A Review of Secure and Privacy-Preserving Hand Gesture CAPTCHA Systems with Enhanced Liveness Detection

Talashree Danao[1], Akansha Mohite[2], Shravani Kalid[3], Manasvi Satpute[4], Pallavi Gholap[5]
[1,2,3,4,5]*Department of Computer Science and Engineering, MIT Arts Commerce and Science*
doi.org/10.64643/ijirt.208575-459

***Abstract*—Text- and/or image-based CAPTCHAs, as a dominant way of differentiating bots from humans, are now often defeated by computer-Vision & optical-character-recognition models prompting development of other verification techniques that are not easily automatable yet still convenient to humans. This review article presents a new option based on the user performance of randomly picked gesture in front of his/her regular webcam while he/she is under live human examination with just few extracted hand-landmarks. Prior work related to this area has looked at implications for spoofing resistance, usability, and privacy aspects. It has been pointed out that a gesture-based CAPTCHA could also be considered as a spoofing detection in video-and-image-based tasks. This is why the paper also examines different spoofing detection methods that are effective for both task types and evaluates whether they can be used for end users without the requirement to handle or store video data. The paper draws together past work on gesture-based CAPTCHAs, hand-landmark recognition, face/gesture spoofing detection, together with an update on recent public availability and vulnerability disclosure of hand-gestor reCAPTCHA. It emphasizes the accuracy of recognition, capability of spoofing detection, factors on usability and privacy issues of the methods considered with a view to pointing out what remains to be done both on research level and real-life application to create a more pleasant and still beneficial user experience of privacy- and computation-preserving local processing.**

***Index Terms*—CAPTCHA; Hand Gesture Recognition; Computer Vision; Liveness Detection; Spoofing Detection; Human Verification; Privacy-Preserving AI; Bot Detection.**

I. INTRODUCTION

Proving that you are a real user navigating your web browser rather than a bot has been around since almost the very beginning of web-security, and has because of this become one of the most heavily fought over long-term frontiers in the field. From distorted text to weird audio excerpts to structured images two-by-two, challenge designers have battled ever-evolving recognition engines until finally research teams managed to defeat even Google's popular reCAPTCHA system at a large scale [2] [1]. In doing so, they narrowed the challenge offered through the classic reading test to a set of images into something that most automata cannot design away namely, a physically embodied action that can be performed and seen.

Hand gestures seem the natural extension to the user's own what about quick quick... " A short collection of gestures like raising a few fingers, adopting a certain pose or waving quickly could all be visually checked using a standard webcam, with no extra hardware, and unlike a written response are hard for a bot to generate convincingly unless it can also generate a live video image. Cranking on this idea, this review examines the building blocks of hand-gesture CAPTCHA types: how they identify gestures from video, how they layer of liveness and scam detection to beat photo and a video stream scamming, and how they respect privacy by doing the video analysis on the user's computer rather than sending the footage to a server.

Problem Statement

Traditional CAPTCHAs are rapidly becoming ineffective at the one task they are designed for. The latest OCR and vision models can tackle text-distortion and image-spam CAPTCHAs at very similar levels and even at times surpass human abilities. Also, from a usability perspective, those CAPTCHAs tend to drive users away and cause them a lot of frustrations. A CAPTCHA that would rely on gestures might be more resistant to being solved automatically, but at the same time would create an opportunity for bots to imitate humans performing the task. For example, a bot could present a paper picture of a hand to the camera. A bot showing a video loop of a human doing the gesture over and over can be a more advanced way of achieving it. A graphics API can be used to simulate a hand to trick a liveness detector. However, it will still not be an easy solution for a gesture-based CAPTCHA to pass the anti-bot measures without the risk of making the user surrender a sensitive camera output.

Objectives

- To summarize how different algorithms can be utilized in recognizing hand gestures through a webcam stream in real time, with a particular focus on landmark detection and sensor technologies.
- To describe how techniques of liveness verification and anti-spoofing can tell if a human hand is present in the frame or if it is actually a photo, a video replay, or a fake synthetic feed.
- To analyze how the use of on-device processing can protect the data collected without reducing the quality and responsiveness of the verification results.
- To discover the main challenges related to the practical application, security, and reliability aspects that gesture-CAPTCHA researchers and systems developers are facing at the moment.

