BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2026
http://excel.fit.vutbr.cz

Detection of Facial Deepfakes Using Interactive Liveness Tests

David Drtil*

Abstract

The face you show in a Teams call, a Google Meet or a WhatsApp video chat has quietly become a trust anchor. One public profile photo is enough for a real-time face-swap tool to wear that face on a consumer GPU, and the passive liveness checks that used to catch the fakes keep falling behind with every new generator. Our answer is to stop assuming the camera is honest. We combine the usual passive analysis of texture and motion with an interactive challenge-response protocol: random physical actions the user has to perform on camera, inside short time windows. The detector scores every frame, and the decision logic fuses per-action peaks with the success or failure of each challenge. Everything runs as a real-time web application built around *MediaPipe*, *UCF* for spatial artifacts and *CViT-v2* as the temporal model. An adversarial test harness drives live attacks through *DeepFaceLive* and *FaceFusion*, in those streams the passive score visibly climbs during the action windows, and the temporal model also flags pre-recorded replay loops that otherwise slip past any static check. A ten-second interactive layer is cheap, deployable, and stays useful exactly when the passive detector is silently out of domain against the next model release.

*xdrtil03@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

[Motivation] Video calls are everywhere now. We meet colleagues on *Microsoft Teams*, family on *Messenger* or *WhatsApp*, and we take for granted that the face on the other side of the call is the person we expect. A single public photo, the one already sitting on Facebook or LinkedIn, is all a tool like *FaceFusion* or *DeepFaceLive* needs to wear that face live on an ordinary gaming GPU. Figure 1 on the poster shows the author swapped onto Tom Cruise in exactly that setup. The consequences are already in the wild: family-emergency phone-video scams, reputation-based extortion built on fabricated footage, and corporate attackers who join an internal meeting with a stolen face and ask for a quick credential, information or cash transfer.

Passive detectors have caught earlier generations of fakes but lose ground with every new one. A Xception model trained on a single manipulation drops from 99.4 % AUC on the same manipulation to 49.1 % on FaceSwap [1]. In practice, a passive detector that was competitive six months ago is often already out of domain.

[Problem] What we need is a detection scheme that stays useful when the passive detector is silently behind the state of the art, and that forces an attacker to visibly degrade the quality of the swap in ways the remaining passive signal still catches. It has to run on a normal laptop webcam, add as little friction for the honest user as possible, and be robust to replay.

[Existing solutions] Passive spatial detectors such as *SBI* [2] and *UCF* [3], and temporal detectors such as *CViT-v2* [4], all generalise poorly across generators. They work well on the manipulation they were trained against and then fall off a cliff on a new one.

On the attack side, recent security studies of deployed facial liveness systems show that all major mobile-app vendors can be bypassed by a live face-swap [5]. Firc et al. frame the broader threat to speaker and facial authentication [6], and a later analysis of face authentication workflows arrives at similar conclusions about deployed systems and their defences [7].

[Our solution] We add an **interactive layer** on top of the passive pipeline. The user is asked to perform randomised actions from a set: rotating the head, blinking, changing expression, or covering part of the

face with a hand. Each action has a ten-second window. During those windows the passive detectors keep scoring every frame, and the decision logic fuses the per-action passive *peak* scores with the success or failure of each challenge. Figure 3 on the poster sketches the whole engine.

[Contributions] (i) a hybrid passive plus interactive framework with per-action peak score aggregation; (ii) a real-time web application built on *MediaPipe*, *UCF* and *CViT-v2* that streams over WebSocket to a thin-client browser; (iii) an adversarial test harness with identity-matched source and target pairs driven through *DeepFaceLive* and *FaceFusion*; and (iv) a replay-safe protocol based on randomised action sequences.

2. Method

The system is split between a thin-client (just a webcam and a browser) and a server-side Processing Layer. Figure 3 on the poster walks through the whole engine. The two analysis branches are drawn separately in Figure 4 for the passive side and in Figure 5 for the interactive side.

Pre-processing. Frames come in, they get resized to 480 px wide, and a face is detected with *MediaPipe* face mesh that gives us the 468 landmarks we also need for the interactive checks. A simple centroid tracker keeps the same face across frames, and five canonical landmarks feed a similarity transform that aligns the crop with a 25 % margin so that boundary artifacts stay inside the frame rather than leaking out of it.

Passive branch. Three detectors run in parallel. The *spatial* one is *UCF* [3] on an Xception backbone and looks for blending and boundary artifacts. The *temporal* one is *CViT-v2* [4] over a 15-frame sliding window, besides catching landmark jitter and texture flicker, it also serves as our watchdog against replay attacks, since pre-recorded loops tend to contain large frame-to-frame jumps and unnatural temporal transitions that a frame-by-frame check simply cannot see. The *frequency* one is a standalone SPSL module that looks for GAN upsampling fingerprints in the image spectrum. A weighted mean with a 10-frame moving average produces the passive score S_{passive} shown live on the overlay in Figure 2.

