International Journal of Neural Systems, Vol. 30, No. 9 (2020) 2050045 (19 pages)

DOI: 10.1142/S0129065720500458

World Scientific
www.worldscientific.com

Neurolight: A Deep Learning Neural Interface for Cortical Visual Prostheses

Antonio Lozano[*,§], Juan Sebastián Suárez[†,‡,¶], Cristina Soto-Sánchez[†,‡,‖], Javier Garrigós[*,**], J. Javier Martínez-Alvarez[*,††], J. Manuel Ferrández[*,‡‡] and Eduardo Fernández[†,§§]

[*]*Departamento de Electrónica, Tecnología de Computadoras y Proyectos*
Universidad Politécnica de Cartagena, 30202 Cartagena, Spain

[†]*Instituto de Bioingeniería, Universidad Miguel Hernández*
03202 Alicante, Spain

[‡]*CIBER-BBN, 28029 Madrid, Spain*
[§]*antonio.lozano@edu.upct.es*
[¶]*jsuarez@umh.es*
[‖]*csoto@goumh.umh.es*
[**]*javier.garrigos@upct.es*
[††]*jjavier.martinez@upct.es*
[‡‡]*jm.ferrandez@upct.es*
[§§]*e.fernandez@umh.es*

Accepted 8 May 2020
Published Online 21 July 2020

Visual neuroprosthesis, that provide electrical stimulation along several sites of the human visual system, constitute a potential tool for vision restoration for the blind. Scientific and technological progress in the fields of neural engineering and artificial vision comes with new theories and tools that, along with the dawn of modern artificial intelligence, constitute a promising framework for the further development of neurotechnology. In the framework of the development of a Cortical Visual Neuroprosthesis for the blind (CORTIVIS), we are now facing the challenge of developing not only computationally powerful tools and flexible approaches that will allow us to provide some degree of functional vision to individuals who are profoundly blind. In this work, we propose a general neuroprosthesis framework composed of several task-oriented and visual encoding modules. We address the development and implementation of computational models of the firing rates of retinal ganglion cells and design a tool — Neurolight — that allows these models to be interfaced with intracortical microelectrodes in order to create electrical stimulation patterns that can evoke useful perceptions. In addition, the developed framework allows the deployment of a diverse array of state-of-the-art deep-learning techniques for task-oriented and general image pre-processing, such as semantic segmentation and object detection in our system's pipeline. To the best of our knowledge, this constitutes the first deep-learning-based system designed to directly interface with the visual brain through an intracortical microelectrode array. We implement the complete pipeline, from obtaining a video stream to developing and deploying task-oriented deep-learning models and predictive models of retinal ganglion cells' encoding of visual inputs under the control of a neurostimulation device able to send electrical train pulses to a microelectrode array implanted at the visual cortex.

Keywords: Visual neuroprosthesis; neural encoding; computational models; deep learning; artificial vision.

[**]Corresponding author.

1. Introduction

Restoring the ability of the human neural system to function properly is one of the main purposes of neural engineering. In the context of this broad and multidisciplinary research field, encompassing disciplines ranging from clinical neurology to computational neuroscience, scientific advancement and engineering development have brought us to today's achievements: EEG-based Brain Computer Interfaces (BCIs),[1] motor controlled BCIs with Utah arrays,[2,3] cochlear implants,[4] retinal prostheses,[5] and Deep Brain Stimulation systems.[6] Regarding the recovery of visual function, several approaches are being extensively explored such as optogenetics,[7] biocompatible material design for neural interfaces,[8] and neuromorphic computing for neuroprosthesis.[9] Specifically, several advances have been made in retinal prostheses, with a few devices having been already clinically tested or being currently in use.[10,11] These devices are limited to very specific causes of blindness, where the optic nerve function is intact. Cortical prostheses appear as a potential solution to those blindness conditions for people with a functional visual cortex, regardless of their retinal or optic nerve condition for a more extensive review on the state of visual prostheses, we refer the reader to Refs. 12–16. Several research groups around the globe are pursuing this aim.[17–21] In this context, the main goal of this work is to integrate the actual knowledge on neural function, psychophysics, signal processing, and neural encoding modeling to build a working pipeline which will lead us towards further experiments and techniques that can be useful for the development of cortical visual prostheses. A general idea of the complete pipeline of a functional cortical visual prosthesis is composed of a video camera which receives the visual information and conveys it to a signal-processing device which assigns orders to the neurostimulator that sends electrical pulse trains to the neural tissue according to those commands[17,18] (see Fig. 2). Despite notable efforts that have been made in this direction[22,23] — creating potential preprocessing and/or stimulation strategies — a step forward towards the actual implementation of these developments must be made in the frame of reference of today's neurostimulation systems and experimental clinical scenarios.