Data and Signal Sources for Gesture-Based CAPTCHA and Liveness Detection

The systems which the authors describe only utilise a restricted set of signals listed below: RGB video taken through a web came is a primary tool for the computer vision methods of hand gesture recognition which are based on the identification of landmarks.

Also, the liveness is judged mostly by the motion and texture-based features of the video.

- Hand-landmark coordinates detected by a compact model on the client's side representing key points of the fingers and their joints.
- Inertial and gyroscope sensors data read by touch devices as an alternative source of information for gesture-based CAPTCHA, potentially usable in camera-less environments.

Motion characteristics of the detected landmarks, serving as temporal descriptors of gestures and separating authentic movements from synthetic ones.

Organization of the Paper

Let's first explain the structure of the rest of this review paper. Section 2 discusses related articles on CAPTCHA development, hand gesture recognition and liveness detection. Section 3 contains information of hand gesture recognition and anti-spoofing methods, together with taxonomies and comparison table. Then, Section 4 outlines the limitations and challenges of such system design. Finally, Section 5 discusses the future possibilities and Section 6 offers summary of a general overview of this review.

II. LITERATURE REVIEW

Initially, the CAPTCHA challenge problem emphasized the use of a difficult AI task to separate a human from an automatic program which then gave rise to a CAPTCHA for a human and extremely challenging for a computer from a computational point of view. [1]

However, later security research highlighted the vulnerability of this approach, which relied on perceptual cues: if computer vision capabilities advance, then reCAPTCHA-style challenges will become easier for machines to solve, undermining the design principle. One solution was demonstrated by a large-scale study of Google's reCAPTCHA, which showed that reCAPTCHA could be reliably bypassed using computer vision; this prompted a shift to looking at alternatives that require a more embodied interaction than selecting the correct checkbox. [2]

An early proposal for such a CAPTCHA, called GESTCHA, relied on the angular-velocity sensors present in most modern touch-enabled phones, extracting features from short bursts of motion and classifying them using Naive Bayes and Random Forests. The result was a gesture recognition challenge that could be solved more reliably and quickly than a reCAPTCHA v3 challenge, according to a user study of 850 participants, though it relied on phone hardware rather than a webcam. [3]

For webcam-based solutions, a real-time pose estimation pipeline was developed that tracks a full hand skeleton based on the pixels of an RGB camera, giving a twenty-one keypoint coordinate tracking system that does not require any depth-sensing hardware. [4]

Meanwhile, lightweight neural networks have been developed that can operate on such a pose-estimation track to classify gestures, either static or dynamic, thereby reducing the resource requirements for a gesture-based CAPTCHA to the computational power of an average smartphone. [5]

A hybrid approach to CAPTCHA design, also outside the realm of vision, was demonstrated in a CAPTCHA that used dynamically generated model-resistant text alongside keyboard typing biometrics to add another layer of authentication, but which served as a reminder that behavioral cues beyond gestures, such as typing rhythm, can serve as an alternative to visual bot detection. [6]

However, the security guarantees of any embodied CAPTCHA ultimately depend on its ability to resist spoofing attacks; one relevant area of research is facing anti-spoofing, which has seen a proliferation of methods designed to detect whether a face presented to a camera is a real face or an image or video of a face. A survey of the literature on face anti-spoofing organizes deep learning approaches to the problem into two-class methods, which learn to distinguish between real and fake faces, and one-class methods, which learn a model of real faces and attempt to reject inputs that do not match that model at test time. [7]

A similar survey on the topic highlights additional methods that rely on different physical properties to distinguish between real and spoofed faces, ranging from texture analysis to temporal information or remote photoplethysmography, and also notes the tendency for anti-spoofing models to perform poorly on inputs that differ substantially from the data on which they were trained, which would be relevant to any spoofing detection system attempting to operate on a public website. [8]

Meanwhile, recent research into self-supervised transformer networks for face anti-spoofing demonstrated that such architectures are able to learn features that are more informative for detecting spoofing attempts, providing a possible avenue for improving the security of gesture-based liveness detection. [9]

