A Hybrid Neural Autoencoder for Sensory Neuroprostheses and Its Applications in Bionic Vision

Jacob Granley  
Department of Computer Science  
University of California, Santa Barbara  
jgranley@ucsb.edu

Lucas Relic  
Department of Computer Science  
University of California, Santa Barbara  
lucasrelic@ucsb.edu

Michael Beyeler  
Department of Computer Science  
University of California, Santa Barbara  
mbeyeler@ucsb.edu

Abstract

Sensory neuroprostheses are emerging as a promising technology to restore lost sensory function or augment human capacities. However, sensations elicited by current devices often appear artificial and distorted. Although current models can often predict the neural or perceptual response to an electrical stimulus, an optimal stimulation strategy solves the inverse problem: what is the required stimulus to produce a desired response? Here we frame this as an end-to-end optimization problem, where a deep neural network encoder is trained to invert a known, fixed forward model that approximates the underlying biological system. As a proof of concept, we demonstrate the effectiveness of our hybrid neural autoencoder (HNA) on the use case of visual neuroprostheses. We found that HNA is able to produce high-fidelity stimuli from the MNIST and COCO datasets that outperform conventional encoding strategies and surrogate techniques across all tested conditions. Overall this is an important step towards the long-standing challenge of restoring high-quality vision to people living with incurable blindness and may prove a promising solution for a variety of neuroprosthetic technologies.

1 Introduction

Sensory neuroprostheses are emerging as a promising technology to restore lost sensory function or augment human capacities [10, 17]. In such devices, diminished sensory modalities (e.g., hearing [49], vision [18, 29], cutaneous touch [45]) are re-enacted through streams of artificial input to the nervous system. For example, visual neuroprostheses electrically stimulate neurons in the early visual system to elicit neuronal responses that the brain interprets as visual percepts. Such devices have the potential to restore a rudimentary form of vision to millions of people living with incurable blindness. However, all of these technologies face the challenge of interfacing with a highly nonlinear system of biological neurons whose role in perception is not fully understood. Due to the limited resolution of electrical stimulation, prostheses often create neural response patterns foreign to the brain. Consequently, sensations elicited by current sensory neuroprostheses often appear artificial and distorted [15, 32]. A major outstanding challenge is thus to identify a stimulus encoding that leads to perceptually intelligible sensations. Often there exists a numeric or symbolic forward model, $f$ (Fig. 1A), constrained by empirical data, that can predict a neuronal or (ideally) perceptual response to the applied stimulus (see [6] for a recent review). To find the stimulus that will elicit a desired response,
one essentially needs to find the inverse of the forward model, $f^{-1}$. However, realistic forward models are rarely analytically invertible, making this a challenging open problem for neuroprostheses.

Here we propose to approach this as an end-to-end optimization problem, where a deep neural network (encoder) is trained to invert a known, fixed forward model (decoder, Fig. 1B). The encoder is trained to predict the patterns of electrical stimulation patterns that elicit perception (e.g., vision, audition) or neural responses (e.g., firing rates) closest to the target. This hybrid neural autoencoder (HNA) could in theory be used to optimize stimuli for any open-loop neuroprosthesis with a known forward model that approximates the underlying biological system.

In order to optimize end-to-end, the forward model must be differentiable and computationally efficient. When this is not the case, an alternative approach is to train a surrogate neural network, $\hat{f}$, to approximate the forward model [38, 31, 33, 34]. However, even well-trained surrogates may result in errors when included in our end-to-end framework, due to the encoders’ ability to learn to exploit the surrogate model. We therefore also evaluate whether a surrogate approach to HNA is suitable.

To this end, we make the following contributions:

- We propose a hybrid neural autoencoder (HNA) consisting of a deep neural encoder trained to invert a fixed, numerical or symbolic forward model (decoder) to optimize stimulus selection. Our framework is general and addresses an important challenge with sensory neuroprostheses.

- As a proof of concept, we demonstrate the utility of HNA for visual neuroprostheses, where we predict electrode activation patterns required to produce a desired visual percept. We show that the HNA is able to produce high-fidelity, patient-specific stimuli representing handwritten digits and segmented images of everyday objects, drastically outperforming conventional approaches.

- We evaluate replacing a computationally expensive or nondifferentiable forward model with a surrogate, highlighting benefits and potential dangers of popular surrogate techniques.

## 2 Background

### Sensory Neuroprostheses

Recent advances in neural understanding, wearable electronics, and biocompatible materials have accelerated the development of sensory neuroprostheses to restore perceptual function to people with impaired sensation. Use of neuroprostheses typically requires invasive implants that elicit neural responses via electrical, magnetic, or optogenetic stimulation. Two of the most promising applications include cochlear implants, which stimulate the auditory nerve to elicit sounds [49], and visual implants (see next subsection) to restore vision to the blind. However, a variety of other devices are in development; for instance, to restore touch [45, 44] or motor function [7]. The latter differ from other sensory neuroprostheses in that they generate stimuli using motor
Restoring Vision to the Blind

For millions of people who are living with incurable blindness, a visual prosthesis (bionic eye, Fig. 2, left) may be the only treatment option [18]. Analogous to cochlear implants, these devices electrically stimulate surviving cells in the visual pathway to evoke visual percepts (phosphenes), which can support simple behavioral tasks [29, 43, 24].

A common misconception is that each electrode in the microelectrode array can be thought of as a pixel in an image; to generate a complex visual experience, one then simply needs to turn on the right combination of pixels [14]. However, recent evidence suggests that phosphenes often appear distorted (e.g., as simple geometric shapes such as lines, wedges, and blobs) and vary drastically across subjects and electrodes [15, 18].

Phosphene appearance has been best characterized in epiretinal implants, where inadvertent activation of nerve fiber bundles (NFBs) in the optic fiber layer of the retina leads to elongated phosphenes [4, 39] (Fig. 2, center). To this end, Granley et al. [20] developed a forward model to predict phosphene shape as a function of both neuroanatomical parameters (i.e., retinal location of the stimulating electrode) and stimulus parameters (i.e., pulse frequency, amplitude, and duration). With this model, phosphenes are primarily characterized by two main parameters, $\rho$ and $\lambda$, which dictate the size and elongation of the elicited phosphene, respectively (Fig. 2, right). Stimulation from multiple electrodes is nonlinearly integrated into a combined perception, and if two electrodes happen to activate the same NFB, they might not generate two distinct phosphenes. In addition, $\rho$ and $\lambda$ vary drastically across patients [4], making stimulus encoding a particularly challenging problem.

3 Related Work

The conventional ‘naive’ encoding strategy simply sets the amplitude of each electrode to the brightness of the corresponding pixel in the target image [9, 29, 42], essentially making the stimulus a downsampled version of the target. Although simple, this encoding strategy only works with near-linear forward models and cannot account for phosphene distortions encountered by real prosthesis patients [4, 15]. When used with a more sophisticated forward model, this strategy often produces unrecognizable percepts (Fig. 2, right).

