Journal of Neural Engineering

PAPER

OPEN ACCESS

RECEIVED
24 November 2025

REVISED
1 February 2026

ACCEPTED FOR PUBLICATION
23 February 2026

PUBLISHED
5 March 2026

Fuzzing the brain: automated stress testing for the safety of ML-driven neurostimulation

Mara Downing[1,*], **Matthew Peng**[1], **Jacob Granley**[1], **Michael Beyeler**[1,2] and **Tevfik Bultan**[1]

[1] Department of Computer Science, University of California, Santa Barbara, CA, United States of America
[2] Department of Psychological & Brain Sciences, University of California, Santa Barbara, CA, United States of America
[*] Author to whom any correspondence should be addressed.

E-mail: maradowning@cs.ucsb.edu

Keywords: coverage-guided fuzzing, neural networks, safety constraints, biomedical implants, neuroprostheses

Abstract

Objective. Machine learning (ML) models are increasingly used to generate electrical stimulation patterns in neuroprosthetic devices such as visual prostheses. While these models promise precise and personalized control, they also introduce new safety risks when model outputs are delivered directly to neural tissue. We propose a systematic, quantitative approach to detect and characterize unsafe stimulation patterns in ML-driven neurostimulation systems. *Approach.* We adapt an automated software testing technique known as coverage-guided fuzzing to the domain of neural stimulation. Here, fuzzing performs stress testing by perturbing model inputs and tracking whether resulting stimulation violates biophysical limits on charge density, instantaneous current, or electrode co-activation. The framework treats encoders as black boxes and steers exploration with coverage metrics that quantify how broadly test cases span the space of possible outputs and violation types. *Main results.* Applied to deep stimulus encoders for the retina and cortex, the method systematically reveals diverse stimulation regimes that exceed established safety limits. Two violation-output coverage metrics identify the highest number and diversity of unsafe outputs, enabling interpretable comparisons across architectures and training strategies. *Significance.* Violation-focused fuzzing reframes safety assessment as an empirical, reproducible process. By transforming safety from a training heuristic into a measurable property of the deployed model, it establishes a foundation for evidence-based benchmarking, regulatory readiness, and ethical assurance in next-generation neural interfaces.

1. Introduction

Machine learning (ML) is rapidly transforming neuroengineering by enabling adaptive encoding and decoding of neural activity in systems that restore or augment human function. In visual prostheses (Fernandez 2018, Ayton *et al* 2020), deep neural networks have been proposed to translate camera images into electrical stimulation patterns delivered to the retina or cortex (de Ruyter van Steveninck *et al* 2022, Granley *et al* 2022a, 2023, Moure *et al* 2025). These networks implement the inverse of a forward model (Chen *et al* 2009, Beyeler *et al* 2019, Granley and Beyeler 2021, Granley *et al* 2022b, van der Grinten *et al* 2024), which maps electrical stimulation to predicted neural or perceptual responses. Inverting this mapping yields a *stimulus encoder* that transforms a desired percept (or its visual proxy) into the per-electrode stimulation patterns expected to elicit it. Learned stimulus encoders are now being explored in early-stage clinical evaluations (Moure *et al* 2025) and are central to designs for next-generation prosthetic vision systems (Beyeler and Sanchez-Garcia 2022, Grani *et al* 2022, 2025). Because they would prescribe electrical stimuli in real time, their outputs must adhere to established limits on charge density, instantaneous current, and active electrode count (Park and Han 2018). Ensuring adherence to these constraints is therefore a prerequisite for clinical translation.

However, the safety of stimulus encoder systems remains critically understudied. Typically, firmware

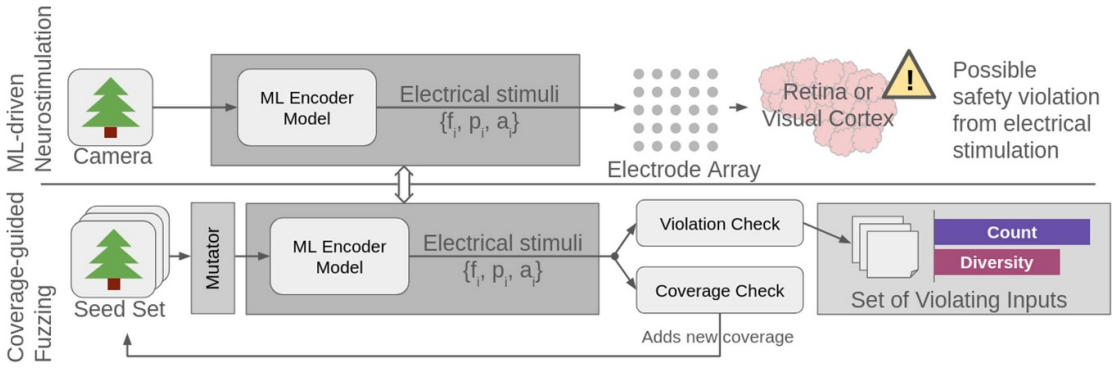

Figure 1. Overview of our framework for discovering safety violations in ML-driven neurostimulation. *Top:* Visual prostheses use deep nets to convert camera input into electrical stimuli applied to the brain. Outputs must satisfy neurobiological constraints; violations may occur even under normal input. *Bottom:* Our coverage-guided fuzzer mutates inputs to explore model behavior, using coverage and violation checks to uncover diverse unsafe outputs. The resulting violations enable quantitative safety evaluation and model comparison.

or hardware safeguards built into the clinical system are relied upon to clip, rescale, or drop unsafe stimuli (Second Sight 2013). While this prevents immediate harm to the user, it obscures whether the underlying encoder performed safely, and prevents iterative refinement to improve model safety while maintaining performance. The limited existing work in this area has aimed to reduce violations by penalizing unsafe stimuli during training (Küçükoğlu *et al* 2025), but there remains a lack of tools that clinicians and researchers can use to systematically validate that model-generated stimuli adhere to safety constraints.

Here we introduce an automated stress-testing framework for evaluating the safety of ML-driven neurostimulation. The approach adapts coverage-guided fuzzing (CGF) (Chen *et al* 2018) to probe encoders for unsafe output regimes. In this setting, fuzzing perturbs input images while monitoring whether the resulting stimulation exceeds predefined limits on charge density, instantaneous current, or the number of active electrodes. Exploration is directed by coverage signals that quantify how broadly the perturbations probe the encoder's output space and its proximity to safety boundaries, enabling the systematic discovery of rare but clinically meaningful failure modes. CGF, in brief, tracks which perturbed inputs cause new behavior and emphasizes those for further mutation, pushing exploration of new behaviors of the model. Figure 1 shows an overview of our CGF framework applied to an ML-driven neurostimulation encoder.

To guide this process, we introduce two complementary output-space coverage metrics. The first, Violation-Output K-Multisection Violation Proportion (VO-KMVP), prioritizes tests that push stimulation parameters toward their physiological limits, revealing the inputs that provoke the most severe violations. The second, Violation-Output K-Multisection Output Coverage (VO-KMOC), measures how broadly the test exercises the range of possible stimulation patterns across electrodes, emphasizing the diversity of violation types and spatial distributions. Together, these metrics characterize both the frequency and the breadth of unsafe behaviors.

We demonstrate this framework on state-of-the-art stimulus encoders for retinal and cortical prostheses (Granley *et al* 2023, van der Grinten *et al* 2024), which were trained to optimize perceptual fidelity but not explicitly constrained for safety. The stress test uncovers over-limit stimulation patterns that conventional testing does not effectively discover, offering quantitative insights that can guide model selection, retraining, and firmware policy. We then demonstrate how CGF can be used in conjunction with performance metrics to evaluate the safety improvements from different regularization strategies (Küçükoğlu *et al* 2025), showing its value as a tool to inform model selection and refinement.

Although our experiments focus on artificial vision, the same principles apply to any neural interface where an ML model prescribes electrical stimulation under biophysical constraints, including next-generation deep brain, spinal, and vagus nerve stimulators (Shenoy and Carmena 2014, Okorokova *et al* 2018, Rao 2019, Drakopoulos and Verhulst 2023). By framing safety evaluation as output-level verification and validation, coverage-guided stress testing offers a generalizable foundation for developing safer and more trustworthy ML-based neurotechnologies.

2. Methods

Our goal is to systematically test whether a trained stimulus encoder ever produces stimulation parameters that exceed established biophysical limits. To do so, we adapt a software testing strategy called CGF (Chen *et al* 2018) to the domain of neurostimulation. In conventional software testing, fuzzing automatically perturbs program inputs to uncover

rare failure modes; here, it perturbs sensory inputs (e.g., images) to expose conditions under which an ML encoder produces unsafe stimulation. This allows the model to be evaluated in a *black-box* fashion (i.e. no internal weights or gradients are needed) and complements the usual forward simulations or loss-based analyses used in model development.

We focus on regression models that map sensory input $\mathbf{x} \in \mathbb{R}^d$ (where d indicates the number of dimensions in the input) to a vector of stimulation parameters $\mathbf{y} = M(\mathbf{x})$. In a neurostimulation system with $|\mathcal{I}|$ electrodes, the model output can be expressed as $\mathbf{y} = \{f_i, p_i, a_i\}_{i=1}^{|\mathcal{I}|}$, where f_i denotes pulse frequency, p_i the pulse duration, and a_i the amplitude of a biphasic square-wave pulse train delivered by electrode i. Safety is characterized by a set of inequality constraints $\{V_k(\mathbf{y}) \leqslant 0\}_{k=1}^{K}$, each corresponding to a physiological limit (e.g., maximum charge density, instantaneous current, or co-activation area). An input $\mathbf{x}$ constitutes a *violation input* if its output violates at least one constraint.

Formally, we aim to discover a large and diverse set of inputs

$$\mathcal{V} = \{\mathbf{x} \mid \exists k : V_k(M(\mathbf{x})) > 0\}, \tag{1}$$

subject to the domain-specific constraints $V_k(\cdot) \leqslant 0$. Each V_k may apply globally (e.g. total current across all electrodes) or locally (e.g. per-electrode charge density), as detailed in section 2.1.

Because a single violation type can dominate the search, we guide exploration using coverage metrics that favor both the discovery of new violations and the diversification of test cases (section 2.3). This balance ensures that the framework not only maximizes the number of unsafe cases found but also explores the search space, providing actionable insight for model redesign or retraining.

2.1. Safety constraints for electrode-based neurostimulation

Electrical stimulation delivered through implanted electrodes must obey strict biophysical limits to prevent tissue damage and patient discomfort. Typical devices control three parameters per electrode (i.e. the pulse frequency f_i, duration p_i, and amplitude a_i of charge-balanced biphasic pulse trains), and safe operation requires that each combination remain within established physiological and device-specific bounds. Our framework treats these limits as formal constraints on the outputs of a model and identifies any violation of them as a potential safety risk.

We categorize violations into two broad types. *Aggregate* violations occur when a property of the stimulation pattern as a whole exceeds a system-wide limit, such as total instantaneous current across all electrodes. *Electrode-wise* violations occur when a single channel violates a local constraint, such as charge density or pulse timing. Formally, we express these as inequalities over the model's output vector $\mathbf{y}$: a configuration is safe when all constraints $V_k(\mathbf{y}) \leqslant 0$ are satisfied and unsafe when at least one $V_k(\mathbf{y}) > 0$.

