approximately 83% of employees estimate spending about a third of their working week in virtual meetings [1]. This significant uptick in online interactions creates a fertile ground for novel social engineering attacks. Specifically, high-quality tools capable of creating video deepfakes, which can convincingly mimic a target's facial appearance, are becoming more accessible and can readily circumvent commercial APIs designed for liveness detection and identity verification [2]. While earlier versions of deepfakes primarily targeted public figures [3], recent advancements in deepfake generation technology have made it possible to impersonate ordinary individuals [4], even with a limited set of training images [5], in a real-time setting [6]. These systems, named *Real-Time Deepfakes (RTDFs)*, are a sophisticated variant of deepfakes involving live, interactive video impersonations of a target, posing an urgent, burgeoning threat to the integrity of online human interactions across the globe [7].

RTDFs have already become prevalent to the extent that the FBI has warned of their imminent threat and pervasiveness [8]. Known recent incidents include an individual in China transferring $620,000 to an impersonator of their friend after a video call [4]. Also, an impersonation of the former president of Ukraine joined a Kremlin critic during a recorded video call to lure the critic into performing embarrassing acts [9].

Conventional techniques [10], [11], [12], [13], [14] have considered deepfake detection, but in an offline and non-interactive setting. Despite being technically impressive, such techniques are not explicitly designed for RTDFs and operate under the assumption of *no interaction between an imposter and the detector*. This assumption creates a threat model that favors imposters, who can leverage offline resources to refine their models to target specific individuals. Meanwhile, detection often needs to be performed in real-time without knowledge of the creation process and in potentially constrained environments.

Contrary to this traditional strategy, we explore an alternate approach where the detector can interactively present non-trivial tasks or *challenges* to the imposter. Under this model, the onus of consistently maintaining high-quality deepfakes in real-time, under challenging situations, is now squarely on the imposter. We leverage this asymmetric advantage to design and validate a *challenge-response* approach for identifying RTDFs.

The fundamental conceptual problem is to develop a suite of practical challenges (i.e., the "real user" experience is not significantly altered) yet maximally informative (i.e., "fake user" video outputs are statistically and visually abnormal). In this work, we propose and categorize challenges designed to exploit limitations specific to components of an RTDF generation pipeline, such as a facial landmark detector and auto-encoder. As we propose a variety of challenge categories, the detection accuracy of each challenge may vary depending on the deepfake generation model and the conditions under which the deepfake is enacted, such as ambient lighting or video quality. Therefore, more than a single challenge may be required for accurate detection.

A desirable feature of the proposed challenges is that they induce human-visible artifacts in the responses and provide robust signals for downstream automated machine-learning detectors. This feature facilitates easy audit and explainability in their practical implementations.

We call the proposed technique GOTCHA, echoing the ubiquitous CAPTCHA test used to identify online robots masquerading as humans. At its core, GOTCHA presents one or more challenges to a suspected RTDF, which could require physical action or involve digital manipulations in the video feed. We validate GOTCHA on a novel video dataset of 56,247 videos derived by collecting data from 47 legitimate users. We conduct both human-based and

automated evaluations to test its potential in practical scenarios while establishing a security-usability tradeoff.

Our evaluation of GOTCHA demonstrates consistent and measurable degradation of deepfake quality across users, highlighting its promise for RTDF detection in real-world settings. Our contributions are as follows:

- We explore a challenge-response approach for authenticating live video interactions and develop a taxonomy of challenges by exploiting vulnerabilities in real-time deepfake generation pipelines.
- We collect video data of 47 real users in person, performing eight challenges. The new dataset consists of 56,247 short real and fake videos.
- We perform human and automated evaluations of the approach, demonstrating consistent and visible degradation of deepfake quality caused by challenges. We train a new self-supervised ML-based fidelity scoring model for this purpose. [1]

Authentication and Liveness Detection. It is important to note that a CAPTCHA-like test does not assume the availability of any prior knowledge or identification of the target user. This is fundamentally different from "what-you-know" (e.g. passwords) or "who-you-are" (e.g., biometrics) type of authentication systems. Such systems require the user to enroll prior to authentication and perform it by using an identity-matching algorithm.

In this work, the imposter is explicitly considered to be *present* during an interaction, and all impersonations are performed live (consequently, in real-time). The proposed solution trivially addresses presentation attacks, thus differentiating it from methods explicitly designed for liveness detection.

2. Real-Time Deepfake Generation

Definition. A *deepfake* refers to a digital impersonation in which an *imposter* mimics audio or visual characteristics to convincingly match a specific *target*'s likeness. When such impersonations can be done live[2] with sufficient fidelity, we call them RTDFs.

This work focuses on talking-head videos, which encompass techniques such as facial reenactment or face-swapping. A typical use case for deploying RTDFs are hoax video call interactions.

2.1. Dissecting the Generation Pipeline

Several approaches exist for generating real-time deepfakes [6], [15], [16]. While their details differ, they all utilize the same key components. The following list briefly describes each of these components, as depicted in Fig. 1. We assume that the computing device being used for video conferencing is instrumented to channel the imposter's video stream through a deepfake generation pipeline, and then forward it to the video-conferencing client.

1. **Reproducibility:** All participants consented to release their data for academic research. Hence, we release originally recorded challenges and corresponding deepfakes for non-commercial research, along with the code for the fidelity score model and survey instruments used for human evaluations.

2. Computer graphics literature traditionally considers real-time to be 30 frames/s; however, for video calling, 15 frames/s is often sufficiently "real-time" or live.

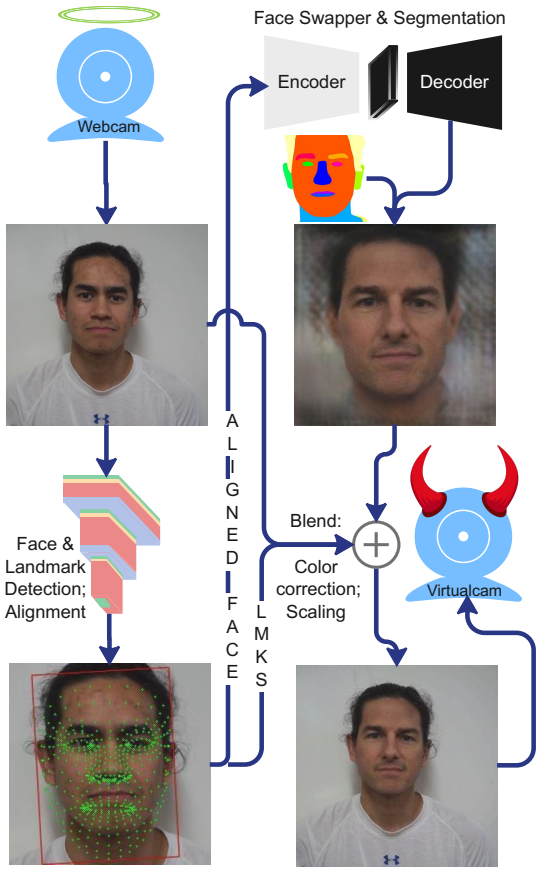


Figure 1. A generic face-swapping RTDF pipeline containing a physical webcam (top), face and landmark detector, face-swapper (auto-encoder), blending operator and a virtual webcam (right). The virtual webcam is piped into a video conferencing software (not shown). Arrows indicate relevant data flows.

- *Face Detector* is a neural network that predicts a bounding box per face in a video frame.
- *Landmark Detection* is a neural network that detects facial key-points called landmarks. Several reeactment techniques use only landmarks as the driving signal for the target's face image. Landmarks also aid in conforming the target's predicted face to fit the imposter's face shape.
- *Face Alignment* module aligns a given input face with respect to the landmarks, which is vital for a robust prediction by the face-swapper. Subsequently, the applied alignment is reversed and re-applied to the predicted face, in order to match back with the imposter.
- *Segmentation* (a) separates the face region into distinct regions of interest (lips, eyes, nose), (b) derives the convex hull of the face that is visible, and (c) determines facial boundaries around occlusions (e.g., hand).
- *Face-Swapper*, typically, involves an autoencoder. The auto-encoder takes the input from the face detector and predicts how the target's face would look under a given set of facial landmarks, occlusions, and lighting.
- *Blending Operator* overlays the inner predicted face of target onto the outer head of imposter. This post-processing step tends to vary across RTDF generation pipelines, involving some combination of blurring, degrading, scaling, compression, and fading.
- *Color Correction*: In case of swapping only the inner

Figure 2. Importance of facial shape similarity. Two imposters with distinct facial shapes result in differing quality outputs, while assuming Ryan Reynolds as target. Qualitative observations imply that better match yields a better fit.

face of the imposter with that of the target, this module can adjust difference in complexion, by sampling color from the outer face region (around forehead and neck) and adjusting the inner swapped face [17].

Note on choice of generative models. Although, this work explores Auto-Encoder-based generative models, but we acknowledge the innovations in synthetic media generation, especially based on NERFs [18], and Diffusion Models [19] coupled with CLIP [20]. However, none of these simultaneously support real-time throughput and high-fidelity deepfake outputs, which are essential ingredients to conceive realistic RTDFs.

2.2. Hurdles to Generating Realistic Deepfakes

Launching a convincing deepfake of a specific target involves overcoming multiple impediments. Below we describe some of the *key impediments we leverage while designing our approach*.

Data Diversity. Creating a persuasive deepfake of a specific individual often requires their diverse facial data. Ideally, this collection should include the target's face captured under various lighting conditions and angles, with or without occlusions, preferably drawn from high-resolution videos. For instance, using a face-swapping framework like DeepFaceLab [6] typically necessitates 4,000 diverse images. In stark contrast, facial reenactment techniques only expect a single image of the target, which is then animated via text [21], speech [22], or facial landmarks [23]. However, these reenactments are far more fragile when confronted with unexpected scenarios. We delve deeper into the implications of this inherent constraint in §6.1.

Face Shape Similarity. Successful impersonation also hinges on the similarity between the facial shapes of an imposter and their target. Any significant mismatch, such as in the spacing between the eyes or the shape of the nose or lips, can result in unrealistic artifacts. These anomalies can manifest as stretching or contraction of the deepfake mask, blending issues, or rigid expressions.

An imposter attempts to project their target's identity, which is often strongly tied to their facial shape and skin color. Hence, having similar face shapes is crucial for a convincing impersonation. However, discrepancies in skin color can often be mitigated using a color transfer technique [24]. Fig. 2 compares two individuals with distinct facial shapes; both face-swapped into the same target identity, indicating that a face shape with a 'closer' match yields a more convincing impersonation.

TABLE 1. INFERENCE SPEED OF AN RTDF. RED AND GREEN COLORS INDICATE PERFORMANCE BELOW AND ABOVE THE REAL-TIME THRESHOLD OF 15 FRAMES/S.

Config. / Speed	α : 8-core i7 CPU	β : α + RTX 3070 (8 GB)	β + WiFi Camera	β + LTE Phone Cam.
Frames/s	1.2	23.5	21.1	20.2

Computational Resources and Real-Time Constraints. Producing high-quality deepfakes is computationally expensive, typically requiring specialized hardware such as GPUs or TPUs. Table 1 showcases the inference speed of DeepFaceLab [6] with various hardware configurations.

The computational load escalates when the deepfake generation has to happen in real time. Envision a scenario where the subject needs to perform an activity such as rapidly moving a hand in front of the face. While human observers might see the motion as a blur, it is still discernible. For deepfakes, handling such high-activity scenarios in real time is challenging. Nimble movements can significantly reduce throughput leading to dropped frames or noticeable lags.

Furthermore, popular cloud service providers such as Google Colab [25] have placed restrictions on training deepfakes on their platform. This constraint adds another layer of complexity on easy access to requisite computational resources, necessitating personal hardware investments to train] high-fidelity deepfakes.

Narrow Portability and Technical Effort: Each new target identity necessitates training a fresh face-swap model from scratch or retraining an existing pre-trained model using the target's data for a substantial duration, taking up to a week.

While there exist target-agnostic, ready-to-use face-swap methods such as FSGAN [15], the resulting deepfakes tend to be more fragile. The fragility is primarily due to the imperfect disentanglement of identity attributes from other aspects like pose and expression. Also, in recent years, the broader community has focused on advancing facial reenactment [16], [26], [27], [28], and such methods trivially degrade under our approach (see Fig 4).

Hence, an imposter aiming to navigate around this impediment has to enhance the existing face-swapping methods or build a custom one from scratch. Therefore, possessing (a) substantial software development proficiency, (b) an understanding of underlying principles, and (c) the potential to conduct extensive trial-and-error iterations to achieve a viable, effective solution.

Thus, an imposter attempting to successfully deepfake a target needs to manage all the constraints listed above and possibly more.

3. Problem Description

Given the design of an RTDF generator and the aspects an imposter needs to consider for using them in practice, we describe our threat model, setup and hypothesis.