In the context of the "Development of a Cortical Visual Neuroprosthesis for the Blind" (CORTIVIS) project,[24] we now face the challenge of creating not only powerful tools but also flexible approaches that can overcome the limitations of the current neurostimulation systems, and meet the needs of researchers.

Therefore, inspired by the success of cochlear implants, which greatly benefited from the development and tuning of signal-processing models according to psychophysics,[25] we designed an end-to-end image processing and stimulation control workflow that aimed to constitute a useful tool for visual neuroprosthesis research.

Considering the state of affairs and the great success of Deep Learning (DL) techniques in both industrial applications and neuroscientific and biomedical research, we propose and implement a DL framework that allows the application of several state-of-the-art image processing techniques which can add great value to visual neuroprosthesis, potentially transforming those systems into more flexible devices capable of performing tasks which supersede the mere direct mapping from pixels to activation maps by being able to execute tasks such as object detection and image segmentation.

In order to allow the designed system to incorporate bioinspired control capabilities of the electrical stimulation parameters (amplitude of the phases, pulse width, pulse frequency, inter-pulse, inter-phase, inter-train; see Fig. 2), we created a neural encoding module which makes use of the DL libraries Keras and TensorFlow,[26,27] allowing us to create and make use of custom-defined or data-driven models of neural encoding of light patterns in order to simulate retina-like visual processing models whose output will be used for the stimulation control.

The possibility of reproducing neuronal activity present on the retina during natural vision has been studied by several researchers[23] with promising results. Furthermore, techniques for computing population coding distances on retinas have been proposed,[28] along with visual perception simulation frameworks.[22,29] These results are encouraging, and the new methodologies could be applied to and tested on the visual cortex.

Moreover, diverse models of animal and human neural visual encoding systems have been implemented, targeting different processing stages, some of them focusing on the retina,[30–33] which is the primary stage of visual processing.

The main goals of this work are to propose a conceptual view of the future direction of neuroprosthetics research and the creation of an integrated tool that can be useful for future experiments in blind volunteers.

This work is organized as follows: first, we present our design methodology, describing the main aspects and structure of the complete system and then detailing the development of the neural encoding models, the software and hardware modules, and the designed stimulation strategies.

Last, we discuss the implications of this work along with the potential outcomes and limitations of the system and the future challenges.

2. System Overview

In this work, end-to-end DL-based visual processing and neural encoding system together with a stimulation module has been designed, integrating both hardware and software components into a flexible tool for visual neuroprosthesis research.

This system, schematized in Fig. 2, is composed of several stages. First, a commercial USB camera device mounted on a pair of glasses captures the video signal, which is received by a computer with a Linux operating system. We implemented the system with both a Raspberry Pi model 3B+ in Sec. 5, *Interfacing Computational Neural Models with the Brain: an Artificial Retina* and a Nvidia Jetson Nano in Sec. 6, The input images can be processed by a DL image-processing module and sent to a model's prediction module — in this work, we show the implementation of a Deep Neural Network (DNN)-based object detection model and an artificial retina-see Secs. 5 and 6.

Finally, the model's output is interpreted as the main command for the neural stimulation

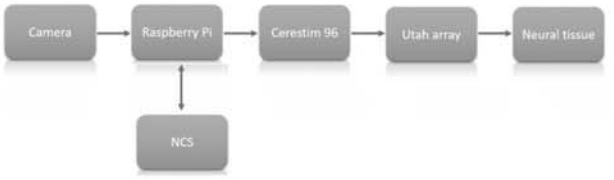

Fig. 1. Illustration of the system's pipeline. For a more detailed explanation of the Neural Compute Stick (NCS) module for DL inference acceleration, we refer to the Sec. 5.3.