Within this schema, we define four clinically motivated safety constraints representative of real retinal and cortical prostheses:

- *Physically impossible stimulus:* Each biphasic pulse must fit within its temporal period defined by its frequency f_i (Hz). When the pulse duration p_i (ms) becomes too long to complete a full cycle, the pulse is physically infeasible:

$$V_{\text{PI}} = 2p_i - \frac{1000}{f_i}. \tag{2}$$

- *Charge density limit:* To avoid electrochemical damage at the electrode-tissue interface, the delivered charge per electrode must remain below a device-specific limit. For epiretinal implants such as the Argus II, this limit is specified in the surgical manual (Second Sight 2013) as a *per-electrode* maximum charge (derived from the FDA charge-density threshold and the device's electrode geometry). Accordingly, we treat the product of pulse duration p_i and amplitude a_i as a per-electrode charge quantity that must not exceed the published limit ϵ_1:

$$V_{\text{CD}} = p_i a_i - \epsilon_1, \tag{3}$$

  where a positive value indicates a violation.
- *Instantaneous current limit:* The total instantaneous current across all electrodes $\mathcal{I}$ must stay below a device-level ceiling ϵ_2 (μA), ensuring hardware stability and avoiding unintended current spread:

$$V_{\text{IC}} = \sum_{i=1}^{|\mathcal{I}|} a_i - \epsilon_2. \tag{4}$$

- *Active electrode limit:* The number of simultaneously active electrodes must remain below ϵ_3 to minimize crosstalk and power consumption (here $[\cdot]$ denotes the Iverson bracket, equal to 1 if the condition inside is true and 0 otherwise):

$$V_{\text{AE}} = \sum_{i=1}^{|\mathcal{I}|} [a_i > 0] - \epsilon_3. \tag{5}$$

In all cases here, a positive value ($V > 0$) denotes a violation. The specific values used for ϵ_1, ϵ_2, and ϵ_3 were derived separately for retinal and cortical prostheses using published literature and FDA specifications in conjunction with consultation with clinical experts (Second Sight 2013, Fernández and Normann 2016, Fernandez 2018, Chen *et al* 2020, Fernández *et al* 2021, U.S. Food and Drug Administration n.d.).

Algorithm 1. Fuzz(S, P)

▷Performs fuzzing on the model to detect safety violations.
▷Calls function PREPROCESS() which computes expected ranges for nodes or output values if required by the coverage metric, function COV() which returns the model coverage as a value between 0 and 1, and function TESTMUTANTS() which is described in Algorithm 2.

Input: S: seed set and P: optional set of input data for pre-processing.
Output: $\mathcal{V}$: set of violating inputs.

```
(P)                                          ▷Computed values are stored globally
C ← COV(S)
V ← ∅
numberOfTests ← |S|
while numberOfTests < testLimit do
    (S, V, C) ← TESTMUTANTS(S, V, C, m)      ▷Generates and tests m mutants, Algorithm 2
    numberOfTests ← numberOfTests + m
end while
return V
```

Although these expressions are taken from visual prosthesis designs, the same formulation applies to any electrode-based neurotechnology (e.g., cochlear, spinal, deep brain, or vagus nerve stimulators) where continuous control of amplitude, frequency, and pulse width must remain within safe biophysical limits (McCreery *et al* 1990, Shannon 1992, Grill and Mortimer 1995, Cameron 2004). Defining safety directly in terms of model outputs allows our framework to evaluate encoder models in a black-box manner, independent of input type or behavioral context.

2.2. Coverage-Guided Fuzzing (CGF)

In traditional software testing, fuzzing repeatedly perturbs program inputs to uncover rare failures such as crashes. Here, CGF serves as an *automated stress test*: the algorithm perturbs sensory inputs (e.g. camera images), observes the resulting stimulation patterns, and records any cases that violate the safety constraints defined in section 2.1. This process requires no access to model internals, making it well-suited for validation of proprietary or closed-source encoders.

A coverage function $\mathrm{Cov}(T) \in [0, 1]$ quantifies how much of the model's behavioral space has been explored by a set of test inputs T. Coverage can be based on different signals (e.g. internal activations, output statistics, violation distributions; detailed in section 2.3), but the goal is the same: higher coverage means a broader sampling of possible model behavior. The fuzzer begins with a seed set S of initial test images which are iteratively mutated to produce new tests. A new test input $\mathbf{x}'$ is added to S only if it increases coverage, that is, when $\mathrm{Cov}(S \cup \{\mathbf{x}'\}) > \mathrm{Cov}(S)$. In this way, the algorithm automatically steers exploration toward novel and potentially unsafe regions of model behavior.

2.2.1. Fuzzing strategy

The high-level procedure is summarized in algorithm 1. Before fuzzing, an optional preprocessing step estimates the expected range of input or output values, if required by the coverage metric. The algorithm then enters an iterative loop: it selects seed images, applies random perturbations ('mutations'), evaluates the resulting model outputs, and updates both the coverage and the list of discovered violations as necessary.

2.2.2. Mutation strategy

Each new test input is generated by applying a random image-level transformation to a seed example, following image transformations from and procedures similar to DeepHunter (Xie *et al* 2019). Transformations include translation, rotation, scaling, shearing, brightness or contrast adjustment, blurring, additive noise, and pixel-level perturbation. At each iteration, the algorithm:

1. selects a seed $\mathbf{x} \in S$ for mutation, weighted by how often it has previously led to new violations (equation (6)),
2. applies a random transformation to create $\mathbf{x}'$ and obtain $\mathbf{y}' = M(\mathbf{x}')$,
3. checks whether $\mathbf{y}'$ violates any safety constraint $V_k(\mathbf{y}')$ and, if so, records $\mathbf{x}'$ in $\mathcal{V}$,
4. evaluates whether $\mathbf{x}'$ increases coverage; if yes, it is added to the seed set S for further exploration.

This procedure, summarized in algorithm 2, repeats for a fixed number of mutations per seed ($m = 10$ in our experiments), progressively building a diverse collection of unsafe examples while exploring the available search space. We choose m as 10 to allow for reasonable exploration of each chosen seed without overwhelming the seed set with mutations of

Algorithm 2. TestMutants(S, $\mathcal{V}$, C, m)

▷Generates m mutant images from seed s, checks violations, and adds each one to the seed set if coverage is increased.
▷Calls function CHOOSE() which chooses a seed as described in equation (6), function MUTATE() which chooses a mutation at random, applies it, and returns the new image, function COV() which returns the model coverage as a proportion between 0 and 1, and function VIOLATES() which returns a boolean indicating whether or not a test produces a violation.

Input: S: seed set and $\mathcal{V}$: set of violating inputs found. C: current proportion of coverage using existing tests in S. m: number of mutants to generate.
Output: S: new seed set (may be unchanged), $\mathcal{V}$: new list of violations (may be unchanged), and C: current proportion of coverage using existing tests in S.

$s \leftarrow$ CHOOSE(S) ▷s is chosen from S
for 1 to m **do**
  $\mathbf{x}' \leftarrow$ MUTATE(s)
  if COV($S \cup \mathbf{x}'$) $> C$ **then**
    $S \leftarrow S \cup \mathbf{x}'$
    $C \leftarrow$ COV(S)
  end if
  if VIOLATES($\mathbf{x}'$) **then**
    $\mathcal{V} \leftarrow \mathcal{V} \cup \mathbf{x}'$
  end if
end for
return $(S, \mathcal{V}, C)$

one seed early in testing. The following equation controls how seeds are weighted for selection at each iteration of the algorithm:

$$P(s) = \begin{cases} 1 - g(s)/\gamma, & \text{if } g(s) > 1 - p_{\min}\gamma, \\ p_{\min}, & \text{otherwise,} \end{cases} \quad (6)$$

where $P(s)$ is the probability of selecting seed s, $g(s)$ counts its prior selections, γ scales sampling frequency, and $p_{\min}$ prevents any seed from being permanently ignored. This equation is adapted from DeepHunter (Xie *et al* 2019).

2.3. Coverage metrics

Effective CGF requires a feedback signal that reflects how much of a model's behavior has been explored. This feedback is called *coverage*. In conventional software testing, coverage often counts which lines of code were executed by a test set. For neural networks, prior work has used neuron activations as a stand-in for lines of code (Pei *et al* 2017, Ma *et al* 2018), but such internal signals often fail to correlate with meaningful conclusions about model *outputs* (Li *et al* 2019, Dong *et al* 2020, Yang *et al* 2022, Huang *et al* 2024).

In the context of neural stimulation, a good coverage metric should encourage the fuzzer to generate new tests that reveal *distinct and physiologically relevant* stimulation patterns—those that either approach the boundaries of safe operation or differ meaningfully in their output configuration. Without such a signal, the fuzzer would produce redundant test cases or fail to uncover rare unsafe conditions.

To systematically investigate which coverage strategies best uncover safety violations, we evaluate eleven metrics grouped into three conceptual families:

- *Basic strategies:* simple heuristics that use no coverage signal. They serve as baselines, measuring the effect of naive approaches to utilizing mutations.
- *Neuron coverage metrics:* white-box approaches that track how many internal neurons are activated by a test. These methods, adapted from software fuzzing for image classifiers, provide a historical reference but are not aligned with safety outcomes.
- *Violation-focused metrics:* new black-box metrics we introduce that operate directly on model inputs and outputs, guiding the search toward diverse and physiologically meaningful safety violations.

Table 1 summarizes all eleven metrics, which are described in detail below. The upper categories list the basic and neuron-based metrics used for comparison, while the lower category presents our six proposed violation-focused metrics.

2.3.1. Design rationale

Our goal is not to invent arbitrary metrics, but to span the most plausible design space for coverage in this domain. We systematically explored metrics that operate in three spaces relevant to an encoder model:

- *Input space:* encouraging diverse sensory inputs,
- *Feature space:* encouraging diversity in latent features,

Table 1. Fuzzing coverage metrics used in this paper.

Basic strategies:	
B-N	Mutates user-provided seeds but does not add new tests to the seed set.
B-A	Mutates and adds all new tests to the seed set.
B-FR	Uses fully random images without mutation or a seed set.
B-local	Perturbs the seed with the most violations locally to generate many similar tests.
Neuron coverage metrics (white-box):	
N-NC	Activates neurons exceeding a fixed threshold (Pei *et al* 2017).
N-KMNC	Partitions each neuron's output into K bins and tracks which are activated (Ma *et al* 2018).
N-NBC	Tracks activations above or below neuron-specific bounds from training data (Ma *et al* 2018).
N-SNAC	Tracks activations that exceed the maximum seen in training data (Ma *et al* 2018).
N-TKNC	Tracks top-K most activated neurons per layer (Ma *et al* 2018).
Novel violation-focused coverage metrics (black-box):	
VO-KMVP	Bins the proportion of violation severity (including no violation) for each constraint.
VO-KMOC	Bins each output dimension across the test set.
VO-KMVP-V	Like VO-KMVP, but only considers proportions which indicate a violation.
VO-VCC	Tracks which constraints have been violated at least once.
I-KMIC	Bins each pixel's value range across the input.
I-Div-approx	Bins feature space from an autoencoder to approximate test diversity.