Ultimately, the academic literature on gesture-based CAPTCHA security has been validated by real-world practicality: in 2026, Google announced that it had begun testing a hand-based gesture recognition challenge, with the company's reCAPTCHA and fraud defense system asking users to perform a motion with their hands in front of their webcam before granting access to the requested service or transaction. The process extracts the twenty-one-point hand-landmark coordinate system and reportedly analyzes the video in real-time while not storing the video for later analysis (unless the user is flagged for further inspection), aligning with many of the design principles discussed in this review. [10]

However, almost immediately after the initial reports of this system, online security forums began discussing how the same hand-gesture challenge could be bypassed by using an ordinary photograph of a waving hand, demonstrating that such a motion tracking-based CAPTCHA could be defeated by a simple presentation attack. [11]

Literature on liveness detection often begins by describing the concept as an automated process for determining whether the claimed human user is physically present and actively engaging with the requested service, rather than using a photo or video in its place, and this definition highlights the importance of spoofing resistance in any biometric liveness detection challenge, including those based on vision-based gestures. [12]

III. METHODS AND TECHNIQUES FOR HAND GESTURE CAPTCHA

A hand-gesture CAPTCHA is a system with a pipeline from rawvideo capture up to a final binary human/bot decision that may involve liveness checks. For example, a bot may use pre-recorded video or images to mimic a gesture that the human user is asked to perform. These checks are designed so that a correct recognition of the intended gesture is not a sufficient condition to pass a CAPTCHA by just showing a video without any further verification.

Gesture Recognition Approaches

Reviewed systems recognize gestures using two broad families of technique, summarized in Figure 2.

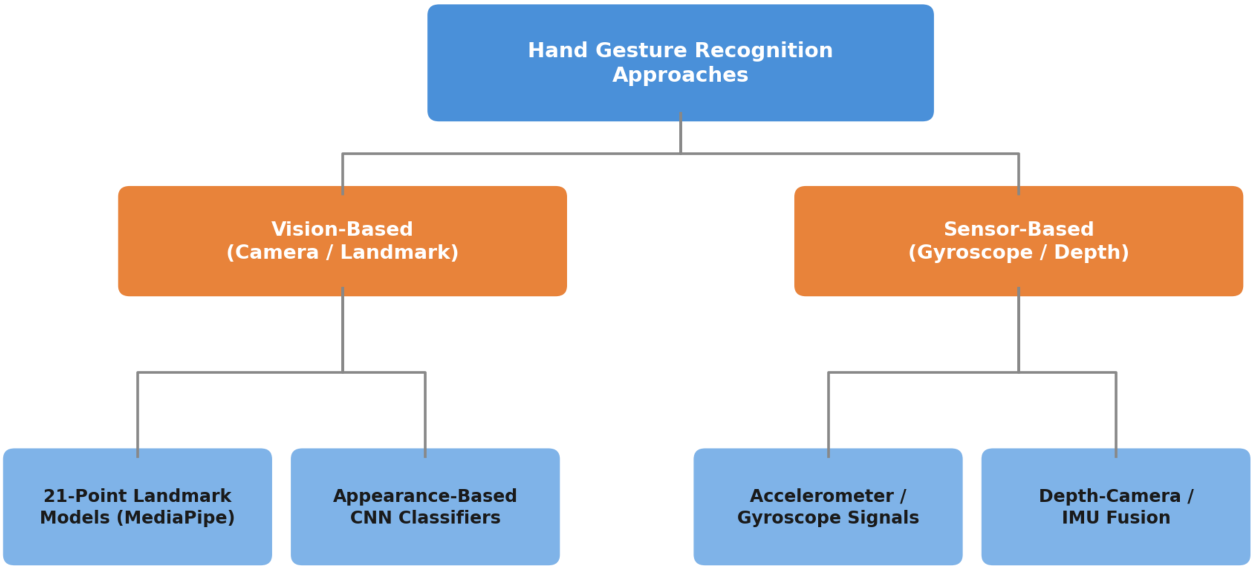

Fig. 2. Taxonomy of hand gesture recognition approaches.

Using landmarks, vision-based recognition is nowadays the primary and probably the most effective way for dealing with webcam CAPTCHAs. The method only takes a regular camera, and no other hardware is needed for that. A palm detector initially identifies the hand in the photo, and then a landmark model is employed to compute the 21 coordinate positions of the joints and knuckles. Based on the given coordinates, either a set of geometric rules can be applied, like counting extended fingers, and so, getting the gesture, or these coordinates can go through a classifier that recognizes gestures (the vocabulary is finite) [4], [5].