Threat Model involves three subjects – an imposter, a target, and a defender. The defender is an individual on a video call, intending to interact with the target. The imposter is another individual on the same call, seeking to deceive the defender by driving an AI-generated talking head portrayed as the target.

Practical examples of this threat model exists in online *Know-Your-Customer (KYC) verification process*, crucial for companies like Airbnb [29] and Uber [30], as well as US government agencies utilizing ID.me [31], involves capturing a selfie with a photo ID. RTDFs threaten these face-matching systems by altering faces in both the ID and live video, bypassing liveness detection mechanisms as reported by Sensity [2].

Additionally, online exams, such as those administered by ETS [32], and interviews are vulnerable against RTDFs. This technology enables impostors to fraudulently take tests or interviews with a fake identity or on behalf of the intended person. Such a tactic comes close to North Korean IT workers, who secured remote employment under fake identities [33].

Setup. When the defender and the potential imposter join an online video call, the defender provides instructions to the potential imposter and requests that they respond to one or more specific actions, which we will call challenges. After receiving the response(s), the defender accepts or rejects the portrayed identity.

Defenders. The defender assumes the existence of generic deepfake generation components without knowledge of any exact specification. The defender does not assume any trust in the imposter or their devices and does not keep the nature of the requested challenges secret. Also, no identifiable information, such as biometrics or face, is collected through an extensive enrollment process. Here, we assume a stronger threat model, as having such information can help the defender.

Imposters. An imposter can train and use deepfake models and have enough computational capacity to perform a successful live impersonation. The imposter is also capable of understanding, interacting, and responding to requests made to them by the defender. The imposter could be aware of the nature of the requested tasks and can deploy adaptive countermeasures.

Initially, we consider a scenario where the target individual has a limited online footprint, leading to a constrained range of available high-quality facial data. Such cases confine an imposter to using images from specific sources like social media or recorded webinars. However, we later broadened our scope to include more data-diverse contexts featuring potential accomplices or public figures as targets (§6.2). In such scenarios, the imposter can access a *rich* set of data, either willingly or inadvertently provided by the target. Access to such a dataset can improve the realism of RTDFs, making them harder to detect. This extension is particularly relevant in contexts such as job interviews, where the target might be an accomplice of the imposter.

Hypothesis. *We posit that a user can perform the specific tasks required by the challenge during a live interaction, which is difficult for a deepfake generation pipeline to model in real-time.*

Figure 3. A method to guide a user and randomize the challenge is illustrated. The user follows on-screen instructions to mimic the actions of an avatar, performing required head movements.

Each component within such a pipeline has optimal operating conditions, defined by the type of component, its architecture, and the biases learned during its training phase. As a result, an imposter can successfully generate a convincing deepfake only when the live inputs fall within the overlapping range of these optimal conditions. Any divergence from this range can compromise the performance of one or more components in the pipeline, leading to discernible artifacts in the final deepfake video. Grounded on our hypothesis and this insight, we design a series of tasks that impair the performance of the RTDF pipeline.

4. Challenges

Definition. A challenge $\mathcal{C}$ is a task that has the following properties:

- degrades real-time deepfakes (optionally visibly),
- is can be performed during a live call, and
- can be specified by a set of randomizable parameters.

In order to verify a potential imposter, a defender picks a random challenge (and its specification), and conveys it to the potential imposter to perform. For example, if the challenge picked is head rotation, then a concrete specification could be 'rotate the head to the left by an angle of $45°$ and downwards by $30°$ for a specified time,' creating a unique challenge. In order to enhance usability, a defender can demonstrate the challenge themselves or by using an automated avatar as shown in Fig. 3.

Since current RTDF pipelines essentially replace the imposter's face with that of a target while preserving the remainder of the scene, hence in this work, we only consider challenges that necessitate non-trivial user actions involving faces, such as performing a specific sequence of head movements, occlusions, face deformations, etc.

4.1. A Taxonomy of Facial Challenges

This subsection introduces a taxonomy of challenge categories considered in this work. Challenges are tailored to impact facial fidelity and are classified based on specific facial transformations they address. Table 2 summarizes the taxonomy and symbolizes the components that get disrupted due to each challenge along with usability characteristics. The categories in the taxonomy are as follows:

- **Head Movement** encompasses challenges that involve physical motions of the head or the entire body, which can modify the point-of-view of the facial appearance. Examples of head movement-based challenges include:
  - *Directed Head Movement*: A user moves their head in specific directions - for example, nodding, shaking, or

TABLE 2. TAXONOMY OF CHALLENGE CATEGORIES CONSIDERED. USABILITY BENEFITS ARE DESCRIBED IN §A.5.

Category	Examples	Component Effected								Usability				
		Face Detector	*Landmark Detection*	*Face Alignment*	*Segmentation*	*Auto-encoder*	*Blending*	*Color Correction*	*Inference Speed*	*Easy-to-Comprehend*	*Appropriate-to-Request*	*Physically-Effortless*	*No-Equipment-Needed*	*Detected-by-Humans*
Face Occlusion	Hand	●	○	○	●	○	●			●	●	○	●	●
	Object	●		○	●	○	●	○		●	○	○		●
Head Movement	Directed Movement	○		○	●		●			●	●	○	●	●
	Whole Body Movement	●		○	●		○			●	○		●	●
Facial Deformation	Manual Deformation		●		●	○			○	●		○	●	●
	Expression Alteration					●			○	●	○	○	●	●
Face Illumination	User-Guided					○	●	●	○	●	○			○
	Display-Induced					○	●	●	○	○	○	●	●	

Benefits: ● = Fully Offered; ○ = Quasi/Partially Offered ; **Empty** = Not Offered ;

tilting the head to various angles. These movements can complicate the task for RTDF generators, which typically see more stable and frontal facial views during training.

  - *Whole Body Movement*: More drastic movements, such as turning around, walking, or standing up, could introduce complex temporal and spatial distortions, altering the facial perspective and distance viewed by the camera.

- **Face Occlusion**. This category includes challenges involving reduced face visibility due to an occluding object. Ways to occlude a face include:
  - *Hand*: A user can partially occlude their face using their hands, to block full facial visibility.
  - *Objects*: External objects can also be used for occlusion. Readily available items, such as masks, sunglasses, and tissues, work for this purpose.
- **Facial Deformations** comprise challenges designed to induce modifications to the facial structure or presentation. These challenges could be:
  - *Manual Deformations*: The user manipulates their facial features using their hands. For example, users press a finger against their cheek or forehead or stretch their lips, creating deformations.
  - *Expression Alterations*: These challenges consist of users altering their facial expressions dramatically. Examples include wide grins, deep-set frowns, elevated eyebrows, and squinting eyes.
- **Face Illumination** includes challenges that alter the illumination on the user's face. Examples include:
  - *User-Guided Illumination*: Users change their position relative to light sources, adjust the light source itself, or use handheld devices like a flashlight or camera flash to introduce diverse lighting conditions, producing shadows and highlights. For example, having light illuminate just one side of the face can cause difficulty replicating it accurately.
  - *Display Induced Illumination*: Face illumination can also be affected by causing the imposter's display to emit sudden flashes or structured light patterns to reflect from the user's face.

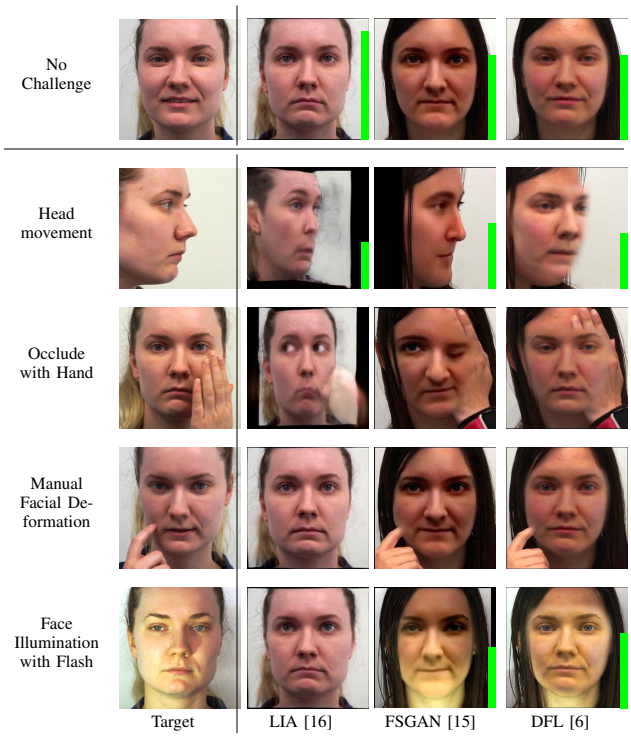


Figure 4. Challenge frame of original and deepfake videos. Each row aligns outputs against the same instance of challenge, while each column aligns the same deepfake method. The green bars are a metaphor for the fidelity score, with taller bars suggesting higher fidelity. Missing bars imply the specific deepfake failed to do that specific challenge. Video version at http://govindm.me/gotcha-figures.

4.2. Dataset Collection and Curation

From the challenge categories described above, we selected eight instances for data collection and further evaluations. Table 3 lists the selected challenges, which span the four identified categories.

Original Videos: Building a Unique Dataset. In the absence of publicly available deepfake datasets incorporating challenge-response mechanisms, we created a novel dataset tailored for this study. The dataset features a group

TABLE 3. COUNTS OF CHALLENGE VIDEOS.

Challenge \ Generator	Original	LIA	FSGAN	DFL
No Challenge	47	2,162	2,103	2,162
Head Movement	46	2,068	1,953	2,115
Occlude w/hand	45	2,068	1,954	2,115
Occlude w/Sunglass	45	2,068	1,952	2,115
Occlude w/Facemask	44	2,021	1,916	2,068
Manual Deform	44	2,021	1,907	2,068
Protrude Tongue	45	2,068	1,944	2,115
Alter Expression	46	2,115	1,998	2,162
Flash	47	2,162	2,088	2,162
Total	**409**	**18,847**	**17,815**	**19,176**

of 47 participants, encompassing various demographics (refer to Fig. A.1). Each participant was recorded performing specified challenges in a talking-head video-style, with only one individual appearing per video to simplify the evaluation process. We outline the step-by-step instructions provided to the participants while recording in §A.1.

The dataset comprises of manually recorded Full HD videos. This results in an expansive archive of almost 50 GB of 409 original video data. Table 3 provides a breakdown of individual original challenges. Numbers < 47 indicate error in recording.

Each participant contributed around 5 to 6 minutes of total video footage. Since this work focuses primarily on headshots, we isolated this area from the full-view footage for each participant. The average dimensions of the headshots—which include the head, neck, upper shoulders, and some margin—were approximately 500×570 pixels, with minor variations across participants. For uniformity across evaluation, we extracted these headshots and resampled them to a standard resolution of 512×512.

Deepfake Videos. With the original video dataset in place, we proceeded to generate deepfake counterparts using three RTDF pipelines as follows:

- *LIA (Latent Image Animator)*: This pipeline is a facial reenactment method outlined in [16]. Given its target-agnostic nature—meaning it does not require specific target data during inference—we employed a pre-trained model for our experiments.
- *FSGAN (Face Swapping Generative Adversarial Network)*: This corresponds to the second version of FSGAN [15]. Similar to LIA, this model is also target-agnostic, hence we utilized a pre-trained model made available by the authors for our study.
- *DFL (DeepFaceLab)*: Notorious for generating hyper-realistic deepfakes [6], this pipeline serves as a baseline for in-the-wild deepfake videos. For our study, we trained *individual DFL deepfake generators for each participant* using their 'no challenge' videos. These videos records the participants in a range of frontal angles while they sit naturally, aiming to mimic the kind of data readily accessible online for non-celebrity individuals. Training continued until convergence for approximately around 300,000 iterations.

Inference Procedure for Deepfake Video Evaluation. To build a comprehensive set of deepfake videos containing challenges, each of the 47 participants was designated as the target. In contrast, the remaining participants acted as imposters. This configuration yielded a total of $47 \times 46 = 2,162$ unique imposter-target pairs. We repeat this procedure across all three RTDF pipelines and create a rich evaluation set for every challenge (for computing details, refer to Section §A.2). Table 3 specifies the counts of videos generated per challenge and pipeline, totalling 500 GB of deepfake videos. Some counts are lower than 2,162 since some imposter-target pairs were missing or resulted in erroneous deepfake videos.

Ethics. All participants signed consent forms that permit us to release their data to accredited academic institutions for non-commercial research purposes. Participants retain the option to withdraw their consent at any time. To mitigate privacy risks, we prohibit the unauthorized distribution and public display of participants' faces. Our institutional review board oversaw these provisions under IRB-FY2022-6482 and formalized them in a data release form, enforcing compliance in future research.

5. Evaluation

To comprehensively gauge the effectiveness of GOTCHA for RTDF detection, we used both human assessments and automated scoring using ML algorithms. Below, we provide details of each evaluation strategy.