To provide an alternative, Shah et al. [40] used a greedy approach to iteratively select the stimuli that best recreate a desired neural activity pattern over a given temporal window, assuming that the brain would integrate them into a coherent visual percept. Ghaffari et al. [19] used a neural network surrogate model combined with an interior point algorithm to optimize for localized, circular neural activation patterns for individual electrodes. Fauvel et al. [16] used human in-the-loop Bayesian optimization to achieve encodings perceptually favored by the subject. Spencer et al. [42] proposed
framing the problem of stimulus encoding as inversion of a forward model of neural activation patterns. To approximate the inverse, their approach either requires simplification or is NP-hard [42]. Furthermore, it cannot predict the perceptual consequences of the predicted neural activity.

van Steveninck et al. [11] proposed an end-to-end optimization strategy with a fixed phosphene model, similar to HNA. However, their approach crucially differs from ours in that they included a secondary deep neural network to post-process the predicted phosphenes. This is a critical limitation, because a low reconstruction loss does not reveal whether a high-fidelity encoder was learned or whether the secondary decoder network simply learned to correct for the encoder’s mistakes. In addition, they used an unrealistic phosphene model that simply upscales and smooths a binary stimulus pattern. It is therefore not clear whether their results would generalize to real visual prosthesis patients.

Relic et al. [38] also utilized the end-to-end approach, but without the secondary decoder network used in [11]. They used a more realistic phosphene model, which accounts for some spatial distortions (e.g., axonal streaks), but not the effects of stimulus parameters. Since including a nontrivial phosphene model in the loop is not straightforward, they instead trained a surrogate neural network to approximate the forward model. We re-implemented Relic’s ‘surrogate’ approach in this paper as a baseline method to compare against, as described in Section 4.

Taken together, we identified three main limitations of previous work that this study aims to address:

1) Unrealistic forward models. Most previous approaches (e.g., [11, 40, 42]) use an overly simplified forward model that cannot account for empirical data [4, 15]. We overcome this limitation by developing (and inverting) a differentiable version of a neurophysiologically informed and psychophysically validated phosphene model [20] that can account for the effects of stimulus amplitude, frequency, and pulse duration on phosphene appearance.

2) Optimization of neural responses. Most previous approaches (e.g., [40, 42]) focus on optimizing neural activation patterns in the retina in response to electrical stimulation (“bottom-up”). However, the visual system undergoes extensive remodeling during blinding diseases such as retinitis pigmentosa [30]. Thus the link between neural activity and visual perception is unclear. We overcome this limitation by inverting a phenomenological (“top-down”) model constrained by behavioral data that predicts visual perception directly from electrical stimuli [4, 20].

3) Objective function. Most previous approaches rely on minimizing mean squared error (MSE) between the target and reconstructed image. Although simple and efficient, MSE is known to be a poor measure of perceptual dissimilarity for images [48] and does not align well with human assessments of image quality [50]. We overcome this limitation by proposing a joint perceptual metric that combines mean absolute error (MAE), VGG, and Laplacian smoothing losses.

4 Methods

Problem Formulation. We consider a system where there is some known forward process $f$ mapping stimuli to responses $f: S \rightarrow R, f(S) \subset R$. In the case of visual prostheses, $f$ may map stimuli to visual percepts. In spinal cord rehabilitation devices, $f$ might map electrical stimuli to neural activation patterns. $f$ may be parameterized by patient-specific parameters $\phi$.

Finding the best stimulus for an arbitrary target response $t \in R$ is therefore equivalent to finding the inverse of $f$. However, since not every response can be perfectly reproduced by a stimulus, the true inverse of $f$ is not well defined. We therefore seek the pseudoinverse (still denoted as $f^{-1}$ for simplicity) instead, which outputs the stimuli that produce the closest response to $t$ under some distance metric $d$:

$$f^{-1}(t, \phi) := \arg \min_{s \in S} d(f(s; \phi), t). \quad (1)$$

This problem is straightforward to solve using an autoencoder approach, where a learned encoder $f^{-1}$ is trained to invert the fixed decoder $f$ (i.e., forward model).

Encoder. We approximate the pseudoinverse $f^{-1}$ with a deep neural network encoder $\hat{f}^{-1}(t, \phi; \theta)$ with weights $\theta$, trained to minimize the distance $d$ between the target image $t$ and predicted image $\hat{t}$:

$$\min_{\theta, \phi \sim p(\phi)} d(t, \hat{t}) \quad (2)$$

4
\[
\hat{t} = f(\hat{\hat{J}}^{-1}(t, \phi; \theta); \phi),
\]  

(3)

where \( \phi \) is sampled from a uniform random distribution spanning the empirically observed range of patient-specific parameters \([4, 20]\).

We use a simple architecture consisting solely of fully connected (FC) and batch normalization (BN) layers. First, the target \( t \) is flattened and input to one FC layer. In parallel, the patient parameters \( \phi \) are input to a BN layer before two hidden FC layers. Then, the outputs of these two paths are concatenated, and the combined vector input through two FC layers, producing an intermediate representation \( x \). Amplitudes are predicted from \( x \) with a FC layer. The amplitudes are then concatenated with \( x \), put through a BN layer, and used to predict frequency and pulse duration, each with a FC layer. The encoder outputs are merged into one stimulus matrix \( \hat{s} \). All layers uses a leaky ReLU activation functions, except for stimulus outputs, which use ReLu activation, enforcing nonnegativity. In total there were 1.4M trainable parameters. We also tried using convolutional, transformers, and other attention layers, but the fully connected architecture performed best.

**Decoder** For our decoder \( f \), we use a reformulated but equivalent version of the model described in \([20]\). We use a square 15 \times 15 array of 150\( \mu \)m electrodes, spaced 400\( \mu \)m apart and centered on the fovea. The size and scale of this device was motivated by similar designs in real epiretinal implants.

\( f \) takes as input a stimulus matrix \( s \in \mathbb{R}^{n \times d} \), where the stimulus on each electrode \( (s_e) \) is a biphasic pulse train described by its frequency, amplitude, and pulse duration. A simulated map of retinal NFBs is used to predict phosphene shape. Following \([4]\), each retinal ganglion cells’ activation is assumed to be the maximum stimulation intensity along its axon. Each axon’s sensitivity to stimulation is assumed to decay exponentially with i) Euclidean distance to the stimulating electrode (radial decay rate, \( \rho \)) and ii) distance to the soma along the curved axon (axonal decay rate, \( \lambda \)). To account for stimulus properties \([20]\), \( \rho, \lambda \), and the per-electrode brightness are scaled by three simple equations \( F_{axis}(s_e, \phi), F_{streak}(s_e, \phi) \), and \( F_{bright}(s_e, \phi) \), respectively.