It is also the case with the convolutional classifiers, which are based solely on the appearance, but instead of the landmark coordinates, they classify the whole hand cropped image. Such models are helpful whenever landmarks become less distinguishable due to lighting or occlusion issues, although they involve somewhat heavier computational per frame. However, the way gesture identity is captured is entirely research says from the device's own sensor - accelerometer or gyroscope when the phone is held.
This eliminates any requirement for a camera and was even the concept behind the initial gesture-CAPTCHA prototype [3] but as you might guess, these methods are only limited to devices that are touch-enabled and so cannot be applied to a purely webcam-based challenge.

Liveness and Anti-Spoofing Techniques

Because gesture recognition alone does not prove that a hand is real the systems examined in this study each have least one (or more) liveness check to confirm that the object performing the gestures is real. Fig. 1 Shows these liveness checks grouped into four categories. Motion-based liveness detection evaluates the consistency of the movement path, velocity and fluidity of a gesture with normal human motion, which is why it is often chosen as the very first defence mechanism since it works with the landmark stream that was already obtained for feature recognition. On the contrary, texture and depth analysis aims to identify physical features which can only be found on the surface of a three-dimensional object that photographs or printed/printed-screens cannot duplicate, like fine skin textures, parallax (a thing in which a hand or object seems to change position when viewed from a different angle), or genuine 3D structure, which are techniques from face anti-spoofing research that have been directly borrowed, here for example, models with sensitivity to texture and depth have been employed to distinguish real face images from the fake ones that are either printed or screen-replayed only, [7], [8].

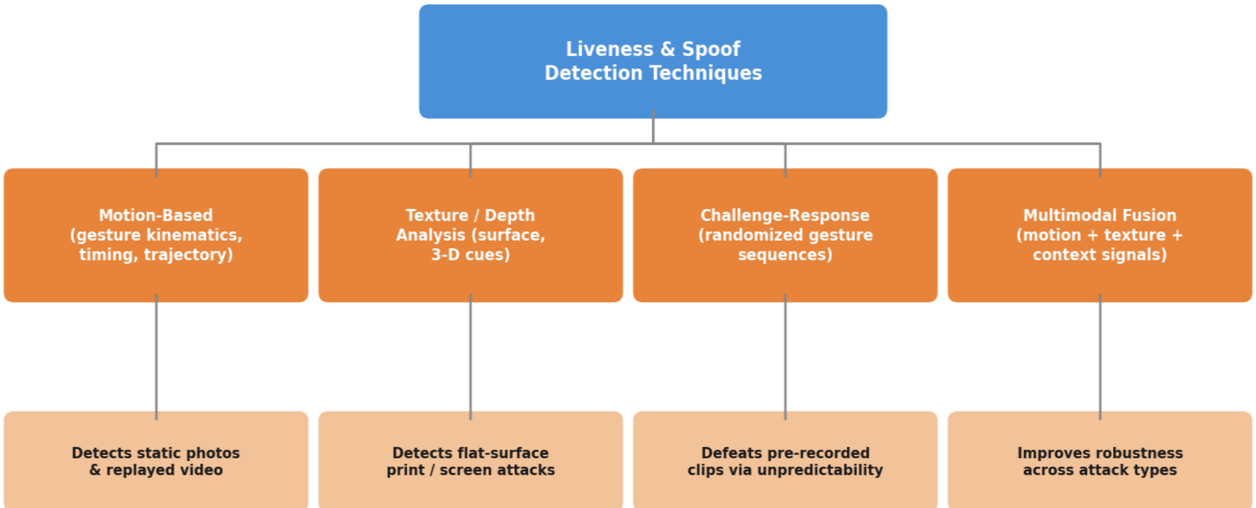

Fig. 3. Taxonomy of liveness-detection and anti-spoofing techniques applicable to gesture CAPTCHAs.