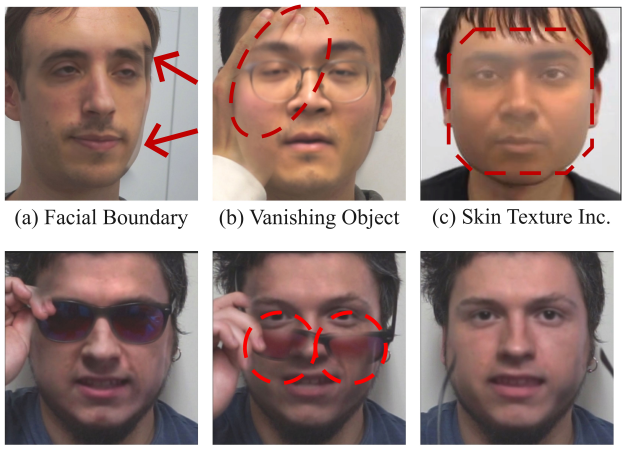
5.1. Human Evaluation

A key characteristic of the proposed challenge-response approach is that the degradations caused by the challenges are easily perceptible to the human eye, and hence, they enable explainable evaluations. To corroborate this statement, we conducted a human study. We investigated the question: ***Q1.*** *Can humans detect deepfakes and can challenges enhance this ability?* and to ensure that the responses were not spurious, we check whether ***Q2.*** *Can humans identify deepfake artifacts successfully?* A preview of the experiment is available at https://app.gorilla.sc/openmaterials/693684.

Dataset subselection: As evaluating the full dataset is unnecessary, we select a representative subset of 150 videos, each of 10 ± 4 seconds duration, as follows:

- *Challenge Category:* We selected a single instance for each challenge category, i.e., face occlusion, head movement, facial deformation, and face illumination, along with 'no challenge' samples.
- *Face Identity:* We manually selected ten imposter-target pairs per challenge. In order to make human evaluation more challenging, we limit to imposter-target pairs with matching sex, complexion, and facial structure, thereby eliminating trivial inconsistencies.
- *RTDF:* We consider Original, FSGAN [15] and DFL [6] samples for evaluation due to their lower artifact rates. LIA [16] is excluded as it fails to perform most challenges (except head movement), making it trivially detectable (see Fig. 4).

Screening: We began by familiarizing each participant with different types of deepfake artifacts they might encounter. Instructions included definitions and example videos of 'no artifact' and four suggestive artifacts – Facial Boundary Artifacts, Vanishing Object, Skin Texture



(a) Facial Boundary (b) Vanishing Object (c) Skin Texture Inc.

(d) Temporal Inconsistency: the sunglass partially vanishes in between

Figure 5. Artifacts defined for human evaluation. (a) has a boundary artifact near the right brow, in (b) the hand vanishes behind the face, (c) has hazy face detail, and (d) the sunglass starts vanishing briefly. Red objects indicate artifact locations. Video version at http://govindm.me/gotcha-figures/.

Inconsistency, and Temporal Inconsistency. For examples see Fig. 5 and for definitions see §A.4, or

After instructions, we tested each participant by asking them to select the most apparent artifact present in 10 previously unseen videos. Out of a pool of 60 recruited participants (males = females = 30), 43 participants qualified passing the test and proceeded to the main task.

Main Task. We tasked each qualifying participant to evaluate 45 videos randomly sampled from the subset of 150 videos. The experiment resulted in an average of 12.9 evaluations per video sample and accumulated 129 evaluations per challenge per RTDF. Each evaluation comprised of four responses (see their screenshots in §A.4):

- *Compliance to challenge* ($\mathcal{C}$): A binary choice determining *whether the subject in the video executed the action specified by a text description of the challenge*.
- *Realism of the video* ($\mathcal{R}$): A slider ranging from 0 to 100% in steps of 5%, captured *how fake the video appears*. Higher scores indicate a less convincing video.
- *Identification of artifacts*: A multi-selection question asked participants to identify *which of the artifacts listed above were present in the video*.
- *Localization of artifacts*: A grid with 18 cells was overlayed over the face and neck regions, and participants needed to select *the locations of visible artifacts*.

Metric and Visualizations. Using the responses to Compliance $\mathcal{C}$ and Realism $\mathcal{R}$, we define the *mean human-score degradation* $\mathcal{H}$ as:

$$\mathcal{H}_{\mathcal{P}}(c) = \frac{1}{n}\sum_{i=1}^{n} \max\left(1-\mathcal{C}(V^i_{\mathcal{P}}(c)), 1-\mathcal{R}(V^i_{\mathcal{P}}(c))\right),$$

where $V(c)$ is a video of challenge c either original or generating using RTDF pipeline $\mathcal{P}$, and n is the response count. This metric assigns a 100% degradation value to a video that fails to comply with the given challenge or a lower score based on its realism value.

Fig. 6 presents the human performance in detecting deepfakes using the above metric. The top part shows

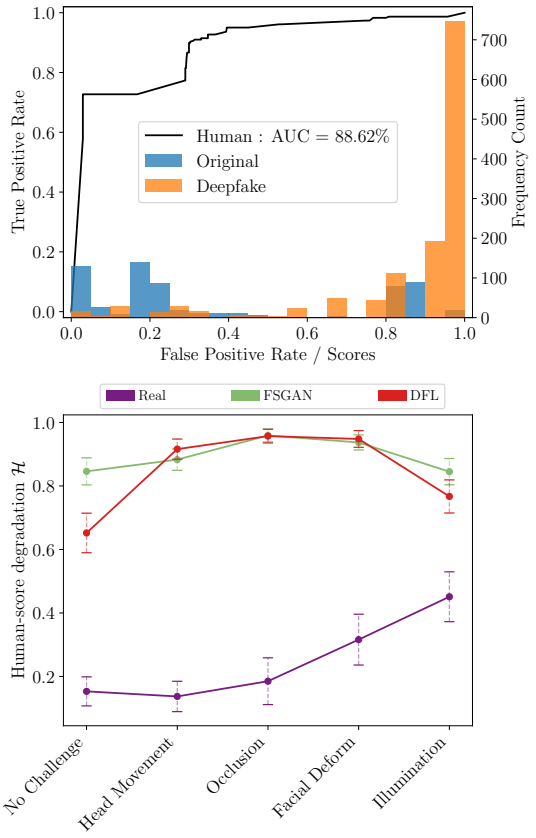


Figure 6. **(Top)** ROC curve of human-score degradation for original and deepfake videos along with histogram of both categories spread across the scores. (**Bottom**) Boxplot of mean degradation scores with 95% confidence interval against the considered challenge categories.

an ROC curve with the frequency count of responses attributed to original and deepfake videos. The bottom part contains mean human scores $\mathcal{H}$ for original videos, DFL, and FSGAN across challenge classifications, averaged with $n = 129$ responses.

Fig. 7 illustrates the distribution of artifacts (including 'no artifact') tagged to a challenge category. Fig. 8 overlays a saliency map-like distribution on a representative challenge image to indicate the locations of detected artifacts. We reiterate that we provided each evaluator with bounding boxes spread across the neck and face region (see Fig. A.4 d). We added, normalized, and smoothened all responses for videos of a challenge and RTDF to produce the illustrated distribution.

Inferences. *Challenges help humans to detect RTDFs* Humans consistently scored RTDFs to have higher degradation compared to original videos with AUC of 88.6% resulting in an accuracy of 81.2% (threshold = 37%). The introduction of a challenge caused a 25% rise in DFL's degradation score and a 12% increase in original videos. In contrast, FSGAN's score rose by only 6%, likely due to its already high baseline scores of 'no challenge' videos. This low gain is attributable to textural artifacts in the target-agnostic FSGAN videos, even without challenges. In contrast, DFL videos appeared more natural as we trained them with target data.

Interestingly, challenges involving illumination intro-

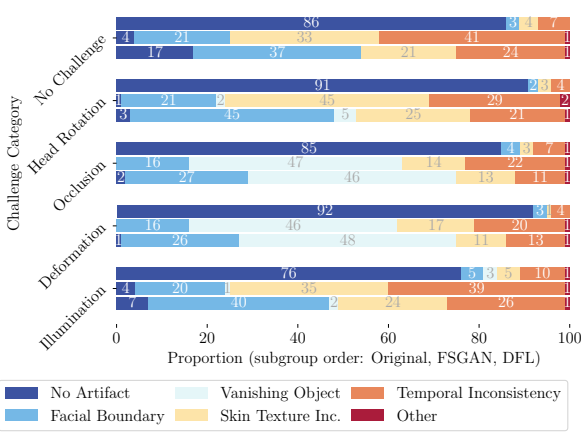


Figure 7. Proportion of artifacts present in each challenge per generator, tagged by human-evaluators. Each groups of three is in the order - Original, FSGAN, DFL.

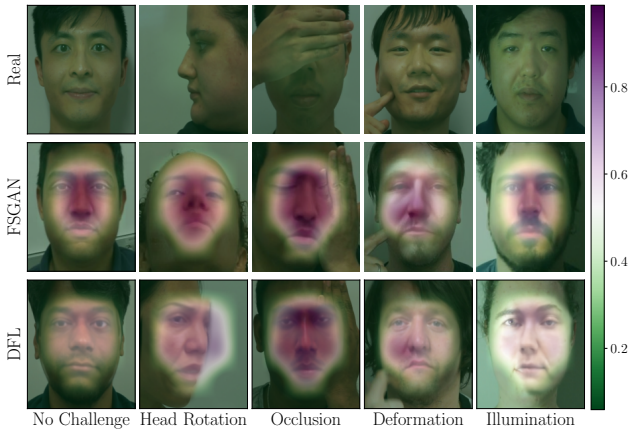


Figure 8. Distribution of artifact locations selected by human-evaluators. The background images are representative examples of each challenge.

duced human error, causing evaluators to misidentify some original videos to be RTDFs and vice versa. This error in judgment was likely caused by confusion in discerning lighting changes when a flash was turned on-and-off.

Humans were effective in identifying RTDF artifacts. Observing Fig. 7, human evaluators consistently labeled original videos as having 'no artifact' while correctly identifying at least one type of artifact in RTDF videos with an accuracy of 92.7%. Specifically, for challenges involving occlusion and facial deformation, the evaluators accurately tagged the RTDF videos as containing a 'vanishing object' 46.8% of the time. This accuracy rate increased to 67.8% when we also considered the signature artifacts specific to each RTDF generator, namely, 'skin texture inconsistency' for FSGAN and 'facial boundary artifacts' for DFL. When combining the above observations, the overall human accuracy for artifact identification reached 80.2%.

Observing Fig. 8, original videos get attributed with fainter distributions, suggesting they have fewer artifacts, while RTDF videos indicate a concentration of more pronounced artifacts. This pattern is consistent across both FSGAN and DFL. As more noticeable artifacts are spread specifically in the inner face region, they suggest that human evaluators are also localizing these imperfections.

Takeaway. Our human-study yields affirmative answers to the questions posed. While the efficacy of challenges in improving detection could be contingent on the quality of the generator output, they undeniably make artifacts more visible, thereby aiding human evaluators in assessing the authenticity of video calls.

5.2. Automated Evaluation

A natural question arises whether RTDFs can be detected in a scalable way. To this end, we evaluated traditional deepfake detection techniques and go on to develop our own in-house detector.

In order to automatically differentiate deepfakes from original videos, we need a fidelity score model along detector to infer compliance. The primary task of a fidelity score model, denoted as f, would be to measure the difference—or degradation—in the quality of an impersonation generated by an imposter with respect to the genuine target, particularly during a challenge scenario. Complementary to fidelity, compliance detection, denoted as $\mathcal{C}$, would inform whether a video contains the corresponding challenge.

Let $V_{\mathcal{P}}(imp, tg, c)$ represent a deepfake video wherein an imposter imp employs an RTDF pipeline $\mathcal{P}$ to impersonate a target tg while undergoing challenge c. Using the same notation, genuine original video are represented by $V_{\text{orig}}(\phi, tg, c)$. We compute compliance $\mathcal{C}$ and fidelity loss $\Delta\mathcal{F}$ and combine them into the **normalized machine-scored degradation** $\mathcal{M}$, an aggregate measure of this quality difference:

$$\text{fake}_i(\mathcal{P}, c, imp, tg) = V^i_{\mathcal{P}}(imp, tg, c)$$
$$\text{orig}_i(c, tg) = V^i_{\text{orig}}(\phi, tg, c)$$
$$\Delta\mathcal{F}(\text{fake}_i, \text{orig}_i) = \frac{w}{n}\sum_{i=1}^{n/w}\frac{f(\text{fake}_i) - f(\text{orig}_i)}{f(\text{orig}_i)}$$
$$\mathcal{C}(\text{fake}_i, \text{orig}_i) = \begin{cases} 1 & \text{if } \mathcal{C}_c(\text{fake}_i, \text{orig}_i) \text{ is True} \\ 0 & \text{otherwise} \end{cases}$$
$$\mathcal{M}_{\mathcal{P}}(c) = \sum_{\forall imp, tg} \max(\Delta\mathcal{F}(\text{fake}_i, \text{orig}_i), 1 - \mathcal{C}(\text{fake}_i, \text{orig}_i)).$$

Here, n stands for the total number of frames in the video, V^i is the i^{th} fragment of the video with w frames, and $\mathcal{C}_c$ is a challenge-specific compliance detector.