The brightness of a pixel located at the point \( x \in \mathbb{R}^2 \) in the output image is given by

\[
f(s, \phi) = \max_{a \in A} \sum_{e \in E} F_{bright}(s_e, \phi) \exp \left( \frac{-||x - e||^2}{2\rho^2 F_{axis}(s_e, \phi)} + \frac{-d_s(x, a)^2}{2\lambda^2 F_{streak}(s_e, \phi)} \right)
\]  

(4)

where \( A \) is the cells’ axon trajectory, \( E \) is the set of electrodes, \( \phi = \{\rho, \lambda, \ldots\} \) is a set of 12 patient-specific parameters, and \( d_s \) is the path length along the axon trajectory \([23]\) from \( a \) to \( x \):

\[
d_s(x, a) = \int_a^x \sqrt{A(\theta)^2 + \left( \frac{dA(\theta)}{d\theta} \right)^2} \, d\theta.
\]  

(5)

This model \( f \) can be fit to individual patients; however, it is computationally slow and not differentiable. For more details on these equations, see \([20]\). We therefore considered two approaches.

- **Differentiable Model**: We reformulated equations \([4, 5]\) into an equivalent set of parallelized matrix operations, implemented in Tensorflow \([1]\). Significant efforts were put towards developing a model optimized for XLA engines on GPU, resulting in speedups of up to 5000x compared to the model as presented in \([20]\), enabling large-scale gradient descent. To enforce differentiability, we numerically approximated \( d_s \) using \(|A| = 500 \) axon segments per axon.

- **Surrogate Model**: We also implemented the surrogate approach from \([38]\) as a baseline method, where \( f \) is approximated with another deep neural network \( f_\theta(s; \theta_f) \) with weights \( \theta_f \). To achieve this we generated 50,000 percepts using randomly selected stimuli passed through \( f \) and fit a deep neural network to produce identical images. As \( f \) is highly dependent on patient-specific parameters \( \phi \), we generated new data and fit a separate surrogate model for each \( \phi \) in our experimental set. Specific implementation details of the surrogate are presented in Appendix \([A]\). Our implementation improves upon \([38]\) by using the more advanced phosphene model described above, which accounts for effects of stimulus properties and allows for optimization of stimulus frequency in addition to amplitude.
Metrics To measure perceptual similarity, we use a joint perceptual objective consisting of a VGG similarity term, a mean absolute error (MAE) term, and a smoothness regularization term. The MAE term is given by

$$L_{MAE} = ||t - \hat{t}||_1.$$  \hspace{1cm} (6)

The VGG term aims to capture higher-level, abstract differences between images \[11, 26\]. Both the target image and the reconstructed phosphene are input to a VGG-19 network pretrained for classification on ImageNet \[12\], and the MSE between the activations on a downstream convolutional layer is computed. Let \(V_l\) be a function that extracts the activations of the \(l\)-th convolutional layer for a given image. The VGG loss is then defined as:

$$L_{VGG} = ||V_l(t) - V_l(\hat{t})||_2^2.$$  \hspace{1cm} (7)

We also include a smoothing regularization term. This term imposes a loss on the second spatial derivative of the predicted image. A low second derivative implies that where the target image does change, it changes slowly. We found this encouraged smoother, more connected phosphenes. To approximate the second derivative, we convolve the image with a Laplacian filter \[36\] of size \(k\), denoted by \(\text{Lap}(\cdot, k)\), and compute the mean. The smoothness loss is given by:

$$L_{Smooth} = \frac{1}{||\hat{t}||} \sum_i (\frac{\partial^2}{\partial x^2} \hat{t})_i = \frac{1}{||\hat{t}||} \sum_i \text{Lap}(\hat{t}, k)_i.$$  \hspace{1cm} (8)

Our final objective is the weighted sum of the three individual losses, given by Eq. 9 where \(\alpha\) and \(\beta\) are hyperparameters controlling the relative weighting of the three terms:

$$d = L_{MAE} + \alpha L_{Smooth} + \beta L_{VGG}. $$  \hspace{1cm} (9)

We also implement a secondary metric to quantify phosphene recognizability, applicable only for the MNIST reconstruction task. We first pre-train a classifier network on the MNIST targets until it reaches 99% test accuracy, and then fix the weights. The relative accuracy (RA) is then defined as the ratio of the classifiers accuracy on the reconstructed images to its accuracy on the targets \(RA = \frac{ACC}{ACC(t)}\). A perfect encoder would result in \(RA = 100\%\). A similar process was not possible for the COCO task due to the possibility of having multiple objects in each target image.

Training/Optimization We trained using Tensorflow 2.7 \[1\] on a single NVIDIA RTX 3090 with 24GB memory. Stochastic gradient descent with Nesterov momentum was used to minimize the joint perceptual loss. We used a batch size of 16 due to memory limitations imposed by \(f\). The amplitude, frequency predictions are scaled by 2, 20 respectively, while the pulse duration predictions were shifted by 1e-3 prior to being fed through the decoder. This encourages the initial predictions of the network to be in a reasonable range. The Laplacian filter size \(k\) is set to 5. We choose \(l\) to be first convolutional layer in the last block using cross validation (see Appendix D). Similarly, we perform cross validation to find the best values for \(\alpha\) and \(\beta\). Instead of using one value, we found that incrementally increasing the weighting of the VGG loss (\(\beta\)) from 0 while simultaneously decreasing the initially high weight on the smoothing constraint (\(\alpha\)) was crucial for performance, especially when the range of allowed \(\phi\) values was large (see Appendix D).

Datasets We first evaluated on handwritten digits from MNIST \[13\], enabling comparison to previous works \[38\]. Images preprocessing consisted of resizing the target images to the same shape as the output of \(f\) (49x49). We also evaluate on more realistic images of common objects from the MS-COCO \[27\] dataset. We selected a subset of 25 of the MS-COCO object categories deemed more likely to be encountered by blind individuals (e.g. people, household objects), and use only images that contain at least one instance of these objects. We further filter out images by various other criteria, such as being too cluttered or too dim. This process results in a total of approximately 47K training images and 12K test images. See Appendix \[C\] for a full description of the selection process.

Natural images often contain too much detail to be encoded with prosthetic vision. While scene simplification strategies exist \[21\], we focus on the encoding algorithm, so we simply used the ground-truth segmentation masks to segment out the objects of interest. The images were then converted to grayscale, and resized to (49, 49).
Figure 3: Reconstructed MNIST targets for HNA, surrogate, and naive encoders across 4 specific simulated patients. Note that the brightness of the naive encoder is clipped for display.

Table 1: MNIST performance

| Encoding | $\rho=150$ | $\lambda=150$ | $\rho=150$ | $\lambda=1500$ | $\rho=800$ | $\lambda=100$ | $\rho=800$ | $\lambda=1500$ |
|----------|-----------|--------------|-----------|---------------|-----------|--------------|-----------|--------------|
|          | Joint Loss | MAE | RA | Joint Loss | MAE | RA | Joint Loss | MAE | RA | Joint Loss | MAE | RA |
| Naive    | 1.161     | 0.1855 | 90.3 | 1.442 | 0.214 | 78.1 | 8.152 | 1.500 | 34.8 | 8.780 | 1.726 | 28.8 |
| Surrogate| 2.509     | 0.1351 | 53.8 | 3.118 | 0.2431 | 30.7 | 1.692 | 0.2135 | 19.9 | 1.694 | 0.2237 | 18.1 |
| HNA      | 0.559     | 0.064 | 98.1 | 1.029 | 0.1412 | 89.3 | 0.913 | 0.113 | 95.9 | 0.957 | 0.126 | 94.8 |

5 Results

5.1 MNIST

The phosphenes produced from the HNA, surrogate, and naive encoders on the MNIST test set are shown in Fig. 3 and performance is summarized in Table 1. For each MNIST sample, the target image is input to the encoder, which predicts a stimulus. The stimulus is fed through the true forward model $f$, and the predicted phosphene is shown. Since the surrogate method must be retrained for each $\phi$, results are only shown for 4 simulated patients. Our proposed approach outperformed the baselines across all metrics (see Appendix B for a comparison of stimuli).