- *Output and violation space:* encouraging exploration of stimulation patterns and safety limits.

This principled organization ensures that our proposed metrics cover every meaningful axis along which coverage-guided exploration might improve safety testing.

2.3.2. Violation-focused metrics (our approach)

Among the new metrics, two perform consistently best and form the core of our framework: VO-KMVP and VO-KMOC. Both metrics quantify how much of the model's output space has been explored in ways that are relevant to safety—either by testing the *severity* of constraint violations (VO-KMVP) or the *diversity* of stimulation outputs (VO-KMOC).

VO-KMVP quantifies how thoroughly the tests explore the range of each safety constraint. For a given output vector $\mathbf{y}$, each safety constraint V_k can be expressed as an inequality $V_k(\mathbf{y}) = \alpha(\mathbf{y}) - c \leqslant 0$, where $\alpha(\mathbf{y})$ is a biophysical quantity of interest (for example, charge density or total current) and c is the physiological limit of that quantity. The ratio $\alpha(\mathbf{y})/c$ is therefore a dimensionless *violation proportion*: values below 1 correspond to safe stimulation, while values at or above 1 indicate a violation.

Two types of constraints are considered (see section 2.1): aggregate constraints V_A that depend on all electrodes jointly and electrode-wise constraints V_E that apply separately to each electrode $i \in \mathcal{I}$. For each type, the violation proportions are divided into K equal-width bins over a range $[\min, \max]$, and a bin is considered 'covered' once at least one test has produced a value in that bin's range. The coverage of a test set S is then defined as

$$\mathrm{COV}(S) = \frac{\mathrm{PART}(V_\mathrm{A}) + \mathrm{PART}(V_\mathrm{E})}{K \times |V_\mathrm{A}| + K \times |V_\mathrm{E}| \times |\mathcal{I}|}, \tag{7}$$

where

$$\mathrm{PART}(V_\mathrm{A}) = \sum_{v \in V_\mathrm{A}} \sum_{k=0}^{K-1} \times \left[\exists \mathbf{x} \in S : \min_k \leqslant \frac{\alpha_v(\mathbf{y})}{c} < \min_{k+1} \right],$$

$$\mathrm{PART}(V_\mathrm{E}) = \sum_{v \in V_\mathrm{E}} \sum_{i=1}^{|\mathcal{I}|} \sum_{k=0}^{K-1} \times \left[\exists \mathbf{x} \in S : \min_k \leqslant \frac{\alpha_{i,v}(\mathbf{y})}{c} < \min_{k+1} \right].$$

Here $[\cdot]$ denotes the Iverson bracket (equal to 1 if the condition inside is true and 0 otherwise), and $\min_k = \min + k(\max - \min)/K$ defines the lower edge of bin k. Values below the minimum or above the maximum are assigned to the outermost bins. Intuitively, this metric rewards new tests that drive stimulation parameters closer to the safety boundary, helping the fuzzer find the most severe violations.

VO-KMOC complements VO-KMVP by measuring how broadly the tests explore the model's output space, irrespective of whether they cause violations. Let $\mathcal{O}$ denote the set of output dimensions (e.g. all

electrode amplitudes, frequencies, and pulse widths). For each output dimension $o \in \mathcal{O}$, the observed range between the minimum lo_o and maximum hi_o values across a profiling dataset is divided into K bins with boundaries $lo_{o,k} = lo_o + k(hi_o - lo_o)/K$. A bin is marked as covered if at least one test input $\mathbf{x}$ produces an output $\mathrm{val}(\mathbf{x}, o)$ within that interval:

$$\mathrm{COV}(S) = \frac{\sum_{o\in\mathcal{O}} \sum_{k=0}^{K-1} [\exists \mathbf{x} \in S : lo_{o,k} \leqslant \mathrm{val}(\mathbf{x}, o) < lo_{o,k+1}]}{K \times |\mathcal{O}|}. \quad (8)$$

This metric rewards exploration of new amplitude, frequency, or pulse-duration ranges and helps identify diverse but safe operating points. Together, VO-KMVP and VO-KMOC balance *depth* (i.e. how far into dangerous territory the model can go) and *breadth* (i.e. how widely it explores the possible stimulation space). Because both metrics operate directly on physical output values, their results can be interpreted in clinically meaningful units (e.g. microamperes or microcoulombs/cm^2).

2.3.2.1. Additional violation-focused metrics
The remaining novel metrics (VO-KMVP-V, VO-VCC, I-KMIC, and I-Div-Approx) extend this logic to alternative forms of diversity or constraint specificity. Their conceptual roles are listed in table 1, and their formal definitions are provided in appendix. All coverage scores are computed between 0 and 1, with higher values indicating more complete exploration of the model's input-output-violation space.

2.3.3. Neuron-coverage metrics (white-box baselines)
White-box coverage metrics quantify how thoroughly a test set activates the *internal neurons* of a model. They are termed *white-box* because they require direct access to the model's internal activations, analogous to inspecting which lines of code were executed during a software test (Pei *et al* 2017, Ma *et al* 2018). In contrast, our proposed violation-focused metrics operate in a *black-box* setting, relying only on model inputs and outputs.

We implemented five representative white-box metrics from prior work in deep neural network testing and verification (Pei *et al* 2017, Ma *et al* 2018, Li *et al* 2019, Dong *et al* 2020, Yang *et al* 2022). These metrics remain the most widely used baselines in the literature and therefore provide a meaningful comparison for our black-box, safety-driven approach. Their names and conceptual roles are summarized in table 1, and their mathematical definitions are given in appendix for completeness.

In essence, these metrics assess how much of a model's internal computation has been exercised by the current test set:

- *Neuron Coverage* (Pei *et al* 2017): counts how many neurons become active above a fixed threshold at least once.
- *K-multisection Neuron Coverage (KMNC)* (Ma *et al* 2018): divides each neuron's activation range into K bins and measures which bins are hit, promoting exploration of intermediate activations.
- *Neuron boundary coverage (NBC)* and *Strong neuron activation coverage (SNAC)* (Ma *et al* 2018): reward tests that drive neurons below or above their activation limits observed during training.
- *Top-K Neuron Coverage (TKNC)* (Ma *et al* 2018): measures how often each neuron ranks among the K most active units within its layer.

Together, these methods represent the current state of white-box testing in machine learning and remain important historical baselines. They test whether increasing the diversity of internal activations correlates with externally observable safety violations. However, as shown in prior analyses (Li *et al* 2019, Dong *et al* 2020, Yang *et al* 2022, Huang *et al* 2024), Neuron Coverage has limited predictive value for real-world robustness, highlighting the need for output-level, biophysically grounded metrics like those introduced here.

2.4. Measurement of violation diversity

Simply tallying discovered violations gives a false sense of progress. Consider a model that outputs a pulse width larger than the inverse of its frequency so the pulse cannot physically exist. A mutation strategy can be to find one such input and then generate thousands of tiny variations that all trigger the same impossible pulse. The result is a huge violation count, but no insight into the extent of violations across model behaviors. To address this, we measure not only *how many* violations are found but *how* those violations are *distributed* in the input feature space and across the electrode array, so we can tell whether failures are concentrated, trivial to reproduce, or genuinely diverse and actionable.

2.4.1. Input-feature diversity (geometric diversity)
Aghababaeyan *et al* (2023) proposed several image-feature-based diversity metrics for analyzing classifier models and showed that they correlate more strongly with test quality and misclassification discovery than Neuron Coverage alone. Among these, geometric diversity (GD) was found to perform best. It measures how spread out a set of images is in a deep feature space extracted from a pretrained convolutional network.

Following this approach, we compute feature vectors for each violating input image using the VGG16 model (Simonyan and Zisserman 2014). Let $\mathcal{F}$ denote the set of all extracted feature vectors, and let $A_{\mathcal{F}}$ represent the matrix formed by stacking these vectors.

The GD is defined as the determinant of the Gram matrix $A_{\mathcal{F}}A_{\mathcal{F}}^{\mathrm{T}}$:

$$\mathrm{GD} = \det\left(A_{\mathcal{F}}A_{\mathcal{F}}^{\mathrm{T}}\right). \tag{9}$$

A higher determinant indicates that the feature vectors are more linearly independent, implying that the violating inputs occupy a broader region of feature space.

2.4.2. Violation-space diversity

While GD captures diversity in the visual input space, it does not account for how violations are distributed across the stimulation electrodes. For prosthetic devices, it is often more informative to ask whether violations are localized to specific electrodes or distributed across the array. We therefore introduce a complementary metric, violation-space diversity, which measures how violations vary across electrodes and violation types.

For each electrode $i \in \mathcal{I}$ and each electrode-specific constraint (here V_{PI} and V_{CD}), we define a *degree of violation intensity*:

$$\mathrm{degree}_i = \begin{cases} 0 & \alpha_i(\mathbf{y})/c - 1 \leqslant 0, \\ \alpha_i(\mathbf{y})/c - 1 & \text{otherwise.} \end{cases} \tag{10}$$

This quantity equals zero for safe electrodes and increases with the severity of the violation. Because the remaining constraints (V_{IC} and V_{AE}) operate globally rather than per electrode, they are not included directly. However, both relate to electrode amplitudes, so we also include the raw amplitudes a_i in the analysis.

For each test, we construct a concatenated vector of normalized values $\mathbf{v}_i = [\mathrm{degree}_{\mathrm{PI},i}, \mathrm{degree}_{\mathrm{CD},i}, a_i]$ across all electrodes i, resulting in a violation-space matrix A_{VS} of size $|\mathcal{I}| \times 3$. We then measure the overall spread of these vectors across tests using the standard deviation (STD) procedure described by Aghababaeyan *et al* (2023):

$$\mathrm{STD} = \left\| \sqrt{\sum_{i=1}^{n} \frac{A_{\mathrm{VS},j} - \mu_j}{n}},\ 1 \leqslant j < |\mathcal{I}| \times 3 \right\|. \tag{11}$$

Here, μ_j is the mean of feature j across the test set and n is the number of violating samples. A high violation-space diversity score indicates that violations occur with varying intensity across different electrodes rather than clustering in a single region, suggesting a more thorough exploration of the device's safety boundaries.

Finally, because both diversity metrics are impacted by sample size, we compute each on five randomly selected subsets of 200 violating input/output pairs and report the mean across subsets.

2.5. Experimental setup

All experiments were implemented in Python using both PyTorch and TensorFlow frameworks with mixed-precision inference for efficiency. Fuzzing and coverage computations were run on high-performance NVIDIA GPUs: an A6000 for retinal models and an RTX 4090 for cortical models.

We evaluate each coverage strategy for an equal wall-clock runtime to ensure fair comparison despite differences in computational overhead. We determine the total number of evaluated inputs by scaling the ratio between the time to execute a single test, and compute its coverage contribution relative to the time to execute a single test in our baseline strategies. Wall-clock normalization allows us to consider the time taken to evaluate new coverage of each coverage metric in our comparison—a metric that is effective but slow to compute may not be of much utility in model testing compared to a metric that excels in speed and brute force. We measure the time taken for a single input test and coverage computation for each metric to scale how many tests it will perform in comparison experiments.