Realizing a Fidelity Score Model. Initially, our evaluation incorporated well-established offline deepfake detection algorithms, aiming to adapt them as the fidelity score f. Specifically, we considered Self-Blending Images (SBI) [34] and Fully Temporal Convolution Network (FTCN) [35] as candidates, due to their promising achievements on standard deepfake datasets such as Celeb-DF [36], DFDC [37], and FFIW [38]. However, they under-performed on a set of 620 original and 25,400 deepfake videos derived from our dataset (see Fig. 9). We attribute the low performance of previous methods to the introduction of novel artifacts by the challenges, which are largely absent in their respective training corpora. Hence, we created a customized model to score the fidelity of the videos more reliably.

TABLE 4. COMPLIANCE $\mathcal{C}$ AND FIDELITY LOSS $\Delta\mathcal{F}$ ALONG WITH MACHINE-SCORED DEGRADATION $\mathcal{M}_{\mathcal{P}}(c) = \max(\Delta\mathcal{F}, 1-\mathcal{C})$ IN %. STANDARD DEVIATIONS ARE INDICATED AS SUBSCRIPTS OF MEANS. ARROWS SIGNIFY DIRECTION OF INCREASE OF FAILURE RATE IN DEEPFAKES. THE CHALLENGES ARE SORTED IN ASCENDING ORDER OF THEIR EFFICACY.

RTDF $\mathcal{P}$ / **Challenge c**	**Compliance Detection Strategy**	**DFL**			**FSGAN**			**LIA**			**Chal. Avg.**
		$\mathcal{C}\downarrow$	$\Delta\mathcal{F}\uparrow$	$\mathcal{M}\uparrow$	$\mathcal{C}\downarrow$	$\Delta\mathcal{F}\uparrow$	$\mathcal{M}\uparrow$	$\mathcal{C}\downarrow$	$\Delta\mathcal{F}\uparrow$	$\mathcal{M}\uparrow$	
No Challenge	-	100.0	$10.3_{\pm 7.3}$	$10.3_{\pm 7.3}$	100.0	$12.1_{\pm 6.9}$	$12.1_{\pm 6.9}$	100.0	$16.6_{\pm 5.6}$	$16.6_{\pm 5.6}$	13.0
Head Movement	Change in Yaw and Pitch	97.6	$17.0_{\pm 3.5}$	$18.4_{\pm 11.1}$	98.1	$16.8_{\pm 4.0}$	$18.2_{\pm 11.4}$	72.8	$23.6_{\pm 5.9}$	$43.0_{\pm 33.8}$	26.5
Illuminate w/ Flash	Peaks in Face Intensity	51.6	$19.7_{\pm 10.5}$	$58.8_{\pm 40.8}$	83.3	$17.7_{\pm 11.0}$	$31.2_{\pm 32.5}$	51.6	$17.1_{\pm 9.7}$	$57.2_{\pm 42.1}$	49.1
Alter Expression*	Expression Recognition	30.1	$14.7_{\pm 9.4}$	$65.9_{\pm 42.2}$	35.8	$13.9_{\pm 7.3}$	$57.8_{\pm 43.2}$	34.9	$16.4_{\pm 9.2}$	$60.2_{\pm 42.0}$	61.3
Occlude w/ Sunglasses*	Face Segmentation (change in face coverage)	51.8	$11.6_{\pm 5.0}$	$53.3_{\pm 44.2}$	7.2	$10.3_{\pm 4.4}$	$93.6_{\pm 23.1}$	51.8	$14.3_{\pm 7.3}$	$56.2_{\pm 42.7}$	67.7
Protrude Tongue*	Object Detector	20.3	$15.5_{\pm 15.3}$	$81.5_{\pm 34.4}$	19.9	$16.1_{\pm 13.7}$	$81.7_{\pm 35.4}$	20.3	$17.9_{\pm 12.6}$	$82.7_{\pm 33.2}$	82.0
Occlude w/ Facemask*	Face Segmentation (change in face coverage)	22.0	$33.0_{\pm 15.2}$	$81.1_{\pm 34.1}$	21.7	$18.4_{\pm 8.5}$	$83.2_{\pm 32.0}$	22.0	$39.4_{\pm 18.7}$	$83.5_{\pm 31.5}$	82.6
Manual Deform	Finger-on-Face Detector	20.3	$24.0_{\pm 21.2}$	$82.7_{\pm 32.1}$	19.8	$23.9_{\pm 20.8}$	$82.9_{\pm 33.3}$	20.3	$21.3_{\pm 16.9}$	$82.5_{\pm 33.9}$	82.7
Occlude w/Hand	Face Segmentation (change in face coverage)	11.2	$12.4_{\pm 5.5}$	$88.6_{\pm 29.4}$	6.2	$12.6_{\pm 4.7}$	$94.6_{\pm 21.0}$	11.2	$19.9_{\pm 9.3}$	$91.0_{\pm 25.4}$	91.4
RTDF avg. (chal only)	-	38.1	18.5	66.3	36.5	16.2	67.9	35.6	21.2	69.5	-

*unseen challenge during training

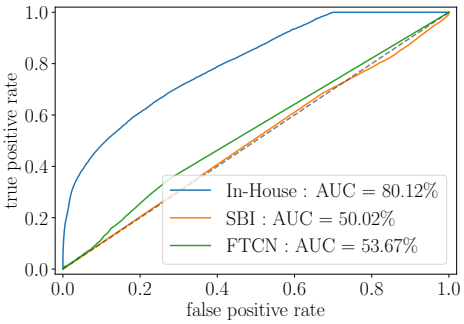


Figure 9. ROC curves for deepfake detectors, SBI [34] and FTCN [35], and our in-house fidelity model. The Area-Under-Curve (AUC) measures the degree of separation between original and deepfake video scores; a higher AUC implies higher discrimination, thus better performance.

Like prior works, we use an ML-based approach utilizing a 3D convolutional neural network as backbone, specifically, a 3D-ResNet18 [39]. 3D-ResNets have been used for predominantly for action recognition in videos and can be used to model temporal information, making them suitable for tasks that require understanding the temporal dynamics in videos. This model quantifies the loss in fidelity induced by each challenge while being trained only on a subset of our dataset and outperforms others (see Fig. 9). The training subset contains the following subselections:

- Original and deepfake videos with four challenges, one from each category and 'no challenge' (the first five challenges listed in Table 4),
- 32 frames per sample,
- 35 out of 47 target identities, and
- Deepfake videos created using only DFL.

The training objective was to optimize a contrastive loss that draws the embeddings of original samples closer together during training while pushing deepfake video embeddings further apart [40]. Using f, we compute the deviation in fidelity $\Delta\mathcal{F}$ of a deepfake video from its ground truth.

Compliance Detectors. While the fidelity score f can measure video artifacts, it does not account for cases where an imposter neglects the request to execute the challenge or the deepfake generators silently fail to replicate the intended action. There are minimal artifacts in such cases despite being a fake video. Hence, we verify compliance to each challenge $\mathcal{C}$ to account for them.

As each challenge c relies on a different signal for verifying compliance, we deploy a set of challenge-specific compliance detectors $\mathcal{C}_c$, based on the following strategies:

- *Occlusion:* We use MediaPipe [41] to segment any visible part of the face and compute face coverage (ratio of visible face to the whole frame). For hand occlusion scenarios, where participants place their hands on their faces four times while facing the camera, we fit a sinusoid to the face coverage data and consider the video complied if the sinusoid has significant amplitude.

  Also, we consider the video compliant for facemasks and sunglass occlusions if the mean face coverage difference from the exact original video deviates by less than 1%.
- *Head Movements:* We predict yaw and pitch for each frame in the video [42]. The video is compliant if the pitch and yaw values indicate significant head movement (> 10 or < -10) in all four directions.
- *Facial Deformations:* We address challenges involving manual deformation, such as poking the cheek with a finger or protruding the tongue using objects and face detectors. We use the detectors to identify and locate the position of respective objects and grant compliance if these objects are detected and their position intersects with the correct facial region (right/left face for the finger, the lower face for the tongue).
- *Expressions:* Facial expressions are predicted using an EfficientNet model trained on the AffectNet

dataset [43], followed by computing Shannon's entropy. A video complies with altering facial expressions if the difference in facial expression entropy between the ground truth and the deepfake video is less than 0.20.

- *Illumination:* Face intensity changes (in grayscale) are computed across the video, focusing on the presence of sharp peaks. The video is compliant if the difference in the number of such peaks between deepfake and original videos is less than 10%.

The final machine-scored degradation derived from fidelity and compliance achieves an AUC of 80.1%. Refer to §A.3 for training and architecture details. Note that we do not assert that this model is the new SoTA. We rather imply that deepfake detection is a simple problem to pose but a complex problem to solve in practice.

We used full videos during scoring, each comprising hundreds of frames per sample. Table 4 describes the normalized machine-scored degradation score of each evaluated RTDF generator *relative to their original counterparts*. Although we split the seen identities during the training of fidelity score to follow best practices for evaluating ML models, we show results on all identities.

Observations. The 'no challenge' condition typically resulted in lower scores than most challenge scenarios, with them scoring 100% compliance due to cooperating participants. Among the evaluated RTDF pipelines, the facial-reenactment-based LIA fared the worst across all challenges. DFL and FSGAN came close while struggling against occlusion challenges, likely due to their models' lack of segmentation capabilities.

The observation that the fidelity score function captured the fidelity loss across all challenges, despite some not seen during training (asterisk marked in Table 4), suggests that it has inherent generalizable capabilities.

Compliance and realism provide complementary signals, with compliance playing a significant role especially in cases of occlusions and manual deformations. On average, failure to comply to a challenge contributes three times the degradation compared to loss in fidelity.

Takeaway. We infer that a machine-assisted scoring method can aid in detecting RTDFs by measuring their deviation from pre-determined original videos, with the effect of challenges disproportionately impacting deepfake videos more.

Summary. Human and Machine evaluations showcase the effectiveness of GOTCHA in drawing out the inherent weaknesses in real-time deepfake generation in an interpretable and scalable way. Interestingly, both evaluations have a perfect alignment in their ordering of challenge categories efficacy:

No challenge $<$ Face Illumination $<$ Head Movement $<$ Facial Deformation $<$ Occlusions.

The prescribed challenges offer reliable protection against deepfakes with a mere 10 seconds of engagement. Moreover, they support both human-interpretative explanations and scalable algorithmic evaluations.

6. Defenses against Real-Time Deepfakes

Our results indicate that GOTCHA can be an efficient tool to counter the threat of RTDFs. When effectively implemented, a single challenge can unmask a deepfake within a brief time-frame of 15 seconds.

However, this effectiveness presupposes a naïve threat. In a more realistic scenario, a savvy imposter could be aware of the nature of the proposed challenges and adapt accordingly. For example, they could curate pre-fabricated responses or advance the technological capabilities of deepfake algorithms to evade detection.

On the other hand, defenders have limited situational awareness, i.e., they are versed with the general attributes of deepfake generation but are blind to specifics about the person they are striving to authenticate. Thus, they can rely only on the inherent limitations of the generation process and make educated assumptions about the data an imposter could potentially access.

In the rest of this section we discuss the inherent limitations of the imposter, and the countermeasures they can employ to evade detection. Armed with the limited resources at our disposal, we outline simple but effective mitigation strategies a defender can employ to confound adaptive imposters. These include randomising each challenge instance and deploying a sequence of challenges. However, it is essential to acknowledge that such mitigation adds complexity to the authentication process, which may affect user experience. We explore these tradeoffs in subsequent discussions.

6.1. Limitation of Imposters

In this subsection, we discuss inherent limitations of the deepfake generators. First we look at the typical architecture of an RTDF and examine the limitation of the essential components of a standard design. Second, we look at the effect of the availability of data on the intended target on the quality of the impersonation. Both these factors become important tools for the design of challenges to help expose RTDFs even against a resourceful and adaptive adversary.

Inherent limitations of RTDF generation pipeline Traditional literature considers RTDF pipelines as monolithic entities; however, we examined how to degrade each component individually. The following description provides insights into the vulnerabilities of typical components found in a generic deepfake generator (refer to §2.1). The examples provided in this subsection are based on experimentation done on DeepFaceLab [6].

Face Detection, Landmark Detection, Face Alignment. These components are quite robust due to their ability to perform effectively on diverse inputs (including faces in profile, distorted faces, and facial expressions). This robustness is primarily due to the abundance of high-quality face datasets available for training, and the maturity of computer vision techniques.