5.2 COCO

The phosphenes produced by HNA and the naive encoder for the segmented COCO dataset are shown in Fig. 4. We omit the surrogate results due to its poor perceptual performance on MNIST. Averaged across all $\phi$, HNA had a joint loss of 0.713 on the test set and MAE of 0.1408, while the naive encoder had a joint loss of 1.873 and MAE of 0.2830.

5.3 Modeling Patient-to-Patient Variations

MNIST encoder performance across simulated patients ($\phi$) is shown in Fig. 5. Since the surrogate encoder has to be retrained for each patient, comparison is infeasible. To visualize the effects of changing $\rho$ and $\lambda$ on the produced phosphenes, Fig. 5A shows the result of encoding two example MNIST digits, both using the naive method and our encoder. As $\lambda$ increases, the naive phosphenes appear increasingly elongated, and as $\rho$ increases, the phosphenes become increasingly large and blurry. The phosphenes from HNA are slightly too dim and disconnected at low $\rho$, but are relatively stable across other values of $\rho$ and $\lambda$.

To compare performance across the entire dataset, we computed the average test set loss across the same range of $\rho$ and $\lambda$ (Fig. 5B). The encoder performs well across a wide range of simulated patients, with larger loss only at low $\rho$. The naive method performs well only on a limited set of $\phi$, with small $\lambda$ and $\rho \approx 200$. The naive loss was higher than the learned encoder at every simulated point. Random sampling of $\rho$ and $\lambda$ for each image results in a joint loss of 0.921, MAE of 0.120,
and RA of 94.0\% for HNA, while the naive encoder results in a joint loss of 3.17, MAE of 0.596, and RA of 63.6\%. The same analysis yielded similar results on COCO (Appendix F). An analysis across other parameters is presented in Appendix E.

In order for prosthetic vision to be useful, different instances of the same objects would ideally produce similar phosphenes, allowing for consistent perception. To evaluate whether our model achieves this, we cluster the target images and resulting phosphenes using T-SNE [46] shown in Fig. 5C. The ground truth images form clusters corresponding to the digits 0-9. The phosphenes from our encoder roughly form similar, slightly less separated groupings, whereas the naive phosphenes do not. To ensure that this was not the result of bad T-SNE hyperparameters, we repeated the clustering across different perplexities and learning rates, obtaining similar or worse results.

5.4 Joint Perceptual Error Ablation Study

To show that the joint perceptual metric performs better than any of its individual components, we train models using just the VGG loss (Eq. 7), and just MAE (Eq. 6). Shown are values for $\rho=150$ and $\lambda=600$. As mentioned previously, encoders trained using just VGG loss fail to converge, thus we pretrain the VGG encoder using MAE and smoothing loss, then transition to using only VGG. We do not consider ablating the smoothing term (Eq. 8) because it is simply a regularization term. Fig. 6 shows the phosphenes produced by HNA trained on the joint, VGG-only, and MAE-only loss.

The VGG encoder had a test VGG loss of 4\% lower than the joint model, and the phosphenes it produces have the correct high level perceptual details, but are oversmoothed and blurry. The MAE encoder had a final test MAE of 9\% lower than the joint model. Despite a lower MAE,
reconstructed images appear to be lower quality. The joint model had a RA of 99.0% The VGG encoder had a RA of 95.9%, while the joint model had a RA of 77.6%

6 Discussion

Visual Prostheses  We found that HNA is able to produce high-fidelity stimuli from the MNIST and COCO datasets that outperform conventional encoding strategies across all tested conditions. Importantly, HNA produces phosphenes that are consistent across representations of the same object (Fig 5C), which is critical to allowing prosthesis users to learn to associate certain visual patterns with specific objects. On the MNIST task, HNA produced high quality reconstructions, nearly matching the targets (Figure 3). On the harder COCO task, HNA significantly outperformed the naive encoder, but was still unable to capture all of the detail in the images. In Appendix G, we demonstrate that this is largely due to the implant’s limited spatial resolution and not a fundamental limitation of HNA.

Another advantage of the HNA is that it can be trained to predict stimuli across a wide range of patient-specific parameter values $\phi$, whereas the conventional naive encoder works well only for small values of $\rho$ and $\lambda$. This may be one reason why the naive encoding strategy has been shown to lead to substantial individual differences in visual outcomes [43, 37]. Our results suggest that stimuli produced with HNA may be able to reduce at least some amount of this patient-to-patient variability.

Furthermore, HNA also proved superior to a surrogate forward model. The latter offer an alternative when the forward model is computationally expensive or not differentiable. Understandably, any inaccuracies in the surrogate model will propagate to the learned encoder during training. However, we observed this phenomenon even when the surrogate performed well individually (see Appendix A), suggesting that even for well trained surrogates, the encoder may still learn to exploit the inexact surrogate instead of learning to invert the true model. We suspect that the surrogate method’s inferior performance here compared to [38] can be explained by our larger stimulus search space. Thus, we cannot currently suggest HNA for forward models where a surrogate model must be used, unless the forward model is sufficiently simple or has a small stimulus space.

As expected, both encoders perform well for simulated patients with small axonal current spread ($\lambda$). Interestingly, both encoders do slightly worse with very small radial current spread ($\rho$), suggesting that some amount of phosphene overlap is needed to build shapes. This does align with some previous behavioural results [25], and may have implications for the design of future visual prostheses such as [35], which attempt to minimize current spread altogether.

Broader Impacts While our work is presented in the context of visual prostheses, the HNA framework may apply to any sensory neuroprosthesis where stimulus selection can be informed by a numeric or symbolic forward model. For example, cochlear implants [49] could use HNA to choose stimuli that result in a desired sound, and spinal cord implants [7] could use HNA to find the best way to relay neural signals through a damaged section of the spinal cord. However, HNA might not apply to all neural interfaces, such as systems without a clear neural or perceptual target (e.g., deep brain stimulation for the treatment of Parkinson’s [2]) or closed-loop systems [5, 8].

Limitations Our end-to-end architecture allows for exact optimization of the forward model, but it also imposes limitations on the stability and speed of training. For example, we observed that convergence of the encoder required customized learning rate scheduling, careful tuning of metric hyperparameters, and modification of the encoder output layer initialization. Additionally, including
the forward model in-the-loop required large amounts of memory and compute time. Notably, these limitations are only applicable during training. Although this may be application-specific, we expect any complex forward model to encounter similar issues, hindering ease of implementation.