Hyperparameters for previously published neuron-coverage metrics were adopted from prior work, while those for our new metrics were tuned on short pilot runs. The goal of tuning hyperparameter K values is to choose a K that splits up the search space into enough distinct bins to count new behavior as new coverage and thus explore further, without counting every test as new coverage. Basic metrics B-N and B-A fall at opposite ends of this spectrum—a K value too low will function like B-N, and a K value too high will function like B-A. We evaluate the effectiveness of hyperparameter choices via the number and type of violations found. All model sizes, hyperparameters, number of tests per coverage metric, and code artifacts are available in the accompanying repository: https://github.com/mara-downing/safety_violation_fuzzing_visual_prostheses.

2.6. Models under test

We applied our framework to two classes of state-of-the-art stimulus encoders representative of current approaches in visual prosthetics: retinal and cortical encoders. Both encoders perform an *inverse mapping* from camera images to electrode stimulation parameters, but differ in anatomical target, output dimensionality, and training objectives.

2.6.1. Retinal stimulus encoders

The retinal encoder evaluated in this work is a deep stimulus encoder (DSE) that maps a grayscale image to three stimulation parameters per electrode (amplitude, frequency, and pulse duration) across a 15×15 epiretinal array with 400 μm spacing (Granley *et al* 2023). Its objective is to generate spatially organized pulse patterns that approximate natural scene structure when viewed through a

prosthetic. Training followed a hybrid optimization procedure that combined image reconstruction losses with human-in-the-loop feedback, without incorporating explicit stimulation safety constraints.

To assess how different safety-oriented design choices influence encoder behavior, we also include a family of DSE variants introduced by Schoinas *et al* (2025): R-DSE-L (baseline), R-DSE-L_{PR} (pulse-duration regularization), R-DSE-L_{FR} (frequency regularization), R-DSE-L_{PR+FR} (joint regularization), and R-DSE-L_{FC} (hard frequency clipping). For all retinal models, safety thresholds were set to $\epsilon_1 = 0.628\,\mu$C for per-electrode charge limit, $\epsilon_2 = 6$ mA for instantaneous current, and $\epsilon_3 = 100$ active electrodes.

A differentiable phosphene model provides the forward link between electrical stimuli and predicted percepts. The version used here (Granley *et al* 2023) extends earlier models of epiretinal activation (Beyeler *et al* 2019, Granley and Beyeler 2021) by representing each electrode's percept as a multivariate Gaussian whose size, eccentricity, and orientation depend on both local axon-fiber geometry and stimulation parameters. This formulation incorporates well-established dependencies of phosphene brightness, size, and elongation on amplitude and frequency (Greenwald *et al* 2009, Horsager *et al* 2009, Nanduri *et al* 2012, Beyeler *et al* 2019). Because Gaussian covariances are tilted along the underlying axon trajectories, the model captures anisotropic current spread and reproduces characteristic crescent-shaped percepts observed in human users (Hou *et al* 2024).

Phosphene parameters are personalized using psychophysical fits for each simulated user, enabling realistic variation across the array. Although percepts from individual electrodes are summed linearly, consistent with paired-electrode experiments showing near-independent activation of spatially separated axon bundles (Hou *et al* 2024), the axon-map structure introduces nonlinear spatial interactions that depend on local fiber geometry. The resulting combination of anatomical coupling and nonlinear stimulus dependence allows the model to approximate how encoder outputs transform into perceptual brightness and shape under realistic retinal constraints, making it a suitable testbed for evaluating safety violations in image-to-stimulation pipelines.

2.6.2. Cortical stimulus encoders

The cortical encoder evaluated in this study is the C-Viseon model introduced in van der Grinten *et al* (2024), which maps a target image to stimulation amplitudes on a subset of 60 electrodes from a 96-channel Utah array implanted in the primary visual cortex (Normann *et al* 2009, Fernández *et al* 2021). In contrast to the retinal encoder, frequency and pulse duration are held constant across electrodes, which simplifies the output space but increases the importance of aggregate current constraints such as V_{AE} and V_{IC}. The encoder is trained end-to-end to minimize reconstruction error between predicted phosphenes and the target image, without explicit safety-oriented regularization.

To evaluate how architectural and loss-design choices influence safety, we include two variants from van der Grinten *et al* (2024): C-Viseon (baseline with no inter-electrode interaction), C-Viseon_{CL} (adds a co-localization loss that penalizes activation of adjacent electrodes), and C-Viseon_{COA} (incorporates an explicit coactivation model that increases effective current when neighboring electrodes are active). All models assume a 10×10 Utah-array layout with 0.4 mm electrode spacing, consistent with human V1 implants.

Safety limits were set to $\epsilon_1 = 20.4$ nC, $\epsilon_2 = 3.6$ mA, and $\epsilon_3 = 30$ active electrodes. These values match published human data from Utah-array stimulation studies (Fernández and Normann 2016, Fernandez 2018, Chen *et al* 2020, Fernández *et al* 2021, Moure *et al* 2025) and reflect conservative thresholds for safe operation in cortical implants.

2.6.3. Evaluation protocol

For each encoder, an input image $\mathbf{x} \in \mathbb{R}^d$ is mapped to a stimulation vector $\mathbf{y} = M(\mathbf{x})$. For retinal models, $\mathbf{y}$ includes frequency, amplitude, and pulse duration at each electrode. For cortical models, $\mathbf{y}$ consists of per-electrode amplitudes only.

All experiments report averages across three simulated users and six seed sets. Seed sets include three images drawn from each model's training distribution and three images from ImageNet (Deng *et al* 2009) to evaluate generalization. A summary of all model variants is provided in table 2, with full architectures linked in the accompanying GitHub repository.

3. Results

We evaluate the proposed CGF framework on both retinal and cortical visual prosthesis encoders introduced in section 2.6. For each model class, we compare all coverage strategies in terms of the number and diversity of safety violations uncovered.

Each fuzzing run begins with a set of seed images: across six tests of each coverage metric, three have seed sets drawn from the model's own training dataset, and three have seed sets drawn from ImageNet (Deng *et al* 2009). These ImageNet seed sets constitute real-world, out-of-distribution inputs to the models. Seeds are iteratively mutated according to the selected coverage strategy (section 2.2), producing candidate images that are passed through the encoder to generate stimulation parameters $\mathbf{y} = M(\mathbf{x})$. These outputs are then evaluated against the safety constraints defined in section 2.1, and the process is continued until a fixed number of test cases have been evaluated. For every coverage metric, we quantified both (i) the total number of unique safety violations

Table 2. Overview of models.

Model	Specifics	Validation Loss
R-DSE	Retinal deep stimulus encoder (Granley *et al* 2023).	0.0500*
R-DSE-L	R-DSE with larger training data and pulse duration modulating amplitude (Granley *et al* 2023, Schoinas *et al* 2025).	0.0625
R-DSE-L_{PR}	R-DSE-L architecture with L2 pulse duration regularization.	0.0703
R-DSE-L_{FR}	R-DSE-L architecture with L2 frequency regularization.	0.0537
R-DSE-L_{PR+FR}	R-DSE-L architecture with L2 pulse duration and frequency regularization.	0.0592
R-DSE-L_{FC}	R-DSE-L architecture with frequency clipping.	0.0628
C-Viseon	Cortical deep stimulus encoder (van der Grinten *et al* 2024)	0.074
C-$Viseon_{CL}$	C-Viseon trained to minimize activation of neighboring electrodes (van der Grinten *et al* 2024).	0.083
C-$Viseon_{COA}$	C-Viseon trained assuming each active electrode amplifies active electrodes nearby (van der Grinten *et al* 2024).	0.088

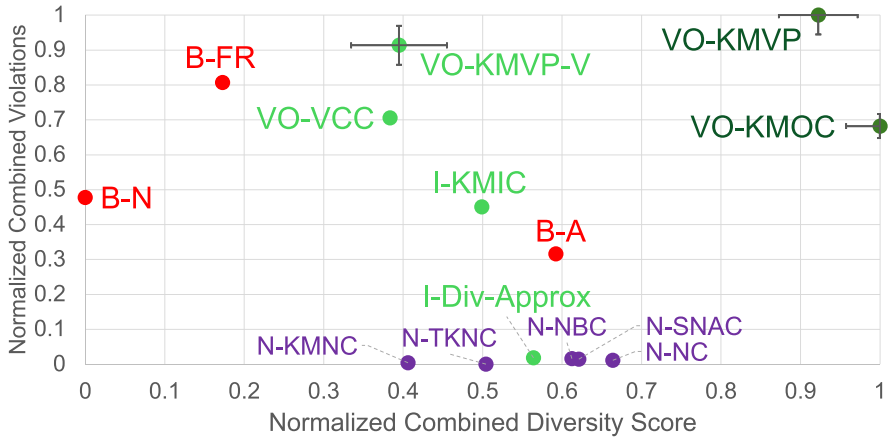

Figure 2. Fuzzing strategy comparison scatterplot for retinal models. *Y*-axis: number of violations found, normalized using linear normalization to a [0–1] range; *x*-axis: normalized combined diversity score (also using linear normalization to a [0–1] range). Each datapoint is the average of six tests, each with a different seed set, and error bars on the three best metrics show the standard error. Our metrics are shown in green, with our two best VO-KMVP and VO-KMOC highlighted in dark green. Neuron coverage metrics are shown in purple, and basic metrics in red.

discovered and (ii) the diversity of those violations across input features and electrodes (section 2.4).

Results for the retinal encoders are shown in figures 2 and 3, and results for the cortical encoders in figures 4 and 5. Each point in figures 2 and 4 represents the mean across simulated participants and seed sets. Error bars show the standard error for the top three options.

For the retinal models, figure 2 plots the number of violations discovered (*y*-axis) against a combined diversity score (*x*-axis), while figure 3 presents the same data as a normalized composite score, equally weighting violations and diversity to highlight the best-performing metrics on a shared scale. Due to it being an outlier, we omit the B-Local basic metric, which yields a normalized combined violation score of 1.60 and a diversity score of −0.78.

Results for the cortical models follow the same format (figures 4 and 5), where B-Local again produces an extreme imbalance between violation count and diversity (violation score 2.80, diversity −0.94).

Across both model families, violation-output coverage metrics (particularly VO-KMVP and VO-KMOC) discover more unique and spatially distributed violations than random or neuron-coverage baselines. Their joint optimization of severity and output-space breadth yields higher diversity without overproducing trivial variants, indicating that CGF can expose clinically relevant failure modes in complex neurostimulation models.

In figures 3 and 5 we include an extra bar to indicate how conventional testing compares when evaluated with the same criteria and normalization procedure—this conventional testing utilizes only the available test set for each model—the same set on which validation loss is computed. For both retinal and cortical models, the total number of violations findable is lower than with fuzzing, as expected, as conventional testing does not include a mutation approach to add new testable inputs once those available are exhausted.