We find that occluding eyes significantly degrades the performance of these modules, as this region is a critical feature for these components (see Fig. 10 b). Additionally, since face landmark detection only provides limited facial structural information, any significant facial deformation or depression can cause noticeable anomalies. E.g., even with MediaPipe [41], which provides 468 landmarks for a full frontal face, RTDFs are unable to accurately capture significant facial depressions (Fig. 10 d).

(a) Facemask (b) Occlusion (c) Look up

(d) Deformation (e) Expressions (f) Illumination

Figure 10. This figure showcases a range of tasks more expansive than those displayed in Fig. 4. The top row presents the ground truth target images, whereas the bottom row contains the corresponding deepfake predictions of target images.

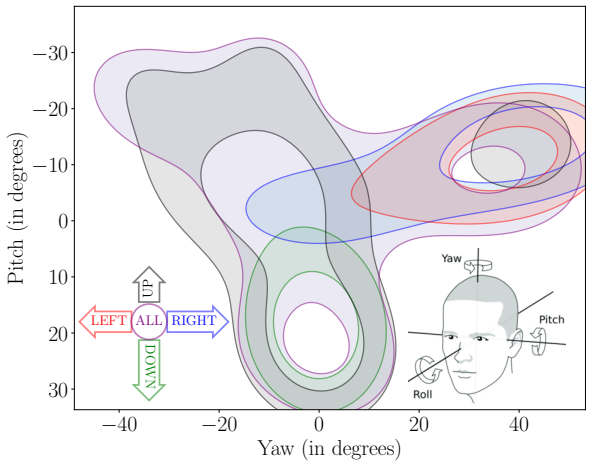


Figure 11. Comparative Analysis of Model Performance Based on Data Diversity: face-swap models trained on direction-specific data exhibit limited adaptability across multiple directions. Conversely, a model trained on a diverse dataset, incorporating movements from all directions, demonstrates robust performance across all test conditions. Each directional model is represented by two successive contours, which signify equally 'good' quality of deepfake generation in its support region. The contours only partially cover the space, highlighting the importance of diverse training data for unbiased generation. The holes would be successive (third, onwards) contours which are not shown to keep the figure discernible.

Despite their robustness, experiments reveal that under certain conditions, these modules can still produce unexpected results, which can then be used to the defender's advantage. These modules can mistakenly overfit to "face-like" features in a given scene (such as a cartoon of a face). In these cases, the rest of the deepfake generation process gives garbled outputs.

In summary, while the above modules themselves serve as essential pre-processing steps for the next components, they are still vulnerable to challenges that induce out-of-distribution data.

Face-swapper module contains most of the target-specific facial information. Hence, this module is freshly trained on target data, making it *the most vulnerable component* to out-of-distribution data. Challenges designed to degrade earlier components eventually affect the face-swapper, which is inadequate for handling degraded inputs.

Experiments indicate that numerous types of challenges prove effective in disrupting the face-swapper, including (a) structured light illumination, (b) object occlusion, and (c) facial deformations, to name a few. In Fig. 10 d, the generated expressions do not match the original expressions (also corroborated by [13]); the deepfake generator fails to copy the distortions faithfully.

Segmentation. An RTDF pipeline optionally includes a face-segmentation module that improves swap quality. Experiments reveal that if this module is absent, the resulting RTDF is brittle to occlusions, producing unnatural artifacts. E.g., Fig. 10 (a) shows that an occluding object (here, face-mask) blends with the face itself.

Blending. Blending is a multi-stage post-processing task that overlays the swapped prediction of the target face on the imposter's face. These operations range from smoothing, color correction, and positioning to scaling. Color correction (CC) is a vital blending step that samples a color from the imposter's outer face and distributes it over the predicted target face, or vice-versa.

Several hyper-parameters control all the above operations. Any imposter would begin an interaction with a well-calibrated set of hyper-parameters that suits the target-imposter pair and environmental conditions (e.g., lighting). Any inconsistency or abrupt change in these settings can lead to blending artifacts.

Since ambient challenges can introduce abrupt changes in the environment, these hyper-parameters become uncalibrated in the changed setting. For instance, in Fig. 10 f, the flash effect illuminates a smaller portion causing a boundary effect on the fake face. Similarly, in Fig. 10 a, the segmentation module inaccurately includes a surgical mask as part of the face, and the CC module imputes incorrect colors, resulting in clear artifacts.

Temporal incoherence is another factor affecting blending. Most RTDFs generate predictions frame-by-frame independently, or with dependencies on a limited number of past frames (usually $\leq$ 10), as it throttles throughput. In scenarios where an object moves rapidly in front of the face, the face-swap can fluctuate across the imposter's face, when observed frame-wise (see Fig. 5 d). This act results in visible temporal inconsistencies in deepfakes.

Dependency on Data Diversity. We hinted to this as a hurdle to generation in §2.2. To further establish its importance, we conduct an experiment. In the experiment, we recorded five videos of an individual with specific head movements – one covering each of the four directions,

namely, left, right, up, and down, and a fifth covering all of them. A pre-trained DFL model was further trained individually per video, resulting in four 'directional' models and one 'all' direction model.

Fig. 11 illustrates contours of equal fidelity for the five models when used to generate the individual facing all four directions. Ideally, if the result was unbiased, every contour should have covered the figure maximally, like the 'all' contour. However, the directional models, constrained by their training data, exhibited limited support. These results show that a lack of diversity in data can result in biased outputs. In contrast, a diverse dataset enables robust model performance.

6.2. Countermeasures

We consider potential countermeasures an imposter could employ to reduce the effectiveness of GOTCHA. In the threat model, we assumed that the nature of the challenges is *not a secret*. Hence, imposters can anticipate a challenge, and could replay pre-fabricated responses or use a more sophisticated deepfake generator. This subsection discusses why such an adaptation would be complicated and how a defender can fortify against them:

Imposter pre-fabricates responses. An adaptive imposter can fabricate a response to an anticipated challenge, and weed out any artifacts using operations akin to professional video editing, which are unrealistic to perform live.

In order to thwart such efforts, the defender can *include randomness when defining challenges*, thus eliminating non-randomizable tasks (e.g., changing iris color). Supposing the defender randomises each instance of a challenge, an imposter's ability to anticipate or precalculate responses gets significantly hampered [44].

Consider, for instance, that the defender picks a challenge category involving hand occlusions. This challenge category includes positioning a finger before the face or resting a hand on the face. They could individually randomise each challenge by altering its parameters, such as the angle of the hand relative to the face, pose, or distance from the face.

A practical method to introduce these challenges could involve presenting an animated puppet head during the interaction, which carries out the challenge. The user's task would then be to mimic the actions of this puppet. Fig. 3 illustrates one such scenario. The degree of difficulty introduced by randomness may vary across different challenges. However, it universally makes real-time response pre-fabrication more challenging.

Incorporating randomness allows assembling a diverse suite of unpredictable instances even of the same challenge, bolstering their resilience against adversarial attempts to predict or preempt.

Imposter uses a more sophisticated RTDF. An imposter could refine their RTDF generator to bypass specific challenges, either by using more training data or optimizing their network architecture.

In our extended threat model, we consider such advanced RTDFs, mainly when abundant target data is available. Such a case often occurs for public figures or when the targets themselves act as accomplices, thereby defeating any "what-you-know" authentication schemes by sharing personal information with the imposter.

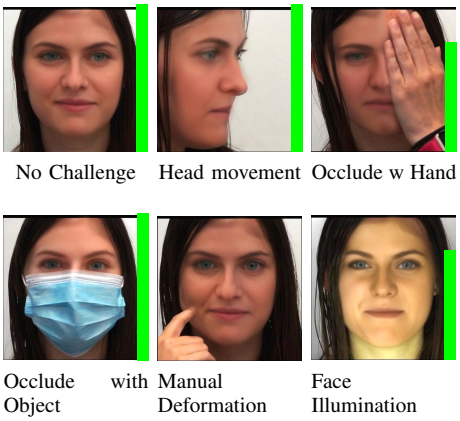


Figure 12. Sample challenge images obtained using a adaptive variant of DFL [6] trained with diverse data for long training time along with a segmentation module. As observed this version is able to do several challenges with lesser artifacts. Green bars are a metaphor to fidelity score, in comparison to Fig. 4. Taller bars imply higher fidelity, and the missing bar denotes failure. Video version at http://govindm.me/gotcha-figures/.

To offset this vulnerability, defenders can employ *a sequence of challenges*. The underlying premise is that while an imposter might successfully bypass a single challenge, evading an entire sequence is considerably more difficult.

To empirically substantiate this approach, we conducted experiments with an advanced variant of DFL [6], which we treated as an adaptive adversary. In this configuration, we enhanced the face-swapping capabilities by training on a more diversified dataset and incorporated a separate, target-specific segmentation module. The modified RTDF underwent over 2M training iterations on a comprehensive dataset, and an additional 1M pretraining iterations on FFHQ [45], cumulatively requiring about three weeks of training.

This advanced RTDF variant proved substantially resilient to the challenges, especially those based on occlusions. However, such results are constrained to individuals with abundant data availability, typically celebrities (see Fig. 12).

We gauged the efficacy of deploying a sequence of four challenges using human evaluations and automated detection algorithms. These assessments replicated the methodologies outlined in §5.1 and §5.2, but included a condition where *evaluators encountered multiple challenge videos for the same identity in conjunction with any prior challenges*. Evaluations were conducted across ten identities, incorporating four challenges plus a 'no-challenge' control. The human evaluation involved an average of 13 responses per video sample.

Fig. 13 illustrates the mean degradation scores — automated and human — when provided with a sequence of challenges. As the imposter navigates through the sequence, both scores escalated close to 90% (for reference, in §5.1 human threshold was 37%). This monotonic boost validates the usefulness of this strategy in authenticating even an adaptive imposter in under a minute.

As previously articulated, challenges exploit vulnerabilities in specific components of RTDF generators. Con-

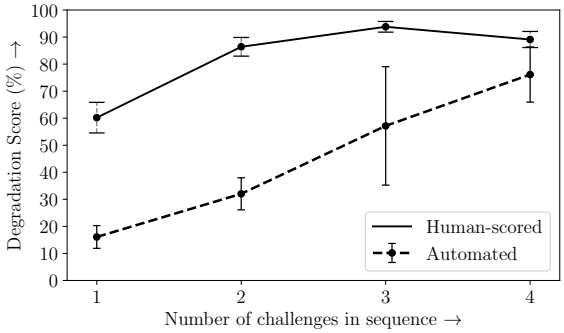


Figure 13. Degradation scored by humans and the in-house model of an adaptive deepfake generator as the challenge sequence progresses. Human scores are given by $\mathcal{H}$ and its 95% confidence interval, while model-score are given by $\mathcal{A}$ and its standard deviation.

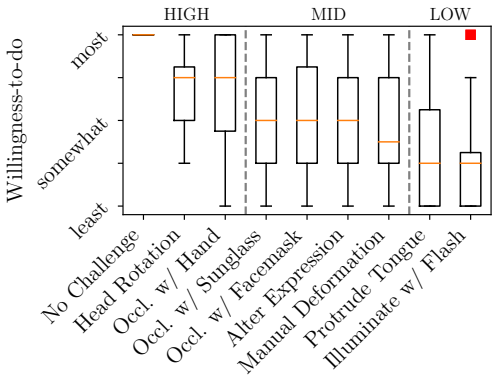


Figure 14. Boxplot of Usability ratings on a five-point Likert scale. Users responded to 'How willingly would you do this task in an online meeting, for verifying yourself?'. The challenges are ordered by decreasing median rating (n = 25). The dotted lines show the separation of high, moderate, and low usable challenges.

sequently, an advanced RTDF would need to universally fortify all its components to sidestep the gamut of possible challenges entirely.

Of course, we do not rule out the existence of such an RTDF. Hence, we acknowledge that challenge-response systems have the inherent uncertainty of remaining effective in the face of rapid technological advancements. However, the resilience of existing CAPTCHA solutions offers an optimistic precedent. As deepfakes mature, GOTCHA emerges as a promising, proactive defense mechanism in the ever-evolving battle against RTDFs.

6.3. Usability

In the preceding discussion, we advocated for the randomization of challenges and the implementation of challenge sequences to strengthen security. While these measures enhance robustness against imposters, they inherently increase the complexity of the authentication process, potentially at the expense of user experience. Thus, designing practical challenges necessitates a nuanced balance between security and usability, considering user-specific contexts, and the optional but beneficial presence of visible artifacts. We discuss them as follows:

User-cooperation. In practical scenarios, a defender needs user cooperation. As the correct response to a challenge is necessary for verification, users must willingly perform the requested actions. Thus, a challenge must be minimally disruptive to the user experience and maximally reveal anomalies in an imposter's video. To gauge the participants' willingness to perform such challenges in a practical scenario like online meetings, we administered a post-participation survey during data collection (Fig. 14).