**Safety Concerns** The use of a deep neural network for stimulus encoding does raise safety concerns. Since we cannot examine the process by which stimuli are chosen, it is possible that an HNA encoder might produce potentially harmful stimuli that could lead to serious adverse events (e.g., seizures). However, this concern is mitigated by the fact that most current neuroprostheses are equipped with firmware responsible for verifying that all stimuli stay within FDA-approved safety limits.

7 Conclusion

In summary, this paper proposes a hybrid autoencoder structure as a general framework for stimulus optimization in sensory neuroprostheses and, as a proof of concept, demonstrates its utility on the prominent example of visual neuroprostheses, drastically outperforming conventional encoding strategies. This may prove a promising solution for a variety of neuroprosthetic technologies.

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL https://www.tensorflow.org/. Software available from tensorflow.org.

[2] Alim Louis Benabid. Deep brain stimulation for Parkinson’s disease. *Current Opinion in Neurobiology*, 13(6):696–706, December 2003. ISSN 0959-4388. doi: 10.1016/j.conb.2003.11.001. URL https://www.sciencedirect.com/science/article/pii/S0959438803001739.

[3] M. Beyeler, G. M. Boynton, I. Fine, and A. Rogem. pulse2percept: A Python-based simulation framework for bionic vision. In K. Huff, D. Lippa, D. Niederhut, and M. Pacer, editors, *Proceedings of the 16th Science in Python Conference*, pages 81–88, 2017. doi: 10.25080/shinma-7f4c6e7-00c.

[4] Michael Beyeler, Devyani Nanduri, James D. Weiland, Ariel Rogem, Geoffrey M. Boynton, and Ione Fine. A model of ganglion axon pathways accounts for percepts elicited by retinal implants. *Scientific Reports*, 9(1):1–16, June 2019. ISSN 2045-2322. doi: 10.1038/s41598-019-45416-4. URL https://www.nature.com/articles/s41598-019-45416-4.

[5] Bardia Bozorgzadeh, Douglas R. Schuweiler, Martin J. Bobak, Paul A. Garris, and Pedram Mohseni. Neurochemostat: A Neural Interface SoC With Integrated Chemometrics for Closed-Loop Regulation of Brain Dopamine. *IEEE Transactions on Biomedical Circuits and Systems*, 10(3):654–667, June 2016. ISSN 1940-9990. doi: 10.1109/TBCAS.2015.2453791. Conference Name: IEEE Transactions on Biomedical Circuits and Systems.

[6] Bingni W. Brunton and Michael Beyeler. Data-driven models in human neuroscience and neuroengineering. *Current Opinion in Neurobiology*, 58:21–29, October 2019. ISSN 0959-4388. doi: 10.1016/j.conb.2019.06.008. URL http://www.sciencedirect.com/science/article/pii/S0959438818302502.

[7] Marco Capogrosso, Tomislav Milekovic, David Horton, Fabien Wagner, Eduardo Martin Moraud, Jean-Baptiste Mignardot, Nicolas Buse, Jerome Gandar, Quentin Barraud, David Xing, Elodie Rey, Simone Duis, Yang Jianzhong, Wai Kin D. Ko, Qin Li, Peter Detemple, Tim Denison, Silvestro Micera, Erwan Bezard, Jocelyne Bloch, and Grégoire Courtine. A brain–spine interface alleviating gait deficits after spinal cord injury in primates. *Nature*, 539(7628):284–288, November 2016. ISSN 1476-4687. doi: 10.1038/nature20118. URL https://www.nature.com/articles/nature20118.

[8] Christopher A. R. Chapman, Noah Goshi, and Erkin Seker. Multifunctional Neural Interfaces for Closed-Loop Control of Neural Activity. *Advanced Functional Materials*, 28(12):1703523, 2018. ISSN 1616-3028. doi: 10.1002/adfm.201703523. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/adfm.201703523. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/adfm.201703523._
[24] Lewis Karapanos, Carla J. Abbott, Lauren N. Ayton, Maria Kolic, Myra B. McGuinness, Elizabeth K. Baglin, Samuel A. Titchener, Jessica Kvansakul, Dean Johnson, William G. Kentler, Nick Barnes, David A. X. Nayagam, Penelope J. Allen, and Matthew A. Petoe. Functional Vision in the Real-World Environment With a Second-Generation (44-Channel) Suprachoroidal Retinal Prosthesis. *Translational Vision Science & Technology*, 10(10):7–7, August 2021. ISSN 2164-2591. doi: 10.1167/tvst.10.10.7. URL https://tvst.arvojournals.org/article.aspx?articleid=2776583 Publisher: The Association for Research in Vision and Ophthalmology.

[25] Justin Kasowski and Michael Beyeler. Immersive Virtual Reality Simulations of Bionic Vision. In *Augmented Humans 2022*, AHs 2022, pages 82–93, New York, NY, USA, March 2022. Association for Computing Machinery. ISBN 978-1-4503-9632-5. doi: 10.1145/3519391.3522752. URL https://doi.org/10.1145/3519391.3522752.

[26] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal Style Transfer via Feature Transforms. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/hash/49182f81e6a13cf5eaa466d51fe6406-Abstract.html.

[27] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common Objects in Context. Technical Report arXiv:1405.0312, arXiv, February 2015. URL http://arxiv.org/abs/1405.0312 [cs] type: article.

[28] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[29] Yvonne Hsu-Lin Luo and Lyndon da Cruz. The Argus® II Retinal Prosthesis System. *Progress in Retinal and Eye Research*, 50:89–107, January 2016. ISSN 1350-9462. doi: 10.1016/j.preteyeres.2015.09.003. URL https://www.sciencedirect.com/science/article/pii/S1350946215000701.

[30] Robert E Marc, Bryan W Jones, Carl B Watt, and Enrica Strettoi. Neural remodeling in retinal degeneration. *Progress in Retinal and Eye Research*, 22(5):607–655, 2003. ISSN 1350-9462. doi: https://doi.org/10.1016/S1350-9462(03)00039-9. URL https://www.sciencedirect.com/science/article/pii/S1350946203000399.

[31] David Montes de Oca Zapiain, James A. Stewart, and Rémi Dingreville. Accelerating phase-field-based microstructure evolution predictions via surrogate models trained by machine learning methods. *npj Computational Materials*, 7(1):1–11, January 2021. ISSN 2057-3960. doi: 10.1038/s41524-020-00471-8. URL https://www.nature.com/articles/s41524-020-00471-8 Number: 1 Publisher: Nature Publishing Group.

[32] Craig D. Murray. Embodiment and Prosthetics. In Pamela Gallagher, Deirdre Desmond, and Malcolm MacLachlan, editors, *Psychoprosthetics*, pages 119–129. Springer, London, 2008. ISBN 978-1-84628-980-4. doi: 10.1007/978-1-84628-980-4_9. URL https://doi.org/10.1007/978-1-84628-980-4_9.