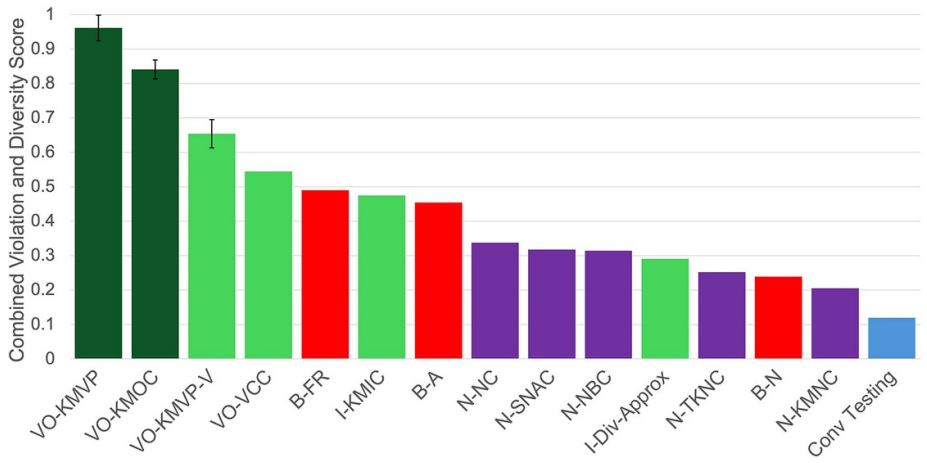

Figure 3. Fuzzing strategy comparison by combined violation and diversity score for retinal models. Each bar represents an average of normalized violations and normalized diversity score (with the linear normalization shown in figure 2), equally weighted. Each bar is the average of six tests, each with a different seed set, and error bars on the three best metrics show the standard error. Our metrics are shown in green, with our two best VO-KMVP and VO-KMOC highlighted in dark green. Neuron coverage metrics are shown in purple, and basic metrics in red. Conventional testing (model test set with no mutations) is shown in blue.

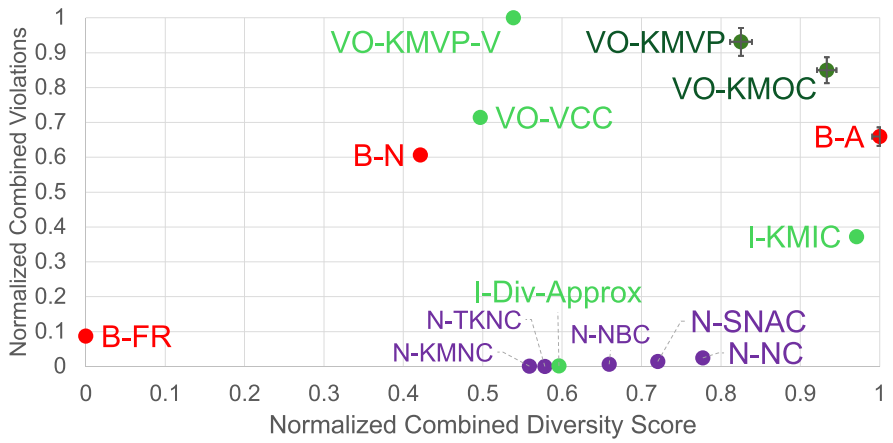

Figure 4. Fuzzing strategy comparison scatterplot for cortical models. *Y*-axis: number of violations found, normalized using linear normalization to a [0–1] range; *x*-axis: normalized combined diversity score (also using linear normalization to a [0–1] range). Each datapoint is the average of six tests, each with a different seed set, and error bars on the three best metrics show the standard error. Our metrics are shown in green, with our two best VO-KMVP and VO-KMOC highlighted in dark green. Neuron coverage metrics are shown in purple, and basic metrics in red.

For the cortical model, conventional testing shows good GD over the input feature space but worse violation space diversity than both VO-KMOC and VO-KMVP coverage metrics, as it did not include any method of forcing the search to find new output or violation values. The better GD on the input feature space is to be expected, as this measures the diversity of features in the input images.

For the retinal model, conventional testing missed one key discovery which our fuzzing is able to capture—the existence of V_{PI} violations. Not one of the 10 000 test images produced a V_{PI} violation with the retinal model, which could lead a developer to believe that the model was successful in preventing these violations, despite our fuzzing results clearly indicating these are possible. Conventional testing thus had low violation space diversity (lower than every other tested strategy except B-Local) and GD of its input feature space lower than that of N-NC, N-SNAC, B-A, and VO-KMOC.

As some patients may have lower thresholds for their own personal comfort or safety, we compute combined violation and diversity scores for the top three metrics, top basic metric, and top Neuron Coverage metric under modest lowering of each ϵ threshold value. Results are shown in table 3 for retinal and table 4 for cortical. For the cortical results, the top basic metric is in the top three metrics so only four metrics are analyzed.

In all cases, the results are in line with our results using unperturbed epsilon threshold values, and relative metric rankings remain the same. As these scores are normalized with the same procedure as figures 3 and 5, values above 1 are possible—lowering safety

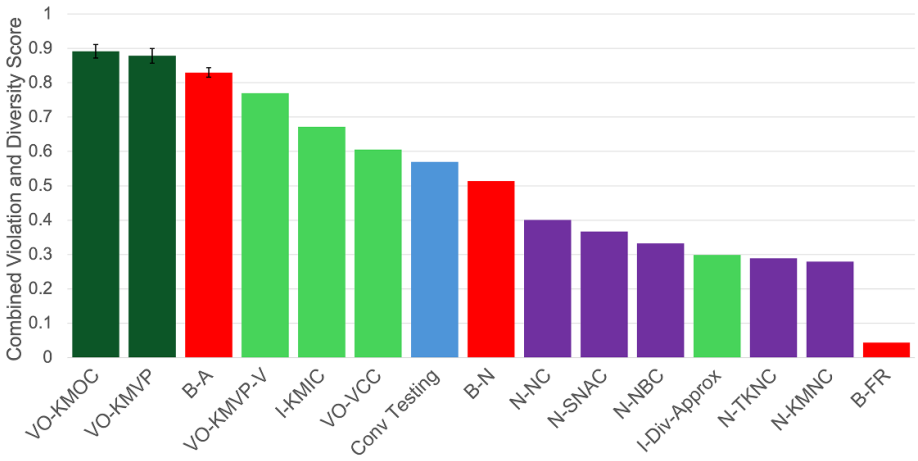

Figure 5. Fuzzing strategy comparison by combined violation and diversity score for cortical models. Each bar represents an average of normalized violations and normalized diversity score (with the linear normalization shown in figure 4), equally weighted. Each bar is the average of six tests, each with a different seed set, and error bars on the three best metrics show the standard error. Our metrics are shown in green, with our two best VO-KMVP and VO-KMOC highlighted in dark green. Neuron coverage metrics are shown in purple, and basic metrics in red. Conventional testing (model test set with no mutations) is shown in blue.

Table 3. Combined normalized violation count and diversity score for retinal models under ϵ threshold perturbations. The top two metrics for each test, measured by combined violation and diversity score, are in bold.

Perturbation	VO-KMVP	VO-KMOC	VO-KMVP-V	B-FR	N-NC
ϵ_1 reduced 10%	**0.9844**	**0.8172**	0.5798	0.4915	0.3208
ϵ_2 reduced 10%	**0.9677**	**0.8277**	0.5924	0.5008	0.3232
ϵ_3 reduced 10%	**0.9326**	**0.8131**	0.6175	0.4982	0.3249

Table 4. Combined normalized violation count and diversity score for retinal models under ϵ threshold perturbations. The top two metrics for each test, measured by combined violation and diversity score, are in bold.

Perturbation	VO-KMOC	VO-KMVP	B-A	N-NC
ϵ_1 reduced 10%	**0.9258**	**0.9164**	0.9053	0.4127
ϵ_2 reduced 10%	**1.0146**	**0.9320**	0.9269	0.4013
ϵ_3 reduced 10%	**0.9662**	**0.9403**	0.8970	0.3999

threshold values can allow for higher violation counts than seen in the unperturbed threshold experiments.

3.1. Violation discovery for model selection

We next examine how violation-focused fuzzing can differentiate models trained with varying degrees of safety regularization. Six retinal encoders and three cortical encoders were tested using the VO-KMVP metric under identical conditions (table 2). Each model shared the same architecture but differed in regularization terms, clipping strategies, or additional loss components designed to reduce unsafe stimulation.

Figure 6 summarizes the total number of violations discovered for each model and constraint type (V_{PI}, V_{CD}, V_{IC}, V_{AE}). Models with explicit loss penalties on pulse duration or frequency (R-DSE-L_{PR}, R-DSE-L_{FR}, R-DSE-L_{PR+FR}) consistently reduced the number of violations compared to the baseline R-DSE-L model, while simple frequency clipping (R-DSE-L_{FC}) achieved the largest overall reduction in V_{PI} violations. However, improvements were not always without penalty; for instance, regularizing pulse width and frequency (R-DSE-L_{PR+FR}) resulted in lower V_{PI} violations but higher V_{CD} violations than the R-DSE-L model, and regularizing just pulse width (R-DSE-L_{PR}) resulted in lower V_{CD} violations but higher V_{IC} violations. The unregularized R-DSE model produced catastrophic outputs, exceeding 500 000 combined V_{PI} and V_{CD} events, and is thus omitted from figure 6.

For cortical encoders, coactivation-aware and lateral-inhibition models (C-$Viseon_{COA}$, C-$Viseon_{CL}$) lowered the rate of overstimulation violations (V_{AE}) relative to the base C-Viseon model, but none fully eliminated unsafe conditions. Cortical models showed lower counts of V_{IC} violations than retinal models and zero V_{PI} violations, but higher counts of V_{CD} violations and V_{AE} violations. Across both retinal and cortical architectures, the fuzzing framework successfully exposed violations that were not apparent from training or validation loss alone, illustrating its potential as a quantitative benchmark for model comparison.

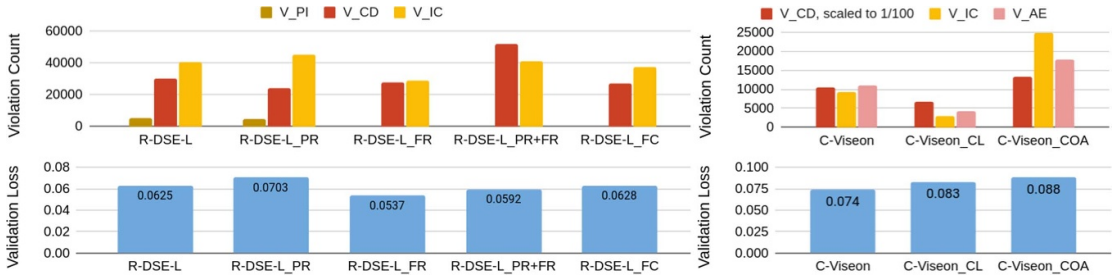

Figure 6. Bar charts of violations found and validation loss for each trained model variant (described in table 2). Left: retinal models; Right: cortical models. V_{CD} and V_{IC} violations are present in both retinal and cortical models, but V_{PI} violations are only shown for retinal (impossible in cortical) and V_{AE} violations are only shown for cortical (not discovered in retinal).