Some challenges inherently possess superior usability; for example, simple actions like removing glasses or gesturing could be adopted more readily than disruptive actions like spilling water or poking cheeks. We compared Fig. 14 with Fig. 6 and observed that moderately usable challenges were the most effective in inducing degradation. Also, Table 2 symbolizes subfactors such as comprehensibility, appropriateness, physical effort, equipment necessity, and human detectability for each category to help make more informed deployment decisions, described further in §A.5.

The sensitivity of the call also influences user cooperation. Defenders have more leeway to enforce compliance in critical identity verification scenarios, such as job interviews or online exams. Conversely, automated detection can address concerns about challenge appropriateness and potential participant reluctance in less sensitive settings like business meetings. GOTCHA offers real-time automated evaluation without the need to keep recordings. It reassures users that a machine will independently assess their responses, which are then promptly deleted post-verification. To alleviate selfie anxiety in participants, GOTCHA may only display the outlines of participants' faces and remove extensive detail [46].

User-context. A well-designed challenge should account for the situational context in which it will be deployed. The efficacy and appropriateness of a challenge may depend on many factors, such as the user's specific actions, environmental lighting conditions, network bandwidth, and pertinent security requirements.

Human Evaluation Constraints. When designing challenges that permit human evaluation, it is crucial to consider the limitations of human response time and perceptual acuity. For instance, rapid challenges, such as those based on fast illumination changes, could confound human evaluators, making real-time assessment impractical. Thus, challenges should be tailored to be discernible and interpretable within human reaction times and, preferably, induce visible artifacts that facilitate easier identification of deepfakes.

7. Related Work

Our work is related to several topics. We describe where we intersect and where we differ with each of them:

Deepfake Detection. In recent years, automatic deepfake detection has burgeoned due to an arms race with deepfake generators. They target tell-tale artifacts such as distinct patterns in the frequency domain [10], eye features [11], inner & outer facial features [12], expressions [13], and biological signals. Notably, Vahdati et

al. [47] detect real-time facial reenactment deepfakes by calculating errors caused by inconsistent facial landmarks between the driving video and a frame reconstruction of the speaker. Despite competition against generators and the constant threat of becoming obsolete, several methods are technically impressive. Hence, extending them to real-time detection of an incoming feed is possible.

We found that in practice, however, existing automatic detectors are limited; they too are prone to the same impediments that the generators suffer from and more, namely, poor dataset quality, inconsistent pre-processing, computational resource constraints, and susceptibility to adversarial perturbations (§2). These observations were corroborated by prior work suggesting that these models learn non-interpretable, spurious features [38], [48], [49]. We also empirically confirm that well-known detectors [34], [35] performed unreliably on our dataset (§5.2).

GOTCHA is not a deepfake detector; rather, it is a way to strengthen deepfake detection using challenges and making more confident detection decisions.

Deepfake Datasets. Parallel to the evolution of deepfake detectors, researchers curated several deepfake datasets, such as DFDC [37], FFIW [38] and FaceForensics++ [50]. While such legacy datasets contain diverse deepfake videos regarding environment settings, synthetic methods, and face multiplicity, they do not feature the diverse set of actions assessed in this work in a principled manner. Thus, we could not use traditional deepfake datasets and had to create our own. This decision underscores our commitment to accurately assessing our work to detect deepfakes in novel scenarios.

Human Deepfake Evaluators. To provide a baseline for comparison, some studies evaluate performance of humans in detecting deepfakes. Groh et al. [51] conducted an online study with 15,016 participants and concluded that detectors and humans perform overall comparably, with 82% of humans outperforming the leading detector. Also, when they work together, automated detectors boost human performance. In follow-up work, when Josephs et al. [52] amplified deepfake artifacts, human accuracy and confidence improved, indicating that humans are good at detecting artifacts if they are discernible and are within their reaction speed. GOTCHA's challenges take priority to make the deepfake artifacts human-visible.

Challenge-Response-based Authentication Systems. *Liveness Detection.* Uzun et al. [53] used challenge-response to detect whether a user is live or not by asking a caller to read out a random text, then verify that (1) they read the correct text and (2) matches their voice and face against a pre-enrolled user. Li et al. [44] assessed that practical facial liveness detection systems do not protect against deepfakes, and as deepfakes have evolved into being real-time, an imposter can trivially bypass the first verification step.

Biometric Authentication. Sluganovic et al. [54] track saccade movements after a marker pops on the screen and compares the movement with those of the enrolled user. However, GOTCHA does not require extensive biometric enrollment and works under a stronger threat model.

Real-time Deepfakes. Yasur et al. [55] addressed deepfake detection on audio calls using a similar approach and created their challenges, such as clapping, coughing, and playback, based on four constraints – realism, identity, task, and time. They evaluated their challenges on five audio deepfake generators using a private dataset, ASVspoof21 [56] and realistic-in-the-wild dataset [57]. They concluded that such challenges degrade audio quality and force the generator to expose itself. GOTCHA corroborated their claims but for video calls.

Active Illumination [58] works by projecting hues from the user's screen and then correlating these with the hue reflected off the user's face. This approach assumes deepfake systems cannot accurately generate complex hue illuminations in real time. It analyzes the correlation frame-by-frame, with low correlations indicating it is a deepfake and vice-versa. Corneal Reflection [59], similarly, relies on projecting a specific shape and capturing its reflection in the user's cornea for verification. The technique assumes that webcams can consistently capture eye details and that deepfake algorithms cannot mimic such reflections convincingly. The above works also address video RTDFs and are example challenges of GOTCHA.

8. Limitations and Conclusion

Limitations and Future Work. While challenges effectively aid in detecting real-time deepfakes, they have limitations, which we will explore in future work.

Demographic Variations: Our dataset is limited to 47 identities, covering a variety of races, genders, and ages (as shown in Fig. A.1). While diverse, this sample size cannot account for the full spectrum of human facial diversity. We acknowledge that this limitation could introduce bias into our qualitative and quantitative evaluations.

Contextual Variability: Our study relies on data collected in a controlled setting with cooperative participants. Actual conditions could present a host of variables we have not accounted for—such as poor network connections, indoor vs. outdoor settings, or adversarial tactics deployed by imposters to confound detection attempts.

Machine-assisted Scoring Improvements: The automated scoring function comprises of a 3D-CNN for fidelity scoring and a challenge-specific compliance detector. Potential avenues for enhancement include augmenting it with a single compliance detector, which can work for all challenges, and using a hyper-parameter optimized architecture along with uncertainty quantifiers to elevate its accuracy and reliability.

Conclusion. We presented GOTCHA, a challenge-response approach designed to authenticate live video interactions in an environment increasingly susceptible to real-time deepfakes. By exploiting inherent weaknesses in contemporary deepfake generation algorithms, GOTCHA employs a carefully curated set of challenges that produce easily identifiable and human-visible artifacts in videos generated by state-of-the-art deepfake pipelines.

Through validation on a novel and unique challenge dataset, with the help of human evaluators and using a fidelity score model, we empirically establish the efficacy of GOTCHA. The observations made during these assessments, and backed by insights, we offer a nuanced understanding of GOTCHA's robustness against adaptive adversaries, and delineate a security-usability tradeoff.

Resource Availability

Dataset. The full dataset of original recorded challenges, corresponding deepfakes, and derivative deepfake generators are released under a CC-BY-NC-SA 4.0 license. This data will be available exclusively to accredited institutions/organizations for non-commercial research.

Code. We release the code to train the fidelity score function (and dataset) at https://github.com/mittalgovind/GOTCHA-Deepfakes.

Human Evaluation Instruments. We also release the instruments utilized during human evaluations for easy cloning and preview at https://app.gorilla.sc/openmaterials/693684.

Acknowledgement

This material is partially supported by the National Science Foundation under Grant No. 1956200. The authors were partially supported by the AI Research Institutes Program supported by NSF and USDA-NIFA under grant no. 2021-67021-35329, NSF SaTC grant 2154119, and a Cyber NYC gift from Google Research.

References

[1] GETVOIP, "The State of Video Conferencing in 2023." https://getvoip.com/blog/state-of-conferencing/, 2023. [Online; accessed 22-May-2023].

[2] Sensity, "Deepfakes vs biometric KYC verification." https://sensity.ai/blog/deepfake-detection/deepfakes-vs-kyc-biometric-verification/, 2022. [Accessed: 22-May-2023].

[3] NPR, "Deepfake video of Zelenskyy could be 'tip of the iceberg' in info war, experts warn." https://www.npr.org/2022/03/16/1087062648/deepfake-video-zelenskyy-experts-war-manipulation-ukraine-russia. [Accessed: 22-May-2023].

[4] Reuters, "Deepfake scam in China fans worries over AI-driven fraud." https://www.reuters.com/technology/deepfake-scam-china-fans-worries-over-ai-driven-fraud-2023-05-22/. [Accessed: 22-May-2023].

[5] E. Mollick, "A quick and sobering guide to cloning yourself." https://www.oneusefulthing.org/p/a-quick-and-sobering-guide-to-cloning. [Accessed: 30-Apr-2023].

[6] I. Perov, D. Gao, N. Chervoniy, K. Liu, S. Marangonda, C. Umé, M. Dpfks, C. S. Facenheim, L. RP, J. Jiang, S. Zhang, P. Wu, B. Zhou, and W. Zhang, "DeepFaceLab: Integrated, flexible and extensible face-swapping framework," *arXiv:2005.05535 [cs, eess]*, June 2021. arXiv: 2005.05535.

[7] Bloomberg, "Deepfake Imposter Scams Are Driving a New Wave of Fraud." https://www.bloomberg.com/news/articles/2023-08-21/money-scams-deepfakes-ai-will-drive-10-trillion-in-financial-fraud-and-crime. [Accessed: 26-Aug-2023].

[8] CyberScoop, "The growth in targeted, sophisticated cyberattacks troubles top FBI cyber official ." https://cyberscoop.com/fbi-worries-about-future-cyber-threats. [Accessed: 25-May-2023].

[9] Guardian, "Kremlin critic Bill Browder says he was targeted by deepfake hoax video call." https://www.theguardian.com/world/2023/may/25/kremlin-critic-bill-browder-says-he-was-targeted-by-deepfake-hoax-video-call. [Accessed: 25-May-2023].

[10] F. Marra, D. Gragnaniello, L. Verdoliva, and G. Poggi, "Do GANs Leave Artificial Fingerprints?," in *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pp. 506–511, Mar. 2019.

[11] H. Guo, S. Hu, X. Wang, M.-C. Chang, and S. Lyu, "Eyes tell all: Irregular pupil shapes reveal gan-generated faces," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2904–2908, IEEE, 2022.

[12] X. Dong, J. Bao, D. Chen, T. Zhang, W. Zhang, N. Yu, D. Chen, F. Wen, and B. Guo, "Protecting celebrities from deepfake with identity consistency transformer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9468–9478, 2022.

[13] G. Mazaheri and A. K. Roy-Chowdhury, "Detection and localization of facial expression manipulations," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1035–1045, 2022.

[14] G. Boccignone, S. Bursic, V. Cuculo, A. D'Amelio, G. Grossi, R. Lanzarotti, and S. Patania, "Deepfakes have no heart: A simple rppg-based method to reveal fake videos," in *International Conference on Image Analysis and Processing*, pp. 186–195, Springer, 2022.

[15] Y. Nirkin, Y. Keller, and T. Hassner, "Fsganv2: Improved subject agnostic face swapping and reenactment," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 1, pp. 560–575, 2022.

[16] Y. Wang, D. Yang, F. Bremond, and A. Dantcheva, "Latent image animator: Learning to animate images via latent space navigation," in *International Conference on Learning Representations*, 2022.

[17] E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Computer graphics and applications*, vol. 21, no. 5, pp. 34–41, 2001.

[18] S. Athar, Z. Xu, K. Sunkavalli, E. Shechtman, and Z. Shu, "Rignerf: Fully controllable neural 3d portraits," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20364–20373, 2022.

[19] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in Neural Information Processing Systems*, vol. 34, pp. 8780–8794, 2021.

[20] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, 2022.

[21] Synthesia, "Text-to-synthetic videos." https://www.synthesia.io/. [Accessed: 25-May-2023].

[22] J. Thies, M. Elgharib, A. Tewari, C. Theobalt, and M. Nießner, "Neural voice puppetry: Audio-driven facial reenactment," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pp. 716–731, Springer, 2020.

[23] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe, "First order motion model for image animation," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[24] E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Computer Graphics and Applications*, vol. 21, no. 5, pp. 34–41, 2001.

[25] Unite.AI, "Google Has Banned the Training of Deepfakes in Colab." https://www.unite.ai/google-has-banned-the-training-of-deepfakes-in-colab/, 2022. [Accessed: 22-May-2023].

[26] M. Stypułkowski, K. Vougioukas, S. He, M. Zięba, S. Petridis, and M. Pantic, "Diffused heads: Diffusion models beat gans on talking-face generation," *arXiv preprint arXiv:2301.03396*, 2023.