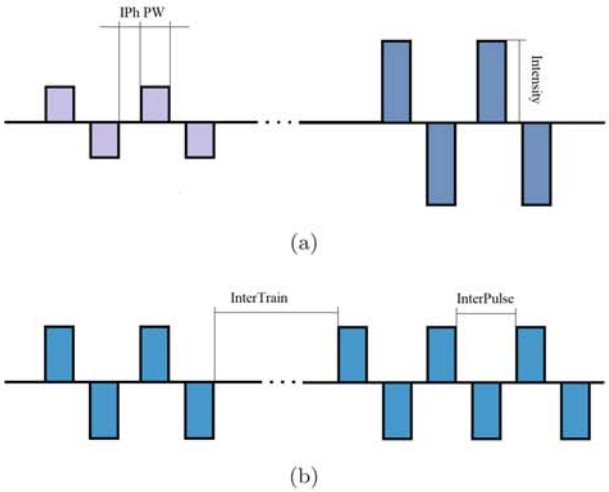

Fig. 2. Two main train modulation strategies are contemplated: Intensity modulation (A) and Frequency modulation (B).

(Cerestim96, Blackrock Microsystems, Inc., Salt Lake City, Utah), which provides customized electrical pulse trains to the visual cortex through intracortical microelectrodes such as the Utah electrode array.[34]

In order to handle the video stream, open source Python libraries for scientific computing have been used: openCV scipy, and numpy,[35–37] along with the state-of-the-art DL libraries TensorFlow and Keras[26,27] as the tools to implement the image processing and the neural coding previous to the stimulation control signals. For the implementation of the retina model in the Raspberry Pi, we explored the possibility of deploying the visual processing models in a specialized DL acceleration hardware device — the NCS[38] which relies on a Vision Processing Unit (VPU).

In Sec. 3, we detail the function and features of the main blocks of the proposed system.

3. Software Interface Design: Modules Organization

In order to provide a modular, easy-to-use, and extendable software corpus, we organized our Python library, named *Neurolight*, in the following structure:

- **Main experiment.** This contains the main thread in which the needed submodules are imported and the camera and stimulation device

configurations are set before launching the "experiment" code.

- **Experiments.** Each experiment is defined as a sequence of common steps: retrieving a new frame from the camera, preprocessing the incoming image (image normalization and resizing), optionally updating the video buffer, performing the model's visual processing, which can consist of DL model inference tasks and/or retina-like signal encoding, adapting the predictions to create stimulation commands, and sending those commands to the stimulator. The main workflow of the experiment module is detailed in Algorithm 1 (an example of an experimental workflow is described in Sec. 5.2).
- **VisionModels.** This module makes it possible to load/define Keras (TensorFlow) DL models, including object detection, face identification, image segmentation, edge detection, and custom retinal processing models. It also contains the retina model's prediction normalization functions, which are necessary to interface the neurostimulator. In addition, it wraps the NCS API functions necessary to compile the designed Tensorflow models in order to deploy them in the VPU.
- **StimAPI.** This module incorporates the main functions which build the necessary messages that allow us to interact with the neurostimulator. These are the basic command blocks used by "StimControl".
- **StimControl.** This is a higher level API that performs the communication operations needed to control the stimulator. Each StimControl function calls StimAPI one or more times in order to create and send functional commands.
- **Utils.** This contains various helper functions, such as generating custom image filters (for example, Gaussian filters) or mapping the desired electrodes to the actual stimulator output channels.

4. A DL Neural Interface: The Case for a DL Powered Visual Neural Prosthesis

4.1. *Overcoming hardware and physical interface design limitations*

When designing a neural prosthesis, the state of the scientific knowledge on the brain physiology and its computational mechanisms, along with the technological state of the art — from electronic design and manufacturing processes to signal processing — shapes the device design decisions.

One of the main critical factors that will have an impact on its performance is the number of electrodes inserted into the neural tissue, which ranges from a single electrode in the case of some Deep Brain Stimulation applications[6,39] to a hundred electrodes contained in a Utah array in the case of motor and visual prostheses,[16,40–43] although the current technological trend is to augment the number of functional electrodes to the order of magnitude of thousands, such as in the system developed by Neuralink.[44]

In all these cases, including the most promising near-future forecasts, the resolution of a neural interface is highly limited when compared to the specificity of the activations in a normal sighted person's neural population.