[33] Mohammad Amin Nabian and Hadi Meidani. A Deep Neural Network Surrogate for High-Dimensional Random Partial Differential Equations. *Probabilistic Engineering Mechanics*, 57:14–25, July 2019. ISSN 0266-8920. doi: 10.1016/j.probengmech.2019.05.001. URL http://arxiv.org/abs/1806.02957 arXiv:1806.02957 [physics, stat].

[34] Stefanos Nikolopoulos, Ioannis Kalogeris, and Vissarion Papadopoulos. Non-intrusive surrogate modeling for parametrized time-dependent partial differential equations using convolutional autoencoders. *Engineering Applications of Artificial Intelligence*, 109:104652, March 2022. ISSN 0952-1976. doi: 10.1016/j.engappai.2021.104652. URL https://www.sciencedirect.com/science/article/pii/S0952197621004541.

[35] D. Palanker, Y. Le Mer, S. Mohand-Said, and J. A. Sahel. Simultaneous perception of prosthetic and natural vision in AMD patients. *Nature Communications*, 13(1):513, December 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-28125-x. URL https://www.nature.com/articles/s41467-022-28125-x.

[36] Sylvain Paris, Samuel W Hasinoff, and Jan Kautz. Local Laplacian Filters: Edge-aware Image Processing with a Laplacian Pyramid. page 11.

[37] Eli Peli. Testing Vision Is Not Testing For Vision. *Translational Vision Science & Technology*, 9(13):32–32, December 2020. ISSN 2164-2591. doi: 10.1167/tvst.9.13.32. URL https://tvst.arvojournals.org/article.aspx?articleid=2772087 Publisher: The Association for Research in Vision and Ophthalmology.
[38] Lucas Relic, Bowen Zhang, Yi-Lin Tuan, and Michael Beyeler. Deep Learning–Based Perceptual Stimulus Encoder for Bionic Vision. In Augmented Humans 2022, AHs 2022, pages 323–325, New York, NY, USA, March 2022. Association for Computing Machinery. ISBN 978-1-4503-9632-5. doi: 10.1145/3519391.3524034. URL https://doi.org/10.1145/3519391.3524034

[39] J. F. Rizzo, J. Wyatt, J. Loewenstein, S. Kelly, and D. Shire. Perceptual efficacy of electrical stimulation of human retina with a microelectrode array during short-term surgical trials. Invest Ophthalmol Vis Sci, 44 (12):5362–9, December 2003. ISSN 0146-0404 (Print) 0146-0404 (Linking).

[40] Nishal P. Shah, Sasidhar Madugula, Lauren Grosberg, Gonzalo Mena, Pulkit Tandon, Pawel Hottowy, Alexander Sher, Alan Litke, Subhashis Mitra, and E.J. Chichilnisky. Optimization of Electrical Stimulation for a High-Fidelity Artificial Retina. In 2019 9th International IEEE/EMBS Conference on Neural Engineering (NER), pages 714–718, March 2019. doi: 10.1109/NER.2019.8716987. ISSN: 1948-3554.

[41] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. Technical Report arXiv:1409.1556, arXiv, April 2015. URL http://arxiv.org/abs/1409.1556 [cs] type: article.

[42] Martin J. Spencer, Tatiana Kameneva, David B. Grayden, Hamish Meffin, and Anthony N. Burkitt. Global activity shaping strategies for a retinal implant. Journal of Neural Engineering, 16(2):026008, January 2019. ISSN 1741-2552. doi: 10.1088/1741-2552/aaf071. Publisher: IOP Publishing.

[43] Katarina Stingl, Ruth Schippert, Karl U. Bartz-Schmidt, Dorothea Besch, Charles L. Cottrell, Thomas L. Edwards, Florian Gekeler, Udo Greppmaier, Katja Kiel, Assen Koitschev, Laura Kühlewein, Robert E. MacLaren, James D. Ramsden, Johann Roider, Albrecht Rothermel, Helmut Sachs, Greta S. Schröder, Jan Tode, Nicole Troelenberg, and Eberhart Zrenner. Interim Results of a Multicenter Trial with the New Electronic Subretinal Implant Alpha AMS in 15 Patients Blind from Inherited Retinal Degenerations. Frontiers in Neuroscience, 11, 2017. ISSN 1662-453X. doi: 10.3389/fnins.2017.00445. URL https://www.frontiersin.org/articles/10.3389/fnins.2017.00445/full. Publisher: Frontiers.

[44] Gregg A. Tabot, John F. Dammann, Joshua A. Berg, Francesco V. Tenore, Jessica L. Boback, R. Jacob Vogelstein, and Sliman J. Bensmaia. Restoring the sense of touch with a prosthetic hand through a brain interface. Proceedings of the National Academy of Sciences, 110(45):18279–18284, November 2013. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.1221113110. URL https://pnas.org/doi/full/10.1073/pnas.1221113110.

[45] Daniel W. Tan, Matthew A. Schiefer, Michael W. Keith, James Robert Anderson, Joyce Tyler, and Dustin J. Tyler. A neural interface provides long-term stable natural touch perception. Science Translational Medicine, 6(257):257ra138–257ra138, October 2014. doi: 10.1126/scitranslmed.3008669. URL https://www.science.org/doi/10.1126/scitranslmed.3008669. Publisher: American Association for the Advancement of Science.

[46] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. Journal of machine learning Research, 9(11), 2008.

[47] Fabien B. Wagner, Jean-Baptiste Mignardot, Camille G. Le Goff-Mignardot, Robin Demesmaeker, Salif Komi, Marco Capogrosso, Andreas Rowald, Ismael Seäñez, Miroslav Caban, Elvira Pirondini, Molyvan Vat, Laura A. McCracken, Roman Heimgartner, Isabelle Fodor, Anne Watrin, Perrine Seguin, Edoard Paoloes, Katrien Van Den Keybus, Grégoire Eberle, Brigitte Schurch, Etienne Pralong, Fabio Becce, John Prior, Nicholas Buse, Rik Buschman, Esra Neufeld, Niels Kuster, Stefano Carda, Joachim von Zützewitz, Vincent Delattre, Tim Denison, Hendrik Lambert, Karen Minassian, Jocelyne Bloch, and Grégoire Courtine. Targeted neurotechnology restores walking in humans with spinal cord injury. Nature, 563(7729):65–71, November 2018. ISSN 1476-4687. doi: 10.1038/s41586-018-0649-2. URL https://www.nature.com/articles/s41586-018-0649-2.

[48] Zhou Wang and Alan C. Bovik. Mean squared error: Love it or leave it? A new look at Signal Fidelity Measures. IEEE Signal Processing Magazine, 26(1):98–117, January 2009. ISSN 1558-0792. doi: 10.1109/MSP.2008.930649. Conference Name: IEEE Signal Processing Magazine.

[49] Blake S. Wilson, Charles C. Finley, Dewey T. Lawson, Robert D. Wolford, Donald K. Eddington, and William M. Rabinowitz. Better speech recognition with cochlear implants. Nature, 352(6332):236–238, July 1991. ISSN 1476-4687. doi: 10.1038/352236a0. URL https://www.nature.com/articles/352236a0 Number: 6332 Publisher: Nature Publishing Group.