[27] X. Li, S. De Mello, S. Liu, K. Nagano, U. Iqbal, and J. Kautz, "Generalizable one-shot neural head avatar," *Arxiv*, 2023.

[28] M. Kim, F. Liu, A. Jain, and X. Liu, "Dcface: Synthetic face generation with dual condition diffusion model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12715–12725, 2023.

[29] "How Airbnb Verifies Identities." https://www.airbnb.com/help/article/1237. Accessed: 2024-02-29.

[30] "Uber Delivery Agents: Identity Verification." https://www.uber.com/en-GB/blog/identity-verification/. Accessed: 2024-02-29.

[31] "ID.me for KYC by US Government Agencies." https://network.id.me/platform/identity-verification/. Accessed: 2024-02-29.

[32] "ETS for At-Home Testing." https://www.ets.org/gre/test-takers/general-test/register/at-home-testing.html. Accessed: 2024-02-29.

[33] T. Register, "'How not to hire a North Korean plant posing as a techie' guide updated by US and South Korean authorities." https://www.theregister.com/2023/10/19/north_korea_fake_freelance_avoidance/. [Accessed: 24-Oct-2023].

[34] K. Shiohara and T. Yamasaki, "Detecting deepfakes with self-blended images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18720–18729, 2022.

[35] Y. Zheng, J. Bao, D. Chen, M. Zeng, and F. Wen, "Exploring temporal coherence for more general video face forgery detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15044–15054, 2021.

[36] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-df: A large-scale challenging dataset for deepfake forensics," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3207–3216, 2020.

[37] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. C. Ferrer, "The deepfake detection challenge (dfdc) dataset," *arXiv preprint arXiv:2006.07397*, 2020.

[38] T. Zhou, W. Wang, Z. Liang, and J. Shen, "Face forensics in the wild," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5778–5788, 2021.

[39] K. Hara, H. Kataoka, and Y. Satoh, "Learning spatio-temporal features with 3d residual networks for action recognition," in *Proceedings of the IEEE international conference on computer vision workshops*, pp. 3154–3160, 2017.

[40] O. Kopuklu, J. Zheng, H. Xu, and G. Rigoll, "Driver anomaly detection: A dataset and contrastive learning approach," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 91–100, 2021.

[41] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, *et al.*, "Mediapipe: A framework for building perception pipelines," *arXiv preprint arXiv:1906.08172*, 2019.

[42] T. Hempel, A. A. Abdelrahman, and A. Al-Hamadi, "6d rotation representation for unconstrained head pose estimation," in *2022 IEEE International Conference on Image Processing (ICIP)*, pp. 2496–2500, IEEE, 2022.

[43] A. V. Savchenko, "Facial expression and attributes recognition based on multi-task learning of lightweight neural networks," in *2021 IEEE 19th International Symposium on Intelligent Systems and Informatics (SISY)*, pp. 119–124, 2021.

[44] C. Li, L. Wang, S. Ji, X. Zhang, Z. Xi, S. Guo, and T. Wang, "Seeing is living? rethinking the security of facial liveness verification in the deepfake era," in *31st USENIX Security Symposium (USENIX Security 22)*, pp. 2673–2690, 2022.

[45] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.

[46] "Why iProov Genuine Presence Assurance Doesn't Use Selfies." https://www.iproov.com/blog/genuine-presence-assurance-selfie-anxiety. Accessed: 2024-02-29.

[47] D. S. Vahdati, T. Duc Nguyen, and M. C. Stamm, "Defending low-bandwidth talking head videoconferencing systems from real-time puppeteering attacks," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 983–992, 2023.

[48] B. Le, S. Tariq, A. Abuadbba, K. Moore, and S. Woo, "Why do facial deepfake detectors fail?," in *Proceedings of the 2nd Workshop on Security Implications of Deepfakes and Cheapfakes*, WDC '23, (New York, NY, USA), p. 24–28, Association for Computing Machinery, 2023.

[49] K. Chandrasegaran, N.-T. Tran, and N.-M. Cheung, "A closer look at fourier spectrum discrepancies for cnn-generated images detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 7200–7209, 2021.

[50] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "FaceForensics++: Learning to detect manipulated facial images," in *International Conference on Computer Vision (ICCV)*, 2019.

[51] M. Groh, Z. Epstein, C. Firestone, and R. Picard, "Deepfake detection by human crowds, machines, and machine-informed crowds," *Proceedings of the National Academy of Sciences*, vol. 119, p. e2110013119, Jan. 2022.

[52] E. Josephs, C. Fosco, and A. Oliva, "Artifact magnification on deepfake videos increases human detection and subjective confidence," *arXiv preprint arXiv:2304.04733*, 2023.

[53] E. Uzun, S. P. H. Chung, I. Essa, and W. Lee, "rtCaptcha: A Real-Time CAPTCHA Based Liveness Detection System," in *Proceedings 2018 Network and Distributed System Security Symposium*, (San Diego, CA), Internet Society, 2018.

[54] I. Sluganovic, M. Roeschlin, K. B. Rasmussen, and I. Martinovic, "Using reflexive eye movements for fast challenge-response authentication," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1056–1067, 2016.

[55] L. Yasur, G. Frankovits, F. M. Grabovski, and Y. Mirsky, "Deepfake captcha: A method for preventing fake calls," in *Proceedings of the 2023 ACM Asia Conference on Computer and Communications Security*, ASIA CCS '23, (New York, NY, USA), p. 608–622, Association for Computing Machinery, 2023.

[56] J. Yamagishi, X. Wang, M. Todisco, M. Sahidullah, J. Patino, A. Nautsch, X. Liu, K. A. Lee, T. Kinnunen, N. Evans, *et al.*, "Asvspoof 2021: accelerating progress in spoofed and deepfake speech detection," *arXiv preprint arXiv:2109.00537*, 2021.

[57] N. M. Müller, P. Czempin, F. Dieckmann, A. Froghyar, and K. Böttinger, "Does audio deepfake detection generalize?," *arXiv preprint arXiv:2203.16263*, 2022.

[58] C. R. Gerstner and H. Farid, "Detecting real-time deep-fake videos using active illumination," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 53–60, 2022.

[59] H. Guo, X. Wang, and S. Lyu, "Detection of real-time deepfakes in video conferencing with active probing and corneal reflection," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, IEEE, 2023.

[60] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[61] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics* (Y. W. Teh and M. Titterington, eds.), vol. 9 of *Proceedings of Machine Learning Research*, (Chia Laguna Resort, Sardinia, Italy), pp. 297–304, PMLR, 13–15 May 2010.

Supplementary Material for GOTCHA

Outline

The supplementary material contains further details and figures, along with their backlinks, as follows:

1) Dataset collection, referred in §4.2.
2) Compute Environments used for training and/or inference of RTDFs (§4.2).
3) In-house fidelity score function, referred in §5.2.
4) Human Evaluation experiments, referred in §5.1.
5) Explanation of Usability Benefits, referred in §6.3.

A.1. Data Collection

Instructions for Participants. We gave participants the following instructions and recorded a video for each task:

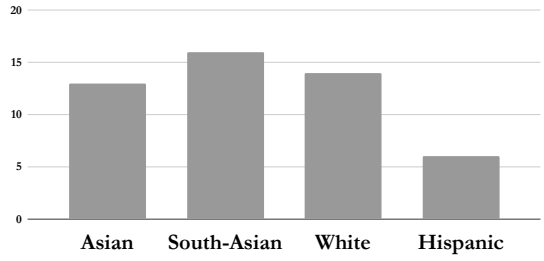
1) While the camera moves around you for about two minutes, capturing the face from different angles, please sit still.
2) Rotate your head from side to side, then look up and down. Please spend about 5 seconds on each side, rotating as comfortably as possible, totaling around 25 seconds.
3) Cover your eyes with a hand, followed by covering the left half, the right, and finally, the lower half of the face.
4) Put on the provided sunglasses, and then take them off.
5) Wear the provided clear glasses, ensuring they reflect a lamp light shining. After setting up the reflection, we start recording. Finally, the participant should remove the glasses.
6) Put on a face mask and count from 1 to 10 out loud. Then, remove the facemask.
7) Press a finger against a cheek.
8) Stick out a small portion of the tongue.
9) Laugh for 10 seconds, then frown, as if angry, for another 10 seconds.
10) Slowly stand up and then sit back down.
11) Dim the room light and keep the flashing light on the face for 10 seconds.

Although we only evaluated eight challenges in this work, we collected more to develop insights into what challenges work and why.

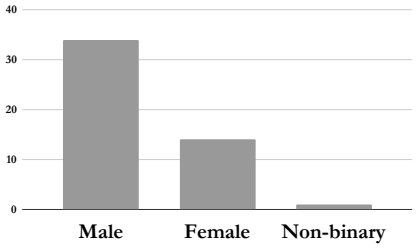
It is important to note that we intentionally refrained from introducing artificial randomness into the dataset. While randomness could help prevent pre-computed solutions, it does not significantly contribute to creating more degrading artifacts. This approach ensured the dataset had natural variability, capturing a broad distribution of challenge performances across participants. Unless otherwise noted, all source images used in this paper are samples from the dataset collected.

Demographics. The demographics of the collected dataset is as follows (also illustrated in Fig. A.1):

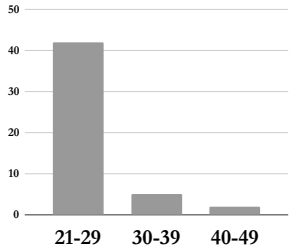
- *Race* – Asian: 27.7%, Hispanic: 12.8%, South Asian: 29.8%, White: 29.8%
- *Age* – 21-29: 87.5%, 30-39: 9.4%, 40-49: 3.1%
- *Gender* – Male: 59.4%, Female: 37.5%, Non-Binary: 3.1%.



(a) Race



(b) Gender



(c) Age

Figure A.1. Bar plots of Demographic distribution of participants (n = 47) recorded to create the original part of our dataset, with Y-axes indicating count per sub-category – (a) Race, (b) Gender, and (c) Age.

A.2. Compute Environments

We trained DFL [6] models using a 16-core Intel(R) Xeon(R) Platinum 8358 CPU @ 2.60GHz with 80 GB of RAM and one NVIDIA A100 GPU equipped with 80 GB VRAM. We performed all RTDF pipeline inference tasks (e.g., during video calls) on an octa-core Intel(R) Core(TM) i7-9700K CPU @ 3.60GHz machine with 32 GB RAM and an NVIDIA RTX 3070 GPU with 8 GB VRAM.

A.3. Fidelity Score Function

Given the unreliability of existing state-of-the-art deepfake detectors with our challenge videos, we felt the need to develop a custom deepfake detector for our automated evaluation. Drawing inspiration from the pioneering work by Kopuklu et al. [40], who employed a contrastive learning paradigm for detecting anomalies during car-driving, we designed a similar approach.

Dataset Composition. In line with our earlier discussion in §5.2, the dataset selection involved:

- Choosing both original and deepfake videos spanning four out of eight challenge types, including the 'no challenge' category.

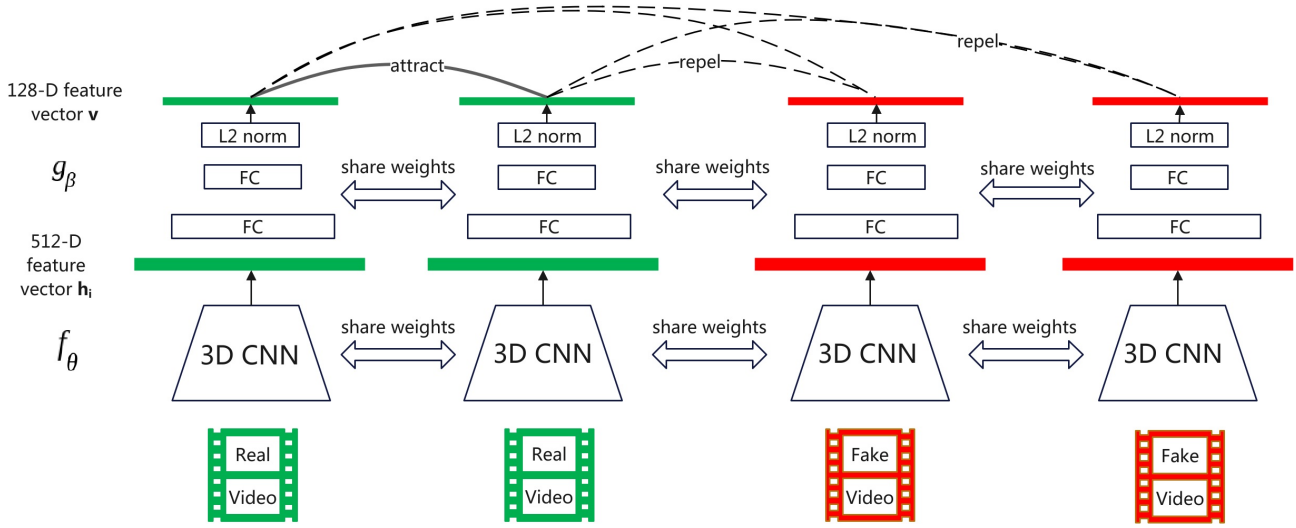


Figure A.2. Training workflow of the fidelity score function. A 3D CNN processes a batch containing original and fake videos. The CNN along with the fully connected head optimize a loss function which maximizes the similarity of original videos (solid line) while minimizing the similarity in original-fake video pairs further apart (dotted lines).