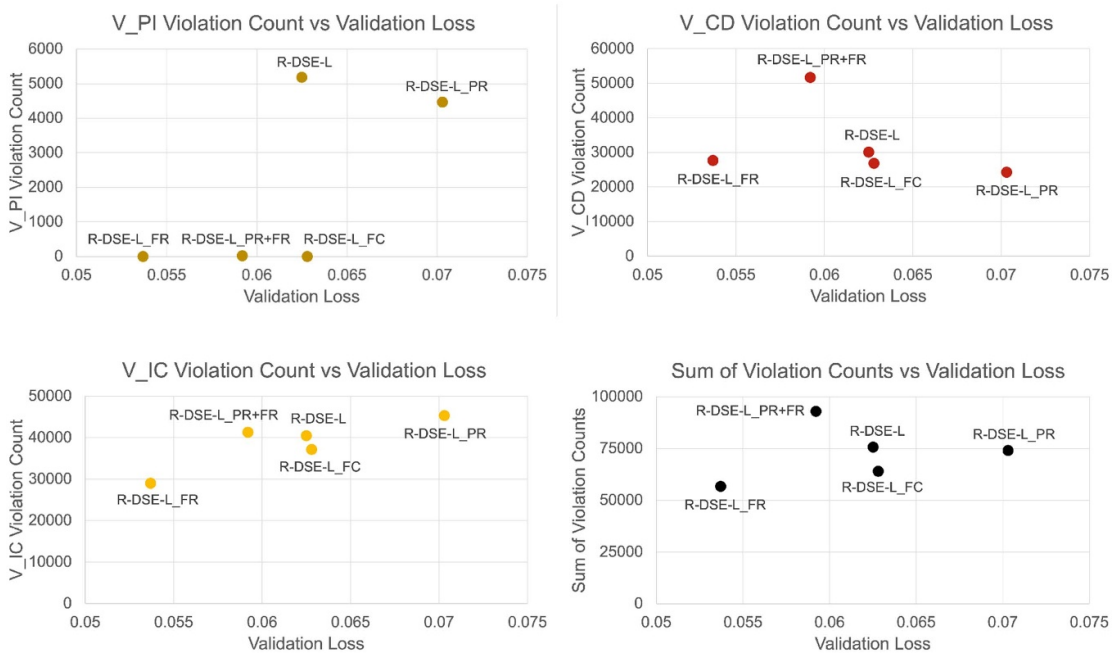

Figure 7. Violation counts (y-axis) plotted against validation loss (x-axis) for retinal models. Each plot shows a different type of violation (or the sum of all three discovered types for the model) on the y-axis, with color corresponding to the type of violation as denoted in figure 6. In some cases (ex. V_{IC} violation count) the models with fewer violations tend to have lower validation loss, whereas in other cases (ex. V_{CD} violation count) the models with fewer violations tend to have higher validation loss.

When considering the tradeoff between safety and perceptual fidelity, we can see in figure 6 that this is not a simple mapping—under some training criteria, a model can show improvements in perceptual fidelity and safety (for example, R-DSE-L_{FR}) and in other cases a safety improvement may come at the cost of perceptual fidelity or vice versa. We demonstrate in figure 7 that this tradeoff can differ by violation type as well (plotted with retinal models, as this allows 5 points for comparison), as V_{IC} violations on the whole show improvement with lower validation loss, opposite of V_{CD} violations. Our framework allows for this investigation and comparison, which is essential to future work in the field improving model architecture and training both for perceptual fidelity and safety.

3.2. Severity of violations

While comparing the count of safety violations is meaningful under the current medical and FDA regulated understanding of visual prosthesis safety, our approach can also be used to distinguish the degree to which these violations occur—metrics VO-KMVP and VO-KMOC as well as our violation space diversity metric utilize this degree in their calculations. We provide figure 8 to visualize this spread on the base cortical model C-Viseon, with separations for possible marginal, moderate, and catastrophic violation severity levels. The exact distinctions between and the dangerous effects caused at each level are not known clinically at this time.

For whole-input (V_A) violation types we see that higher severity violations are less common, whereas

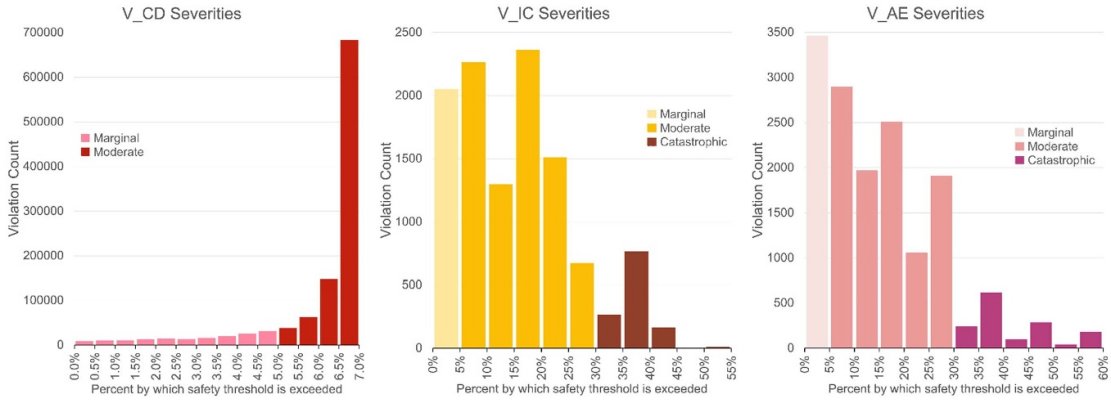

Figure 8. Prevalence of V_{CD}, V_{IC}, and V_{AE} violations by severity level for a patient using the C-Viseon cortical model. The lightest color indicates sample marginal violation levels, medium color moderate (corresponding to the type of violation as denoted in figure 6), and the darkest color indicates catastrophic safety violations. We make the distinction that 5% over the safety limit separates marginal from moderate, and 30% over the safety limit separates moderate from catastrophic. Due to the model design, there are no V_{CD} violations in this catastrophic category.

for V_{CD} violations we can see that when an individual electrode has a violation, it is more likely at the maximum level allowed by the model—these models are capped at 21.76 nC.

4. Discussion

This study introduces a CGF framework (Chen *et al* 2018) for systematically uncovering output-level violations in machine-learning models for neural stimulation. By formalizing domain-specific inequality constraints and defining violation-output coverage metrics, we provide a quantitative method to test whether trained models respect physiological limits under diverse and perturbed inputs. Applied to deep stimulus encoders for retinal and cortical visual prostheses (Granley *et al* 2023, van der Grinten *et al* 2024), the framework identified unsafe stimulation patterns that conventional loss functions and validation metrics failed to expose.

4.1. Violation-focused fuzzing reveals hidden failure modes

Across both prosthesis types, violation-output coverage metrics (VO-KMVP and VO-KMOC) provided the most informative search signals. By quantifying exploration directly in the space of stimulation outputs, they enable black-box evaluation of trained encoders without requiring architectural access or inspection of internal activations. These metrics consistently uncovered a broader and more diverse set of violations than neuron-coverage or random baselines, demonstrating that safety can be assessed as an observable property of the input–output mapping itself.

Because the violations are expressed in physical units, the results can be interpreted in terms of quantities that matter clinically, such as charge density, instantaneous current, and the number of active electrodes. Framing safety in these domain-relevant units allows direct comparison against established biophysical limits and provides a clear link between model behavior and known failure modes in implantable systems.

Taken together, these results show that violation-guided fuzzing offers a practical and quantitative approach for identifying unsafe operating regimes in ML-driven neurostimulation.

4.2. Retinal and cortical insights

For epiretinal encoders (Granley *et al* 2023), fuzzing revealed that models optimized for perceptual fidelity can still generate physically impossible or unsafe pulses when exposed to out-of-distribution inputs. Frequency regularization mitigated these violations, while duration penalties alone had limited effect, highlighting the nonlinear coupling between pulse width and frequency in charge accumulation (Horsager *et al* 2011, Nanduri *et al* 2012, Ghaffari *et al* 2020, Hou *et al* 2024).

In cortical encoders (van der Grinten *et al* 2024), loss terms discouraging co-activation of neighboring electrodes improved safety margins by reducing total current and the number of simultaneously active sites, consistent with intracortical studies of spatial interference and excitability (Chen *et al* 2020, Fernández *et al* 2021, Moure *et al* 2025). Conversely, models that explicitly incorporated current-spread or interaction terms inspired by experimental findings on crosstalk and waveform asymmetries (Wilke *et al* 2011, Haji Ghaffari 2021, Yücel *et al* 2022) tended to amplify unsafe amplitudes.

These results show that even biologically motivated modeling choices can introduce new risks if not empirically verified, underscoring the need for systematic post-training validation rather than reliance on training objectives or validation loss alone.

Our framework goes beyond stress testing: it can serve as a tool for model selection, diagnostic

analysis, and safety-aware refinement. As shown in figure 6, violation fuzzing enables comparative evaluation of training strategies and can guide iterative improvements. For example, we demonstrate how targeted regularization reduces violations without degrading accuracy. This supports developers in choosing architectures or loss functions that not only perform well but also respect safety constraints. These discovered weaknesses could then be used in combination with conventional performance metrics to inform modifications to regularization, architecture, or training strategies, in order to iteratively, manually, produce better versions of the model.

4.3. Toward principled model validation

Violation-guided fuzzing reframes safety evaluation as an empirical and reproducible testing process rather than an indirect training objective. It enables quantitative comparison of architectures, regularization schemes, and constraint formulations under identical conditions, offering interpretable robustness measures that complement perceptual or reconstruction metrics. This perspective is particularly relevant for implantable neurotechnologies, where reliability depends not only on performance but also on demonstrable adherence to physiological limits (Shannon 1992, Merrill *et al* 2005). At the same time, long-term usability and trust strongly influence whether implant recipients rely on their devices in everyday life (Nadolskis *et al* 2024). Integrating physiological validation with real-world user experience is therefore critical to translating engineering progress into functional benefit.

Prior work that invokes 'safety' in ML-driven neurostimulation has treated it synonymously with minimizing delivered current or charge during optimization (Shah and Chichilnisky 2020, Küçükoğlu *et al* 2025, Willis *et al* 2025). While such regularization can reduce mean output power, it does not verify that trained models remain within physiological limits once deployed, particularly under novel or perturbed inputs. Our findings show that aggregate or spatially localized violations can still arise from complex pulse interactions even when average current is minimized. By empirically testing deployed encoders and quantifying both the frequency and severity of violations, our framework complements these loss-based efforts and establishes a foundation for evidence-based safety benchmarking, which is a necessary step toward regulatory approval and ethical deployment of adaptive neural interfaces. Beyond compliance, ensuring demonstrable safety and reliability is an ethical imperative for responsible neurotechnology and AI governance (Yuste *et al* 2017).

4.4. Limitations and future directions

The present study establishes a foundation for systematic, output-level safety evaluation, yet several natural extensions remain.