[50] Guangtao Zhai and Xiongkuo Min. Perceptual image quality assessment: a survey. Science China Information Sciences, 63(11):211301, November 2020. ISSN 1674-733X, 1869-1919. doi: 10.1007/s11432-019-2757-1. URL https://link.springer.com/10.1007/s11432-019-2757-1

13
Appendix

A Surrogate Model

This section covers specific implementation details about the surrogate model as well as observations on its performance.

A.1 Implementation Details

Dataset To create training data for the surrogate model \( \hat{f}_\phi \), we used the phosphene model described in [20] and implemented in pulse2percept v0.8 [3]. 50,000 stimuli were created by first selecting a number of electrodes to stimulate between 1 and 30 randomly chosen electrodes, then randomly selecting an amplitude between 1 and 10 (specified as a multiple of the assumed threshold current) and frequency between 1 and 200 Hz for each electrode. In addition, between 10 and 100 electrodes were chosen to act as “noise” electrodes, where either amplitude or frequency was given a nonzero value, but not both. The purpose of these electrodes was for the surrogate model to learn that both a nonzero amplitude and a nonzero frequency are required to produce a visible percept. We used an 80-20 train-test split. As the surrogate model is highly dependent on patient-specific parameters \( \phi \), we generated new data and fit a separate surrogate for each of the following \( \phi \): \((\rho, \lambda) \in \{(150, 100), (150, 1500), (800, 100), (800, 1500)\} \).

Network Architecture The surrogate model \( \hat{f}_\phi \) used a fully-connected architecture. The input to the model was a stimulus matrix \( s \in \mathbb{R}^{n_{\text{ele}} \times 3} \geq 0 \), which was identical to the input to \( f \). The stimulus matrix was split into amplitude and frequency components (pulse duration was not used due to poor model performance), which were fed through a FC layer. The outputs of both FC layers were concatenated and fed through another FC layer. Concurrently, the model computed the element-wise product of the amplitude and frequency components and passed it through a separate FC layer. The outputs of the previous two layers were then concatenated and fed through a final FC layer with output size \( 49 \times 49 \).

The model was trained for 45 epochs using AdamW [28] optimizer and MAE loss.

A.2 Approximating the Forward Model

The surrogate model was able to accurately approximate the true phosphene model \( f \). Table 2 shows MAE over the validation set (10,000 percepts) for all 4 trained \( \hat{f}_\phi \). Visually, the predicted percepts were nearly identical to the ground truth.

| \( \phi \) | \( \rho = 150 \lambda = 100 \) | \( \rho = 150 \lambda = 1500 \) | \( \rho = 800 \lambda = 100 \) | \( \rho = 800 \lambda = 1500 \) |
|-------|-----------------|-----------------|-----------------|-----------------|
| MAE   | 0.0119          | 0.0189          | 0.0078          | 0.0115          |

A.3 Predicted Stimuli

Despite the low surrogate validation error, training with the surrogate model would often result in the encoder suggesting almost adversarial stimuli; that is, stimuli that if fed through the true forward model \( f \) would lead to drastically different percepts than if fed through the surrogate model \( \hat{f} \) (see [1]). With these adversarial-like stimuli, the encoder appears to be performing well under the surrogate model, but performs poorly when the same stimuli are input to the true forward model. We identify this as the primary disadvantage of using a surrogate model and resolving this issue remains an open research problem for end-to-end training with surrogate methods.

We noticed several issues caused by the effects of varying stimulus parameters on phosphene appearance. For example, increasing amplitude increases size and brightness, while increasing frequency increases brightness only. We noticed a larger mismatch between the surrogate and the forward model on the extreme ends of the spectrum (e.g., very high frequency, low amplitude),
resulting in the encoder settling into a minimum that does not exist in the true forward model. It is important to note this disparity appears despite a high training accuracy of the surrogate alone. Although these examples are specific to the bionic vision application, we expect surrogate models derived to describe other neuromodulation technologies to suffer from similar limitations.

Figure A.1: The encoder would often suggest stimuli that lead to drastically different percepts when fed through the surrogate model ($\hat{f}$, middle row) as compared to the true forward model ($f$, bottom row). Examples are shown for $\rho = 800; \lambda = 100$ (left) and $\rho = 800 \lambda = 1500$ (right).


B Predicted Stimuli

Here, we directly examine the stimuli resulting from HNA and naive encoders. Stimuli and their resulting phosphenes for example images from the test set are shown in Figure B.1. The naive encoder produces stimuli with constant frequency (20 Hz) and pulse duration (0.45 ms), which are not shown.

We make the following observations about the predicted stimuli:

- Both encoders activate electrodes corresponding to the shape of the target image. In naive stimuli, the amplitude directly corresponds to the pixel brightness. In HNA stimuli, the distributions of amplitude, frequency, and pulse duration across the electrodes is more complex and harder to characterize, but lead to higher-quality phosphenes.
- HNA uses amplitudes inversely proportional to $\rho$.
- For small $\rho$, HNA primarily uses amplitude to control brightness. For large $\rho$, HNA primarily uses frequency to modulate brightness, keeping amplitudes low to limit phosphene size.
- HNA uses small pulse durations to create lines parallel to the underlying axon NFB (i.e., it utilizes the streaked phosphenes to its advantage), and large pulse durations to create lines perpendicular to the underlying NFB. In other words, HNA was able to exploit application-specific (i.e., neuroanatomical) information that is baked into the forward model.
- On average, HNA uses more electrodes, larger frequencies and pulse durations, and smaller amplitudes than the naive encoder. A large active electrode count and high pulse durations may not be desirable for some prostheses, due to tissue activation and frame rate limits. We found that it was easy to constrain these parameters using regularization on the output stimuli, at the cost of slightly decreased performance.

Figure B.1: Top: Example MNIST target images, and the phosphenes produced by HNA and naive encoders, encoded at various $\rho$ and $\lambda$ values. Center: The stimuli corresponding to the HNA phosphenes. From top to bottom, stimulus frequency (Hz), amplitude (xTh), and pulse duration (ms) are shown. The number of ‘active’ electrodes stimulated above threshold levels is given below each stimuli. Bottom: Stimuli corresponding to the naive phosphenes.
C COCO Dataset

For the COCO task (Section 5.2), we used subset of images from the MS-COCO dataset [27]. MS-COCO was chosen due to its selection of common household objects relevant to the daily life of prosthesis users, as well as availability of ground-truth segmentation masks. To select the images suitable for prosthetic vision, we filtered out images according to the following criteria:

1. **Too cluttered.** Any image with greater than 15 total objects was removed. Removed: 15566
2. **Select chosen objects.** Any image that did not have at least 1 object from the selected categories that was larger than 4% of the total image was removed. Removed: 42289
3. **Too many.** Any image with greater than 5 objects meeting criteria 2 was removed. Removed: 1017
4. **Too dim.** Any objects in the image with average pixel brightness less than 50 were discarded. If this resulted in an image having 0 remaining objects, the image was removed. Removed: 434

This resulted in a total of 47532 training images and 11883 test images (80-20 train-test split). The objects in the remaining images were segmented out using the ground-truth segmentation masks, resized to (49, 49), and converted to grayscale. The distributions of classes used is shown in Figure C.1.