- Ensuring 32 frames per sample, with each frame resized to 224×224.
- Incorporating 35 of the 47 target identities.
- Restricting to deepfake videos generated using the DFL approach.

This curated selection resulted in 175 original sample and 8,050 deepfake sample for the training phase. All samples were normalized with a global mean and standard deviation.

Framework Overview. Our primary objective was to learn representations that distinctly differentiate between authentic videos and deepfakes. By maximizing the similarity among genuine samples while minimizing the similarity between authentic and fake samples in a latent space, we aim to detect anomalies. The overall structure, visually represented in Fig. A.2, comprises:

- We use a **base encoder** $enc_\theta(.)$ to extract vector representations of input video clips. $enc_\theta(.)$ is a 3D-CNN with parameters θ. Specifically, We use a ResNet-18 as the candidate function to map an input clip $\mathbf{x} \in \mathbb{R}^{N \times H \times W \times 3}$ into $\mathbf{h} = \{h_i \in \mathbb{R}^{512}, s.t., \mathbf{h}_i = enc_\theta(\mathbf{x}_i), \forall i\}$.
- After we use a **projection head** $g_\beta(.)$ to map $\mathbf{h}$ to a latent vector embedding $\mathbf{v}$. Specifically, we use an MLP with one hidden layer and a ReLU activation, parameterized by β, to transform $\mathbf{v} = \{\mathbf{v}_i \in \mathbb{R}^{128}, s.t., \mathbf{v}_i = g_\beta(\mathbf{h}_i), \forall i\}$. After MLP, we normalize the embedding $\mathbf{v}$ using L2 norm.
- Finally, a **contrastive loss** is used to impose that normalized embeddings from the original videos (positive class) are closer together than embeddings from deepfake videos (negative class).

Within a mini-batch, we have K original videos and M deepfake videos with $i \in \{1, \ldots, K+M\}$. Final embedding of the i^{th} original and deepfake videos are denote as v_{oi} and v_{di}, respectively. There are in total $K(K-1)$ positive pairs and KM negative pairs in every mini-batch. Then the loss takes the final form:

$$\mathcal{L}_{ij} = -\log \frac{exp(v_{oi}^T v_{oj}/\tau)}{exp(v_{oi}^T v_{oj}/\tau) + \sum_{m=1}^{M} exp(v_{oi}^T v_{dm}/\tau)}$$

$$\mathcal{L} = \frac{1}{K(K-1)} \sum_{i=1}^{K} \sum_{j=1}^{K} \mathbf{1}_{j \neq i} \mathcal{L}_{ij}.$$

where $\tau \in (0, \infty)$ is a scalar temperature variable that can controls the spread of the learned distribution [60]. Typically, τ is chosen between 0 and 1 to amplify the similarity between samples. The inner product of embedding vectors measures the cosine similarity, as they are all L2 normalized. By optimizing the above loss term, the encoder learns to maximize the similarity between the original feature vectors and minimizing the dissimilarity between the original feature vectors and the deepfake videos.

We used Noise Constrastive estimation [61] to estimate the full softmax distribution, to speed up the training process because it avoids the expensive normalization step. Instead NCE differentiates between the true data and artificially generated noise and over time, the model learns to assign higher probabilities to true data samples than to noise.

Training Hyperparameter Details. We trained the whole framework from scratch for 120 epochs using Adam optimizer. The learning rate was 1e-3 with weight decay of 1e-5, temperature $\tau = 0.5$ and window size of 16 frames. Each mini-batch had $K = 4$ original and $M = 190$ deepfake samples, keeping $1:47$ ratio of target : imposters. We attempted to use a pre-trained 3D-CNN, however, in literature they have been predominantly used for action recognition tasks, which are devoid of facial features.

Scoring an individual video. For the test time scoring of individual videos, Kopuklu et al. [40] proposed a new evaluation protocol. After the training phase, only the 3D-CNN model is retained. This model encodes all N original training set videos into a set of L2 normalized 512-

dimensional representations. Hence, the original template vector v_o is given by:

$$v_o = \frac{1}{N}\sum_{i=1}^{N}\frac{enc_\theta(x_i)}{||enc_\theta(x_i)||_2}.$$

To get a similarity score of a test video x_{test}, we simply compute the fidelity score f to be cosine similarity between the encoded clip and v_o by:

$$f(x_{test}) = v_o^T\frac{enc_\theta(x_i)}{||enc_\theta(x_i)||_2}.$$

We used this score to calculate the difference between deepfake videos and their original counterparts. Hence, a larger deviation from the ground truth indicates lower fidelity of the deepfake video, serving as the fidelity score for deepfake evaluation.

The self-supervised training paradigm adopted above gives itself to new video datasets and newer challenges with minimal data pre-processing and without the need for providing labels.

A.4. Human Evaluation details

We used Gorilla.sc for creating our human evaluation experiments and Prolific to recruit participants. The workflow for a consenting participant who passed the quiz was: Artifact Definitions → Quiz → Video Evaluations.

Definitions of Artifacts. All participants were provided with the following definitions and two sample videos of artifacts. Fig A.3 lists one of the two samples.

- *No Artifact:* The video exhibits no inconsistencies or visible artifacts, appearing both natural and genuine. This category may also contain harmless artifacts or noise unrelated to deepfake manipulation. Note that the quivering of the whole video (observed on the right) is an artifact of headshot extraction and not a deepfake artifact.
- *Facial Boundary Artifacts:* These are mismatches or inconsistencies visible when the digitally manipulated face meets the original face, manifesting as unnatural transitions, blurring, or skin tone inconsistencies between facial regions.
- *Object Vanishes:* This artifact features abrupt disappearances of specific objects or features within the video frame (e.g., sunglasses or hands), creating an apparent visual inconsistency.
- *Skin Texture Inconsistency:* This artifact results from uneven or inconsistent skin texture or detail. For example, one facial region may appear unusually smooth, overly sharp, or lighted up, contrasting sharply with other areas. Note that the movement of the black boundary frame is an artifact of headshot extraction and not a deepfake artifact. On the left, the person looks side-to-side, and on the right, they stand and sit down quickly.
- *Temporal Inconsistency:* This artifact is characterized by inconsistencies in facial movements or expressions over time. Such inconsistencies are most noticeable when observing a sequence of frames. They can manifest as flickering skin or rigid facial movements that deviate from natural human behavior.

Quiz. We administered a multiple-choice quiz, asking participants to pick the most apparent artifact in ten unseen videos. We have kept the videos close to the original videos shown with artifact definitions, and failure was granted if they scored below $6/10$. We used this process as a proxy to filter un-attentive participants and teach them about artifacts. 43 out of 60 recruited participants passed the quiz. We paid the disqualified participants partially for their time.

Video Evaluations. Fig. A.4 illustrates how we collected responses for each deepfake and original video from each consenting and passing human evaluator.

Video Evaluations of the adaptive adversary. Fig. A.5 illustrates how we collected responses for each deepfake and original video from each consenting and passing human evaluator for adaptive adversaries. The main difference was showing videos from past challenges with the new challenge.

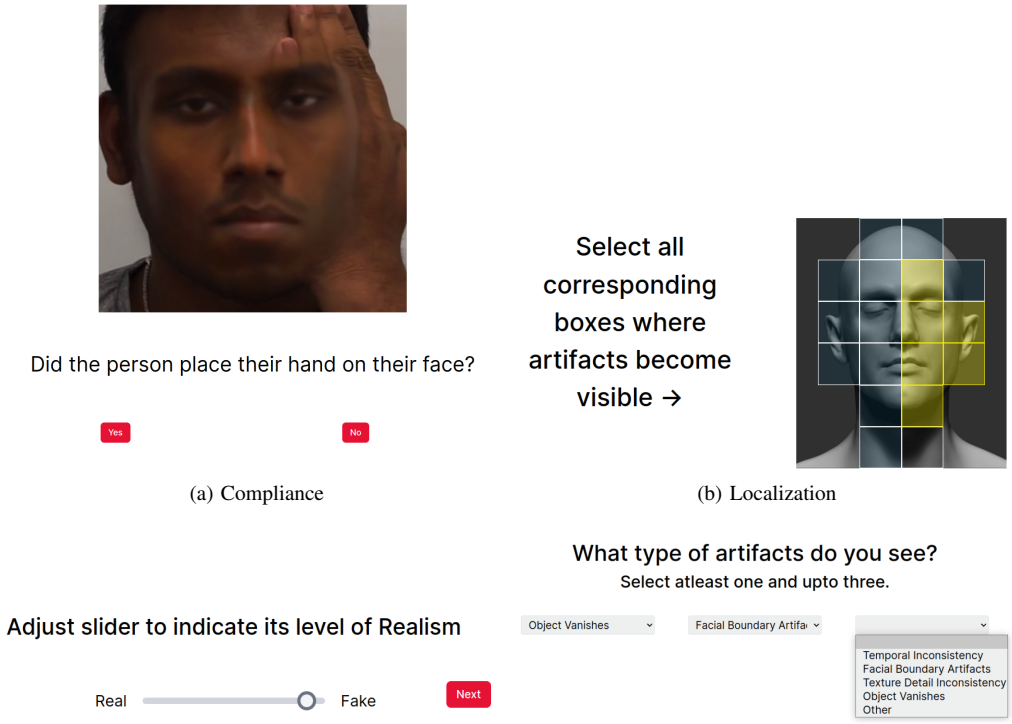
A.5. Usability Benefits

This category contains participant-facing benefits, which indicate how practical a challenge is in a video-call setting. As this category only includes participant-facing benefits, a particular benefit could either awarded an *Offered*, *Not-offered* or *Quasi* (partially offered) status. The following challenges are granted an *Offered* status:

- **Easy-to-Comprehend**: If the challenge requires action (i.e., is active), it is also easily understood, and the task comes naturally to a human participant. *Quasi* is granted if it is difficult to understand, even if done automatically by the camera.
- **Appropriate-to-Request**: If the task must be completed, it can be completed without hesitation or embarrassment. It is still permitted if done automatically without the participant's awareness during the call.
- **Physically-Effortless**: If the participant is not required to do anything except press a button. *Quasi* is granted if performed challenges do not require more than a simple task, which a participant might even undertake naturally during a video call interaction. Passive challenges are deemed physically low-effort by default.
- **No-Equipment-Needed**: Suppose the participant does not require extra equipment to complete the task. They would join the call as usual, with no expectations.
- **Detected-by-Humans**: If the challenge introduces artifacts or inconsistencies that humans could perceive distinctly. *Quasi* is granted if the artifacts could be seen with the keen eye of a human evaluator who might be specifically looking for them. If the challenge would not introduce any particular observable artifacts, *Not-Offered* is granted.

(a) No Artifact (b) Facial Boundary Artifacts (c) Object Vanishes (d) Skin Texture Inconsistency (e) Temporal Inconsistency

Figure A.3. Video samples that were shown to each participant along with definitions.



(a) Compliance (b) Localization

(c) Realism (d) Justification

Figure A.4. Human evaluation of a deepfake video included four sub-responses – (a) Compliance (b) Localization (c) Realism, and (d) Justification. Each sub-response was recorded in a separate window with the video available at all times.

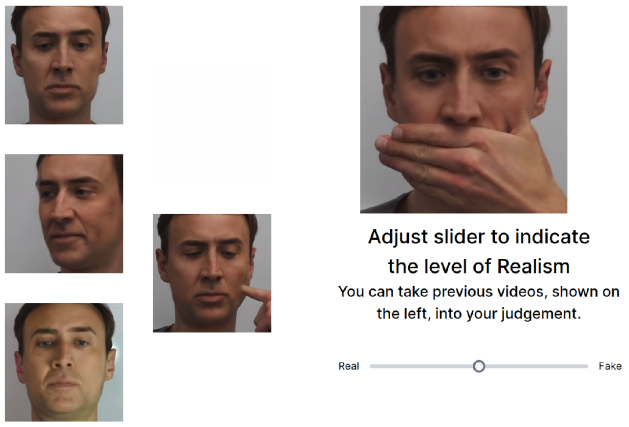


Figure A.5. Human evaluation of a sequence of deepfake challenge videos. The videos were created using the adaptive RTDF pipeline. Each human response is supported with previous challenges in progression.