Our experiments rely on simulation-based encoders and safety thresholds derived from published device specifications, which provide a controlled and reproducible testbed. The next step is hardware-in-the-loop validation, where electrode impedance, charge-transfer efficiency, thermal load, and long-term tissue response can be incorporated directly into the testing pipeline (Fernandez 2018). Integrating these physiological factors will enable more comprehensive assessments, and using violation feedback as a differentiable signal during training may allow joint optimization of perceptual fidelity and safety compliance.

Beyond the biophysics of stimulation, functional outcomes in prosthetic vision depend on cortical plasticity and perceptual learning (Beyeler *et al* 2017, Lunghi *et al* 2019, Caravaca-Rodriguez *et al* 2022, Esquenazi *et al* 2025). Extending violation analysis to include these neural and behavioral constraints could yield a more holistic framework that links device safety to perceptual outcomes and long-term usability.

Although this study focused on visual prostheses, the same methodology applies to other neuromodulatory systems which might use model-generated stimulation parameters such as deep-brain, spinal, and vagus-nerve stimulators (Little *et al* 2013, Okorokova *et al* 2018, Rao 2019, Drakopoulos and Verhulst 2023). As neuroprosthetic systems become increasingly autonomous and incorporate co-adaptive or reinforcement-learning components (Little *et al* 2013, Shenoy and Carmena 2014, Shanechi *et al* 2017, Rao 2019), output-level safety validation will be essential for closed-loop operation (Grani *et al* 2022, Beyeler 2025, Moure *et al* 2025). For applications where electrical stimulation is varied over time and violations are a result of a sequence of stimulations, mutation options may need to be modified but our overall CGF framework and coverage metrics can be applied directly.

By grounding evaluation in measurable, domain-specific constraints, violation-focused fuzzing provides a bridge between algorithmic innovation and clinical reliability. As intelligent neurotechnologies advance toward adaptive and closed-loop operation, such frameworks will be vital not only for improving device performance but also for ensuring ethical, trustworthy, and regulatory confidence in next-generation neural interfaces (Yuste *et al* 2017).

5. Conclusions

This work introduces violation-focused fuzzing as a systematic approach to evaluating the safety of machine-learning-based neural stimulation. By transforming safety from a training heuristic into an empirically measurable property, this framework enables reproducible benchmarking across architectures and regularization schemes and presents two highly effective coverage metrics for this measurement. Applied to deep stimulus encoders for the retina and visual cortex, it uncovers unsafe behaviors invisible to traditional validation metrics and establishes a quantitative basis for model certification. As neuroprosthetic systems advance toward adaptive, closed-loop operation, principled safety testing is essential to ensure both functional reliability and long-term patient trust.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://github.com/mara-downing/safety_violation_fuzzing_visual_prostheses (Downing *et al* 2026).

Funding

Partially supported by the National Science Foundation (NSF) under Awards #2124 039 and #2008 660 to TB and by the National Library of Medicine of the National Institutes of Health (NIH) under Award Number DP2-LM014268 to MB. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NSF or NIH.

Author contributions

Mara Downing 0009-0006-8431-6695
Conceptualization (equal), Data curation (equal), Formal analysis (equal), Investigation (equal), Software (equal), Visualization (equal), Writing – original draft (equal), Writing – review & editing (equal)

Matthew Peng 0009-0008-2945-2170
Investigation (equal), Resources (equal), Software (equal), Visualization (equal), Writing – review & editing (equal)

Jacob Granley 0000-0002-9024-2454
Conceptualization (equal), Resources (equal), Writing – review & editing (equal)

Michael Beyeler 0000-0001-5233-844X
Conceptualization (equal), Funding acquisition (equal), Project administration (equal), Writing – review & editing (equal)

Tevfik Bultan 0000-0003-2993-1215
Conceptualization (equal), Data curation (equal), Formal analysis (equal), Funding acquisition (equal), Investigation (equal), Project administration (equal), Software (equal), Visualization (equal), Writing – original draft (equal), Writing – review & editing (equal)

Appendix. Additional violation-focused coverage metrics

A.1. VO-KMVP-V

K-multisection violation proportion (only Violation) coverage uses the same strategy as *VO-KMVP*, with the caveat that new coverage is only valid if the new coverage is in a bin indicating a violation ($\alpha(\mathbf{y})/c \geqslant 1$ or $\alpha_i(\mathbf{y})/c \geqslant 1$).

A.2. VO-VCC

Violation constraint coverage uses the same strategy as *VO-KMVP*, with $K = 2$ and max $= 2$, which results in a coverage computation where the bins indicate presence or absence of violation but not its degree.

A.3. I-KMIC

K-multisection input coverage splits each pixel's value range into K equal-size bins. Let $\mathcal{P}$ be the set of pixels in the input image and let min and max be the upper and lower valid pixel values. Let $\min_k = \min + k \times (\max - \min)/K$, then

$$\mathrm{COV}(S) = \frac{\Sigma_{p\in\mathcal{P}}\Sigma_{k=0}^{K-1}\left[\exists \mathbf{x} \in S \,.\, \min_k \leqslant \mathrm{val}(\mathbf{x}, p) < \min_{k+1}\right]}{K \times |\mathcal{P}|}. \tag{12}$$

A.4. I-Div-Approx

Diversity Approximation coverage computation begins with a profiling step similar to N-KMNC, N-NBC, and N-SNAC, in which the model is executed on a set of input data to determine high and low values for each feature in the input-diversity feature vector. Next, each feature's range is split into K equal-sized bins. Output values that fall outside the expected range are marked in the highest or lowest bin, depending on whether they are above or below the range. Let $\mathcal{F}$ be the set of features in the feature vector and let $lo_{f,k} = lo_f + k \times (hi_f - lo_f)/K$, then:

$$\mathrm{COV}(S) = \frac{\Sigma_{f\in\mathcal{F}}\Sigma_{k=0}^{K-1}\left[\exists \mathbf{x} \in S \,.\, lo_{f,k} \leqslant \mathrm{val}(\mathbf{x}, f) < lo_{f,k+1}\right]}{K \times |\mathcal{F}|} \tag{13}$$

with the caveat that $lo_{f,0} \leqslant \mathrm{val}(\mathbf{x}, f)$ and $lo_{f,K-1} > \mathrm{val}(\mathbf{x}, f)$ always evaluate to true (values above or below

the range $[lo_f, hi_f]$ are counted as coverage in the lowest or highest available bin).

A.5. N-NC

NC (Pei *et al* 2017) defines an activation condition for each neuron (if the neuron's value is greater than or equal to a threshold t) and for each test case x computes which of the neurons in $\mathcal{N}$ have been activated by running that test case:

$$\text{COV}(S) = \frac{\Sigma_{n\in\mathcal{N}}\left[\exists \mathbf{x}\in S\,.\,\text{val}(\mathbf{x},n)\geqslant t\right]}{|\mathcal{N}|}. \quad (14)$$

Recall that $[expr]$ returns 1 if $expr$ evaluates to true, otherwise it returns 0.

A.6. N-KMNC

KMNC (Ma *et al* 2018) partitions each neuron's value range $[lo_n, hi_n]$ into K equal-size bins. Let $lo_{n,k} = lo_n + k\times(hi_n - lo_n)/K$, then:

$$\text{COV}(S) = \frac{\Sigma_{n\in\mathcal{N}}\Sigma_{k=0}^{K-1}\left[\exists \mathbf{x}\in S\,.\,lo_{n,k}\leqslant \text{val}(\mathbf{x},n) < lo_{n,k+1}\right]}{K\times|\mathcal{N}|}. \quad (15)$$

Note that in N-KMNC coverage, neuron values outside of the range $[lo_n, hi_n]$ are ignored.

A.7. N-NBC

NBC (Ma *et al* 2018) also uses lo_n and hi_n, but contrary to N-KMNC, it measures the number of neurons that take values above hi_n and below lo_n:

$$\text{COV}(S) = \frac{\Sigma_{n\in\mathcal{N}}\left(\left[\exists \mathbf{x}\in S.\text{val}(\mathbf{x},n) < lo_n\right] + \left[\exists \mathbf{x}\in S\,.\,\text{val}(\mathbf{x},n) > hi_n\right]\right)}{2\times|\mathcal{N}|}. \quad (16)$$

A.8. N-SNAC

SNAC (Ma *et al* 2018) uses only hi_n and measures the number of neurons with values above hi_n:

$$\text{COV}(S) = \frac{\Sigma_{n\in\mathcal{N}}\left[\exists \mathbf{x}\in S\,.\,\text{val}(\mathbf{x},n) > hi_n\right]}{|\mathcal{N}|}. \quad (17)$$

A.9. N-TKNC

TKNC (Ma *et al* 2018) tracks which K neurons have the highest values per layer when executing with a test case $\mathbf{x}$. Let $\text{top}(n,\mathbf{x},K)$ denote the set of K neurons in the same layer as neuron n which have the K top (highest) values in that layer for test input $\mathbf{x}$, then:

$$\text{COV}(S) = \frac{\Sigma_{n\in\mathcal{N}}\left[\exists \mathbf{x}\in S\,.\,n\in\text{top}(n,\mathbf{x},K)\right]}{|\mathcal{N}|} \quad (18)$$

References

Aghababaeyan Z, Abdellatif M, Briand L, Ramesh S and Bagherzadeh M 2023 Black-box testing of deep neural networks through test case diversity *IEEE Trans. Softw. Eng.* **49** 3182–204

Ayton L N *et al* 2020 An update on retinal prostheses *Clin. Neurophysiol.* **131** 1383–98

Beyeler M 2025 Bionic vision as neuroadaptive XR: closed-loop perceptual interfaces for neurotechnology (arXiv:2508.05963)

Beyeler M, Nanduri D, Weiland J D, Rokem A, Boynton G M and Fine I 2019 A model of ganglion axon pathways accounts for percepts elicited by retinal implants *Sci. Rep.* **9** 9199

Beyeler M, Rokem A, Boynton G M and Fine I 2017 Learning to see again: biological constraints on cortical plasticity and the implications for sight restoration technologies *J. Neural Eng.* **14** 051003

Beyeler M and Sanchez-Garcia M 2022 Towards a Smart Bionic Eye: AI-powered artificial vision for the treatment of incurable blindness *J. Neural Eng.* **19** 063001

Cameron T 2004 Safety and efficacy of spinal cord stimulation for the treatment of chronic pain: a 20-year literature review *J. Neurosurgery* **100** 254–67

Caravaca-Rodriguez D, Gaytan S P, Suaning G J and Barriga-Rivera A 2022 Implications of neural plasticity in retinal prosthesis *Investig. Ophthalmol. Visual Sci.* **63** 11

Chen C, Cui B, Ma J, Wu R, Guo J and Liu W 2018 A systematic review of fuzzing techniques *Comput. Secur.* **75** 118–37

Chen S C, Suaning G J, Morley J W and Lovell N H 2009 Simulating prosthetic vision: I. Visual models of phosphenes *Vis. Res.* **49** 1493–506

Chen X, Wang F, Fernandez E and Roelfsema P R 2020 Shape perception via a high-channel-count neuroprosthesis in monkey visual cortex *Science* **370** 1191–6

de Ruyter van Steveninck J, Güçlü U, van Wezel R and van Gerven M 2022 End-to-end optimization of prosthetic vision *J. Vision* **22** 20