Figure C.1: Number of images (left) or instances (right) of each category in the processed COCO dataset.
D Hyperparameter Selection

In this section, we detail how HNA hyperparameters \((l, k, \alpha, \text{and} \beta)\) were chosen.

**VGG Loss** To choose the layer of the VGG network to use for VGG loss \((l, \text{Eq} [7])\) we performed cross validation across a set of candidate layers. Previous studies [26] have shown that the first layer with each of the 5 convolutional blocks perform well for neural style transfer. Thus, we choose these as our candidate layers. For cross validation, we trained HNA for 50 epochs using each candidate layer. The resulting phosphenes are shown in Figure [D.1]. Using earlier layers, the VGG term performs similarly to MAE, and phosphenes are disconnected. We chose layer 5_1 based on its perceived ability to capture high-level perceptual differences between images, although layer 4_1 also performs similarly.

![Figure D.1: Phosphenes produced by HNA encoder with different layers chosen for VGG loss. Layer 5_1 denotes the first layer within the fifth convolutional block.](image)

**Laplacian Smoothing** We chose to use a kernel size 5 for the Laplacian filter used to estimate the second derivative \((k, \text{Eq. 8})\). The size of the filter controls the scale on which smoothing is applied \((i.e.,\) smaller filters sizes only encourage continuity within a small local region, whereas larger filters encourage continuity within a larger region). Size 5 was chosen because larger filters were observed to over-smooth the image, while smaller filters still led to highly disconnected phosphenes.

**Joint Perceptual Metric** We performed cross validation to find the best values for \(\alpha\) and \(\beta\). Instead of using one value, we found scheduled weighting to be crucial for performance. The scheduler incrementally increased the weight of the VGG loss \((\beta)\) from 0 while simultaneously decreasing the initially high weight on the smoothing constraint \((\alpha)\). This was motivated by the observation that the VGG loss performed poorly during early iterations when the predicted phosphene was near-random.

Under this scheduled weighting strategy, the loss is dominated early on by the MAE and smoothing terms. This encourages the the model to just output reasonable encodings. As training progresses, the predicted phosphenes become higher quality, causing the VGG loss to perform better, and thus the smoothing term is no longer as important.

Additionally, we found it beneficial to temporarily decrease the learning rate by a factor of 10 for a short ‘warm-up’ duration following each increase in \(\beta\), before resetting to 50% of the prior learning rate. This results in the learning rate gradually decreasing throughout training by a factor of around 100. Throughout the paper, we use \(\alpha = 0\) and \(\beta = 0.00008\) for comparisons of loss values.
E Modeling Other Patient-to-Patient Variations

Previously, results were presented across patient-specific parameters $\rho$ and $\lambda$, because these have the greatest impact on phosphene appearance. However, the forward model has a number of other patient-specific parameters, which HNA is also able to adapt to. For full details on all parameters of the forward model, see [20]. Out of the remaining parameters, $a_2$, $a_3$, and $a_5$ are the most impactful on phosphene appearance. $a_2$ and $a_3$ modulate how much the brightness contribution from each electrode scales with increasing amplitude and frequency, respectively. $a_5$ locally scales the global radial current spread $\rho$ based on each electrode’s amplitude. Figure E.1 (left) illustrates the effect of these parameters on phosphene appearance.

Figure E.1 compares HNA to naive encoder performance across $a_2$, $a_3$, and $a_5$. The ranges for these parameters are based on values empirically observed in retinal prosthesis users [20]. HNA produces relatively consistent phosphenes, and outperforms the naive encoder across all conditions.

Figure E.1: Left: Examples of how $a_2$, $a_3$, and $a_5$ affect single-electrode phosphenes. $a_2$ modulates local brightness scaling with increasing amplitude, $a_3$ modulates local brightness scaling with increasing frequency, and $a_5$ modulates local size scaling with increasing amplitude. Center: Phosphenes predicted with HNA and naive encoders for varying $a_2$, $a_3$, and $a_5$, increasing left to right. Right: Plot showing the joint loss across $a_2$, $a_3$, and $a_5$ for HNA (solid) and naive encoder (dashed line).
F COCO Patient-to-Patient Variations

We repeated the analysis presented in Section 5.3 for the COCO dataset. Figure F.1A shows two example COCO images, encoded by both HNA and the naive encoder, across varying $\rho$ and $\lambda$ values. The heatmaps in Figure F.1B show the log of the joint perceptual loss across simulated patients, for both the naive and HNA encoders. To measure phosphene consistency, we performed T-SNE clustering on a subset of the COCO images which have only 1 object. Unfortunately, T-SNE clustering of the ground-truth COCO images did not form groups corresponding to the object types (Figure F.1C), suggesting that the representation of object instances vary drastically across COCO images. Therefore, it was not meaningful to repeat the analysis presented in Fig. 5C.

Similar to the MNIST results presented in Section 5.3, HNA produced higher-quality representations than the naive encoder, resulting in a lower joint loss for every simulated patient. HNA performed consistently well across all simulated patients (Figure F.1B), with a small increase in loss for small $\rho$ ($\rho < 100$). Similar to MNIST, the naive encoder only performs well for patients with a mid-to-low $\rho$ ($\approx 200$) and low $\lambda$.

Figure F.1: COCO Encoder performance across simulated patients (varying $\rho$ and $\lambda$). A: Phosphenes produced by HNA and Naive encoders of two example images. B: Heatmaps showing the log joint loss across $\rho$ and $\lambda$ for HNA and naive encoders. C: Ground-truth COCO images cannot be clustered using T-SNE into groups corresponding to the object types. The clustering was performed on COCO images that only contained one object.
G Simulating Higher-Resolution Implants

On the COCO task, HNA significantly outperformed the naive encoder, but was still unable to capture all of the detail in the images. Two of the main reasons for this are the limited spatial resolution of the implant and the patient-specific distortions from the forward model. Here, we present results from HNAs trained on implants of higher resolution, at small $\rho$ and $\lambda$. The chosen implants are illustrated in Figure G.1A. For a fair comparison, each HNA was trained for only 50 epochs.

Phosphenes resulting from the HNA trained on the different implants are shown in Figure G.1C, and the losses across implants is plotted in Figure G.1B. As implant resolution increases, the phosphenes look increasingly similar to the ground truth, and small details (e.g. facial details, textures) start to emerge.

Thus, HNAs initial failure to capture high-frequency details in the image appears to be an application-specific limitation for visual prostheses more so than a limitation of the HNA framework. For visual prostheses, learning to reconstruct the high-frequency features of complex images despite distortions and limited implant resolution remains an open problem.

Figure G.1: A: The 4 different implants compared. The main text uses the $15 \times 15$ implant. B: The joint perceptual loss of HNAs trained on the different implants after 50 epochs. C: Example images showing the reconstructed phosphenes using each implant