The challenge-response liveness method increases the vulnerability of video recordings attacking by mandating a random gesture per session so that a video on loop or image that does not move will not fit all challenges; this approach is based on the similar principle that prompted CAPTCHA generations to adopt randomized text distortion, i.e using CAPTCHAs [1], which is applied now to physical motion. Lastly, multimodal fusion is the integration of two or more of these signals such as checking both a motion sequence and a texture signal, this is mainly supported with evidence from research that anti-spoofing algorithms trained based on single type of cue or attack do perform very poorly when they encounter novel types of spoofing [8], whereas integrating different signals greatly improves the models and narrows the performance gap. In fact, a virtual camera was used to relay a static photo and bypassed the Google's hand-gesture reCAPTCHA in the real world which is a very good representation of what can happen if landmark-based recognition is used without an adequate liveness layer backing it up [11]

Comparative Summary of Reviewed Approaches

Table 1. Comparative summary of reviewed gesture-CAPTCHA and liveness-detection approaches.

Approach	Core Technique	Primary Signal	Strength	Limitation
GESTCHA (mobile)	Using naive bayes and Random Forest we go through gyroscope motion sensors.	Gyroscope angular velocity	850 participants were used in a usability study that was successful in solving much faster compared to reCAPTCHAv3	Calls for that you have a handheld device with an in-built gyroscope; it is also camera-free
MediaPipe Hand Landmarker gives accurate hand landmark detection and works both for palm/knuckle landmark detection and hand gesture classification via regression and CNN.	21-point palm/knuckle regression + CNN classifier	Works in live mode (a single camera) on an embedded device that has at least a camera and a CPU.	It only takes a few seconds to see if the hand is pointing at you or is behind you.	Landmark-onl y models are vulnerable to a well-posed static image

Google reCAPTCHA checks hand gestures: the system can detect if you're making a specific hand gesture like raising your right hand, closing one eye, or showing a peace sign using your fingers.	Facial feature detection + short-video liveness assessment: the algorithm identifies certain features of the face and then estimates if a short video is alive or not.	Color webcam movie (Google Cloud Fraud Protection)	It has been implemented at web scale; it removes the footage after the verification is done.	They could bypass the tests of their preliminary stages with a static photo displayed over a virtual webcam camera.
Hybrid of keystrokes combined with generative CAPTCHA	Challenge text generated by large language model and keystroke-dynamics classifier	Character rhythm through typing, rather than sight	Implements a biometric layer, behavioural aspect, that will be very hard to spoof or fake by malicious bots	Cannot be applied for interaction done solely with camera/no-touch
CNN and Transformer architectures are used in face anti spoofing.	Features that can be used for face anti spoofing are depth, rPPG and local texture.	Framed sequence of images representing a human face, i.e., a video	High performance inside the dataset; some of these methods implement the confidence-aware mechanism for the rejection	Past research shows a large drop-off in generalizability performance from in-distribution test samples to out-of-distribution ones. This study aims to better understand the relationship between in-distribution and out-of-distribution data with the help of deep adversarial generative networks

Privacy-Preserving Design

One of the main similarities identified was the desire for on-device computations: extracting landmarks and performing liveness detection where possible are the jobs of the device, so the only information that leaves the device is a brief sequence of location pairs or a binary signal, instead of a full video clip. Also, their own description about hand-gesture reCAPTCHA by Google is aligned with this approach. It was stated to be done in real time without any link to a user, and the footage gets destroyed after the verification [10].

This technique greatly reduces the possibility of biometric information being intercepted and stored over the internet which is a great security concern mainly with hand geometry as biometric identifiers can be derived from them just like with facial geometry.

IV. CHALLENGES AND LIMITATIONS

- Spoofer resistance in practical scenarios: the hand gesture production reCAPTCHA was shown to be easily bypassed through a still picture taken and a virtual-camera software. Without a strong, well-tested liveness detection module, extracting landmarks will not really be enough on its own [11].
- Cross-domain generalization anti-spoofing models are likely to work well in their dataset or attack type but accuracy drops when confronted by other spoofing methods in different camera types lighting conditions or even unseen spoofing methods [7] [8].
- Accessibility and inclusivity: a physical gesture challenge may be difficult or impossible for users with motor impairments, limited camera access, or low-bandwidth connections, so a fallback verification path is required.
- Computational and latency constraints: landmark extraction and liveness scoring must run in real time

inside a browser or mobile app, which limits how large or complex the underlying models can be.