Deng J, Dong W, Socher R, Li Li-J, Li K and Fei-Fei L 2009 Imagenet: a large-scale hierarchical image database *2009 IEEE Conf. on Computer Vision and Pattern Recognition* (https://doi.org/10.1109/CVPR.2009.5206848) pp 248–55

Dong Y, Zhang P, Wang J, Liu S, Sun J, Hao J, Wang X, Wang Li, Dong J and Dai T 2020 An empirical study on correlation between coverage and robustness for deep neural networks *2020 25th Int. Conf. on Engineering of Complex Computer Systems (ICECCS)* (IEEE) pp 73–82

Downing M, Peng M, Granley J, Beyeler M and Bultan T 2026 Safety Violation Fuzzing for Visual Prostheses GitHub (available at : https://github.com/mara-downing/safety_violation_fuzzing_visual_prostheses)

Drakopoulos F and Verhulst S 2023 A neural-network framework for the design of individualised hearing-loss compensation *IEEE/ACM Trans. Audio Speech Language Process.* **31** 2395–409

Esquenazi R B, Meier K, Beyeler M, Wright D, Boynton G M and Fine I 2025 Perceptual learning of prosthetic vision using video game training *J. Vision* **25** 12

Fernandez E 2018 Development of visual neuroprostheses: trends and challenges *Bioelectron. Med.* **4** 12

Fernández E *et al* 2021 Visual percepts evoked with an intracortical 96-channel microelectrode array inserted in human occipital cortex *J. Clin. Investig.* **131** e151331

Fernández E and Normann R A 2016 Cortivis approach for an intracortical visual prostheses *Artificial Vision: A Clinical Guide* (Springer) pp 191–201

Ghaffari D H 2021 Improving the resolution of prosthetic vision through stimulus parameter optimization *Thesis* (available at: http://deepblue.lib.umich.edu/handle/2027.42/169970) (Accepted 24 September 2021)

Ghaffari D H, Finn K E, Jeganathan V S E, Patel U, Wuyyuru V, Roy A and Weiland J D 2020 The effect of waveform asymmetry on perception with epiretinal prostheses *J. Neural Eng.* **17** 045009

Grani F, Soto-Sánchez C, Doblado A R, Peco R L, Gonzalez-Lopez P and Fernandez E 2025 Neural correlates of phosphene perception in blind individuals: A step toward a bidirectional cortical visual prosthesis *Sci. Adv.* **11** eadv8846

Grani F, Soto-Sánchez C, Fimia A and Fernández E 2022 Toward a personalized closed-loop stimulation of the visual cortex: Advances and challenges *Front. Cellular Neurosci.* **16** 1034270

Granley J and Beyeler M 2021 A computational model of phosphene appearance for epiretinal prostheses *2021 43rd Annual Int. Conf. of the IEEE Engineering in Medicine & Biology Society (EMBC)* pp 4477–81 (available at: https://ieeexplore.ieee.org/abstract/document/9629663)

Granley J, Fauvel T, Chalk M and Beyeler M 2023 Human-in-the-loop optimization for deep stimulus encoding in visual prostheses *37th Conf. on Neural Information Processing Systems* (available at: https://openreview.net/forum?id=ZED5wdGous&referrer=)

Granley J, Relic L and Beyeler M 2022a Hybrid neural autoencoders for stimulus encoding in visual and other sensory neuroprostheses *Advances in Neural Information Processing Systems* vol 35 pp 22671–85

Granley J, Riedel A and Beyeler M 2022b Adapting brain-like neural networks for modeling cortical visual prostheses (arXiv:2209.13561)

Greenwald S H, Horsager A, Humayun M S, Greenberg R J, McMahon M J and Fine I 2009 Brightness as a function of current amplitude in human retinal electrical stimulation *Invest. Ophthalmol. Vis. Sci.* **50** 5017–25

Grill W M and Mortimer J T 1995 Stimulus waveforms for selective neural stimulation *IEEE Eng. Med. Biol. Mag.* **14** 375–85

Horsager A, Boynton G M, Greenberg R J and Fine I 2011 Temporal interactions during paired-electrode stimulation in two retinal prosthesis subjects *Invest. Ophthalmol. Vis. Sci.* **52** 549–57

Horsager A, Greenwald S H, Weiland J D, Humayun M S, Greenberg R J, McMahon M J, Boynton G M and Fine I 2009 Predicting visual sensitivity in retinal prosthesis patients *Invest. Ophthalmol. Vis. Sci.* **50** 1483–91

Hou Y, Nanduri D, Granley J, Weiland J D and Beyeler M 2024 Axonal stimulation affects the linear summation of single-point perception in three Argus II users *J. Neural Eng.* **21** 026031

Huang Li, Sun W, Yan M, Liu Z, Lei Y and Lo D 2024 Neuron semantic-guided test generation for deep neural networks fuzzing *ACM Trans. Softw. Eng. Methodol.* **34** 1–38

Küçükoğlu B, Rueckauer B, de Ruyter van Steveninck J, van der Grinten M, Güçlütürk Y, Roelfsema P R, Güçlü U and van Gerven M 2025 End-to-end learning of safe stimulation parameters for cortical neuroprosthetic vision *J. Neural Eng.* **22** 046022

Li Z, Ma X, Xu C and Cao C 2019 Structural coverage criteria for neural networks could be misleading *2019 IEEE/ACM 41st Int. Conf. on Software Engineering: New Ideas and Emerging Results (ICSE-NIER)* (IEEE) pp 89–92

Little S *et al* 2013 Adaptive deep brain stimulation in advanced Parkinson disease *Ann. Neurol.* **74** 449–57

Lunghi C, Galli-Resta L, Binda P, Cicchini G M, Placidi G, Falsini B and Morrone M C 2019 Visual Cortical Plasticity in Retinitis Pigmentosa *Investig. Ophthalmol. Vis. Sci.* **60** 2753–63

Ma L, Juefei-Xu F, Zhang F, Sun J, Xue M, Li B, Chen C, Su T, Li Li, Liu Y *et al* 2018 Deepgauge: Multi-granularity testing criteria for deep learning systems *Proc. 33rd ACM/IEEE Int. Conf. on Automated Software Engineering* pp 120–31

McCreery D B, Agnew W F, Yuen T G and Bullara L 1990 Charge density and charge per phase as cofactors in neural injury induced by electrical stimulation *IEEE Trans. Bio-Med. Eng.* **37** 996–1001

Merrill D R, Bikson M and Jefferys J G R 2005 Electrical stimulation of excitable tissue: design of efficacious and safe protocols *J. Neurosci. Methods* **141** 171–98

Moure P *et al* 2025 Deep learning-based control of electrically evoked activity in human visual cortex (available at: www.biorxiv.org/content/10.1101/2025.09.24.678361v1).

Nadolskis L, Turkstra L M, Larnyo E and Beyeler M 2024 Aligning Visual Prosthetic Development With Implantee Needs *Transl. Vis. Sci. Technol.* **13** 28

Nanduri D, Fine I, Horsager A, Boynton G M, Humayun M S, Greenberg R J and Weiland J D 2012 Frequency and amplitude modulation have different effects on the percepts elicited by retinal stimulation *Invest. Ophthalmol. Vis. Sci.* **53** 205–14

Normann R A, Greger B A, House P, Romero S F, Pelayo F and Fernandez E 2009 Toward the development of a cortically based visual neuroprosthesis *J. Neural Eng.* **6** 035001

Okorokova E V, He Q and Bensmaia S J 2018 Biomimetic encoding model for restoring touch in bionic hands through a nerve interface *J. Neural Eng.* **15** 066033

Park S H and Han K 2018 Methodologic guide for evaluating clinical performance and effect of artificial intelligence technology for medical diagnosis and prediction *Radiology* **286** 800–9

Pei K, Cao Y, Yang J and Jana S 2017 Deepxplore: automated whitebox testing of deep learning systems *Proc. 26th Symp. on Operating Systems Principles* pp 1–18

Rao R P N 2019 Towards Neural Co-Processors for the Brain: Combining decoding and encoding in brain-computer interfaces *Curr. Opin. Neurobiol.* **55** 142–51

Schoinas E, Rastogi A, Carter A, Granley J, and Beyeler M 2025 Evaluating deep human-in-the-loop optimization for retinal implants using sighted participants (arXiv:2502.00177)

Second Sight 2013 *Argus® II Retinal Prosthesis System Surgeon Manual* (Number 900029-001 Rev C) (Second Sight Medical Products, Inc.) (available at: www.accessdata.fda.gov/cdrh_docs/pdf11/h110002c.pdf)

Shah N P and Chichilnisky E J 2020 Computational challenges and opportunities for a bi-directional artificial retina *J. Neural Eng.* **17** 055002

Shanechi M M, Orsborn A L, Moorman H G, Gowda S, Dangi S and Carmena J M 2017 Rapid control and feedback rates enhance neuroprosthetic control *Nat. Commun.* **8** 13825

Shannon R V 1992 A model of safe levels for electrical stimulation *IEEE Trans. Bio-Med. Eng.* **39** 424–6

Shenoy K V and Carmena J M 2014 Combining decoder design and neural adaptation in brain-machine interfaces *Neuron* **84** 665–80

Simonyan K and Zisserman A 2014 Very deep convolutional networks for large-scale image recognition (arXiv:1409.1556)

U.S. Food and Drug Administration Humanitarian device exemption h110002 supplement s017

van der Grinten M et al 2024 Towards biologically plausible phosphene simulation for the differentiable optimization of visual cortical prostheses *eLife* **13** e85812

Wilke R G H, Moghadam G K, Lovell N H, Suaning G J and Dokos S 2011 Electric crosstalk impairs spatial resolution of multi-electrode arrays in retinal implants *J. Neural Eng.* **8** 046016

Willis J A, Wright C E, Zhu R, Ruan Y, Stallings J, Abrego A M, Joshi A A, Leahy R M, Tandon N and Seymour J P 2025 Optimizing electrode placement and information capacity for local field potentials in cortex (https://doi.org/10.1101/2025.04.25.650658)

Xie X, Ma L, Juefei-Xu F, Xue M, Chen H, Liu Y, Zhao J, Li B, Yin J and See S 2019 Deephunter: a coverage-guided fuzz testing framework for deep neural networks *Proc. 28th ACM SIGSOFT Int. Symp. on Software Testing and Analysis* pp 146–57

Yang Z, Shi J, Asyrofi M H and Lo D 2022 Revisiting neuron coverage metrics and quality of deep neural networks *2022 IEEE Int. Conf. on Software Analysis, Evolution and Reengineering (SANER)* (IEEE) pp 408–19

Yücel E I, Sadeghi R, Kartha A, Montezuma S R, Dagnelie G, Rokem A, Boynton G M, Fine I and Beyeler M 2022 Factors affecting two-point discrimination in Argus II patients *Front. Neurosci.* **16** 901337

Yuste R *et al* 2017 Four ethical priorities for neurotechnologies and AI *Nature* **551** 159–63