Interactive branch. *MediaPipe* emits 468 face landmarks and 21 hand landmarks per frame. An *Action Router* activates only the metrics that are needed for the current challenge, which keeps the per-frame cost down. A *Head Pose Solver* recovers pitch, yaw and roll from a PnP problem on five landmarks. A *Hand Tracker* computes the intersection between the hand mask and the face bounding box, and this is what you see rendered in Figure 2 together with the passive score. *EAR* and *MAR*, the eye- and mouth-aspect ratios known from real-time blink detection [8], close the set for the blink and expression challenges. Hard geometric thresholds, yaw above 30°, occlusion of at least 30 %, *MAR* above 0.5 and *EAR* below 0.2, turn each action into a simple success or failure inside the ten-second window.

Decision logic. An action fails the whole session only if it times out. Otherwise the decision logic picks, for every action window, the peak spatial and temporal scores, and weights them by the action category, so that more complex actions (a full hand occlusion) weigh more than the easy ones (a small expression change). An early-exit flag short-circuits the session the moment a single confident-fake frame is seen (passive score ≥ 0.9), which mean that no amount of interactive polish afterwards can rescue an already-flagged swap. The final output that reaches the client is a boolean PASS or FAIL.

3. Experimental Setup

We collect short recordings of roughly ten to fifteen seconds from a standard laptop webcam. Source and target identity pairs for the adversarial runs are filtered by cosine similarity of *ArcFace* embeddings, so that the swap is in-domain if we let the domain gap grow too wide, the swap breaks trivially and the detector numbers look better than they really are.

The authentic set is recorded straight from the webcam. The attack set routes the same sessions through *DeepFaceLive* (with the pretrained Tom Cruise, Alice Eve and Melissa Benoist DFM models, one of which appears in Figure 1) and, where the target model is compatible, through *FaceFusion* running on a single image of the impersonated identity. Per-session logs contain the raw spatial, temporal and frequency scores, the landmark-derived interactive metrics, and the pass or fail outcome.

We evaluate each branch individually and the combined hybrid on accuracy, F1, ROC/AUC and equal error rate, with a particular focus on the score separation *inside* the challenge windows, since that is where our argument lives or dies.

4. Conclusions and Future Work

A ten-second interactive check is cheap, easy to deploy, and still useful when the passive detector is

silently losing ground. The system already runs end-to-end in real time on a single GPU workstation and a commodity browser, so the question from here is no longer *can it run* but *how far does it carry us*. The experimental campaign in progress, with full ROC/AUC and EER numbers to land before the thesis submission on 20 May 2026, will quantify how much the interactive layer actually buys on top of the passive baseline.

Natural next steps are a proper *rPPG* biological channel to round out the passive branch on the design side, an identity rejection gate that drops sessions whose *ArcFace* signature drifts mid-session (we currently log drift but do not act on it), and a wider adversarial sweep with more generators, including *DeepFaceLab* and diffusion-based refinement on top of the existing swaps.

Acknowledgements

I would like to thank my supervisor Ing. Anton Firc, Ph.D., for guidance and for access to the testing infrastructure at FIT VUT.

References

[1] Lingzhi Li, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. Face x-ray for more general face forgery detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[2] Kaede Shiohara and Toshihiko Yamasaki. Detecting deepfakes with self-blended images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[3] Zhiyuan Yan, Yong Zhang, Yanbo Fan, and Baoyuan Wu. Ucf: Uncovering common features for generalizable deepfake detection. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[4] Deressa Wodajo, Solomon Atnafu, and Zahid Akhtar. Deepfake video detection using generative convolutional vision transformer. *arXiv preprint arXiv:2307.07036*, 2024.

[5] Xinyan Wang, Kexin Luo, and Wing Cheong Lau. Living a lie: Security analysis of facial liveness detection systems in mobile apps. In *Applied Cryptography and Network Security (ACNS)*, volume 14585 of *LNCS*, pages 391–416. Springer, 2024.

[6] Anton Firc, Kamil Malinka, and Petr Hanáček. Deepfakes as a threat to a speaker and facial recognition: an overview of tools and attack vectors. *Heliyon*, 9(4):1–33, 2023.

[7] Martin Šalko, Anton Firc, and Kamil Malinka. Security implications of deepfakes in face authentication. In *Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing (SAC)*, pages 1376–1384. ACM, 2024.

[8] Tereza Soukupová and Jan Čech. Real-time eye blink detection using facial landmarks. In *21st Computer Vision Winter Workshop (CVWW)*, 2016.