- Privacy perception versus practice: even where footage is processed locally and deleted, requesting camera access for a routine sign-up or checkout flow raises user discomfort and trust concerns that are independent of the system's technical privacy guarantees [10].
- Evaluation standardization: unlike text-based CAPTCHA research, gesture-CAPTCHA studies do not yet share common benchmark datasets or attack suites, making it difficult to compare accuracy and spoof-resistance claims across papers.

V. FUTURE DIRECTIONS

- Depth and multi-frame cues: incorporating parallax, depth estimation, or multi-frame consistency checks to directly counter single-image and looping-video spoofing attacks.
- Randomized, multi-step challenges: chaining two or more randomly selected gestures within one session to sharply increase the cost of a pre-recorded or generative spoofing attempt.
- Federated or on-device model improvement: updating recognition and liveness models from aggregated, anonymized signals rather than centrally collected video, extending the privacy-preserving philosophy from inference to training.
- Standardized spoof-attack benchmarks: publishing shared datasets of genuine and spoofed gesture attempts, comparable to existing face anti-spoofing benchmarks, so that accuracy and robustness claims become directly comparable across studies.
- Accessible fallback verification: pairing gesture challenges with alternative, equally secure verification paths for users who cannot complete a camera-based check.

VI. CONCLUSION

Hand-gesture verification is one of the most encouraging developments that could come back to stop text- and image-based CAPTCHA's effectiveness being lowered, through presenting the user a person that would perform the challenge easily, and at the same time, making it so that a program could not do that even in theory. The harder and less settled problem is liveness: the real-world bypass of a deployed hand-gesture reCAPTCHA using nothing more than a stock photo and virtual-camera software shows that recognition without robust spoof detection provides a false sense of security. Progress will depend on combining motion, texture, and depth-based liveness cues, testing against standardized spoof-attack benchmarks, and keeping video processing on the user's own device so that stronger security does not come at the cost of user privacy.

REFERENCES

[1] L. Von Ahn, M. Blum, N. J. Hopper, and J. Langford, “CAPTCHA: Using hard AI problems for security,” in *Advances in Cryptology—EUROCRYPT 2003*, pp. 294–311, Springer, 2003.

[2] S. Sivakorn, J. Polakis, and A. D. Keromytis, “I'm not a human: Breaking the Google reCAPTCHA,” Black Hat ASIA, pp. 1–12, 2016.

[3] A. I. Pritom, M. A. Al Mashuk, S. Ahmed, N. Monira, and M. Z. Islam, “GESTCHA: A gesture-based CAPTCHA design for smart devices using angular velocity,” Multimedia Tools and Applications, vol. 82, no. 1, pp. 521–549, 2023, doi: 10.1007/s11042-022-13272-6.

[4] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, and M. Grundmann, “MediaPipe Hands: On-device real-time hand tracking,” arXiv:2006.10214, 2020.

[5] M. S. Abdallah, G. H. Samaan, A. R. Wadie, F. Makhmudov, and Y.-I. Cho, “Light-weight deep learning techniques with advanced processing for real-time hand gesture recognition,” Sensors, vol. 23, no. 1, Art. no. 2, 2023, doi: 10.3390/s23010002.

[6] A. Aghaei Nia, “A hybrid CAPTCHA combining generative AI with keystroke dynamics for enhanced bot detection,” arXiv:2510.02374, 2025.

[7] Z. Yu, Y. Qin, X. Li, C. Zhao, Z. Lei, and G. Zhao, “Deep learning for face anti-spoofing: A survey,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 45, no. 5, pp. 5609–5631, 2023.

[8] P.-K. Huang, J.-X. Chong, M.-T. Hsu, F.-Y. Hsu, C.-H. Chiang, T.-H. Chen, and C.-T. Hsu, “A survey on deep learning-based face anti-spoofing,” APSIPA Transactions on Signal and

Information Processing, vol. 13, no. 1, Art. no. e34, 2024.

[9] A. Keresh and P. Shamoi, “Liveness detection in computer vision: Transformer-based self-supervised learning for face anti-spoofing,” arXiv:2406.13860, 2024.

[10] Biometric Update, “Google brings liveness detection to anti-bot verification,” Jun. 22, 2026.

[11] Tom's Hardware, “Google testing controversial webcam-based reCAPTCHA that asks for a hand scan to prove you're human—testers beat it with a stock photo,” Jul. 2, 2026.

[12] Wikipedia contributors, “Liveness test,” *Wikipedia*, 2026.