With this fact in mind, and following the success of devices such as cochlear implants, which can at least partially restore a useful sense of hearing even with a limited number of electrodes,[45,46] an optimization effort in the signal-processing part of the pipeline must be addressed in order to make the best possible use of the current hardware. In this work, we propose and develop a functional pipeline which makes use of this set of DL techniques that act, when necessary, as the signal-processing systems that map the external environment with the electrode activations. Machine Learning techniques, and specifically Artificial Neural Networks constitutes a powerful and

Algorithm 1. Experiments module.

```
Create train configurations
Select electrodes to use
Create encoding model
while processFlag do
  Obtain and process frame
  Update video buffer
  Model processing
  Stimulation command
  Check exitCondition (e.g. timeout, user signal, safety)
  if not exitCondition then
    processFlag ← True
  end if
end while
```

flexible framework for visual signal processing that makes it possible to develop applications in a robust, data-driven way and allows us, as researchers and engineers, to develop and deploy such systems in the highly demanding and dynamic clinical research environment.

4.2. *On the maturity of DL technologies*

When working on image processing tasks, DL models have far outperformed traditional approaches in most applications, yielding high accuracy and robustness in image classification, object detection, and image/instance segmentation, among other problems.

Remarkable efforts are being made to find model architectures which are relatively lightweight in comparison with other deep networks,[47] making them suitable for inference in computing devices with limited resources.

In addition, the rapid development of DL-specific hardware is allowing these models to be deployed in real-time in a progressively more energy-efficient way.[38,48] Model compression and acceleration by using "parameter pruning and sharing, low-rank factorization, transferred/compact convolutional filters, and knowledge distillation"[49] is another field which is yielding practical results.

The research and development in DL techniques for signal processing and pattern recognition is being applied to commercial,[50] industrial,[51,52] agricultural,[53] and biomedical[54,55] fields, among others. In addition, computational neuroscientists are developing their own implementations and techniques based on Artificial Neural Networks for different endeavors, for example, by using Recurrent Neural Networks to infer underlying neural population dynamics and predict behavioral variables in macaque and human motor cortical datasets[56] or making use of Convolutional Neural Networks for modeling several visual pathways.[30,57–60] For a comprehensive work on the integration possibilities of DL and neuroscience, see Ref. 61.

While this kind of technologies seem to be still in a transformation process, where new architectures, model optimization techniques, and associated hardware technologies are rapidly evolving, some of the applications are already robust, especially in the image pre-processing related tasks, which makes them suitable for real-world applications. Our proposal is that DNNs can be successfully applied to several visual neuroprosthesis enhancement tasks that will be presented here in the clinical, prototype, and final stages, and the development of the first DL-powered neural interface framework and prototype is our main contribution.

4.3. *A powerful, flexible framework for visual processing and bioinspired neural coding in basic and clinical research*

When building visual prostheses, several challenges arise, including but not limited to the restricted electrode count mentioned in Sec. 4.1. The ability to *extract the most relevant visual features* from a complex and dynamic visual environment and to transmit them in a meaningful way to the brain, with the aforementioned limited number of electrodes, will be a critical factor in the success of neural prostheses.

These challenges, we claim, can be at least partially tackled by the use of DL models for image segmentation, object detection, emotion recognition, depth estimation, artificial retinas, and the wide range of image-processing models for cortical prostheses, whose limits will depend on the ability of future researchers to figure out the correct combination of preprocessing and image-to-stimuli translation strategies. In Fig. 3, we propose a framework with different stages where several strategies are chosen or combined in order to produce a final stimulus command aimed at generating a meaningful visual perception. In what follows, we explain and comment on several parts of the system.

Complex environment. Given the current state-of-the art of neuroprosthesis technologies and the neuroscientific knowledge at the time of this work, the attempt to encode all the richness of a natural image into extremely accurate, individually selected computational neural units encoding very specific image features is still a work in progress that faces several limitations,[62] although more selective stimulation techniques are being investigated.[63] This challenges blind clinical volunteers and users — independently of their implant location (retinal, lateral geniculate nucleus LGN, visual cortex) — to make sense of a complex and highly dynamic environment

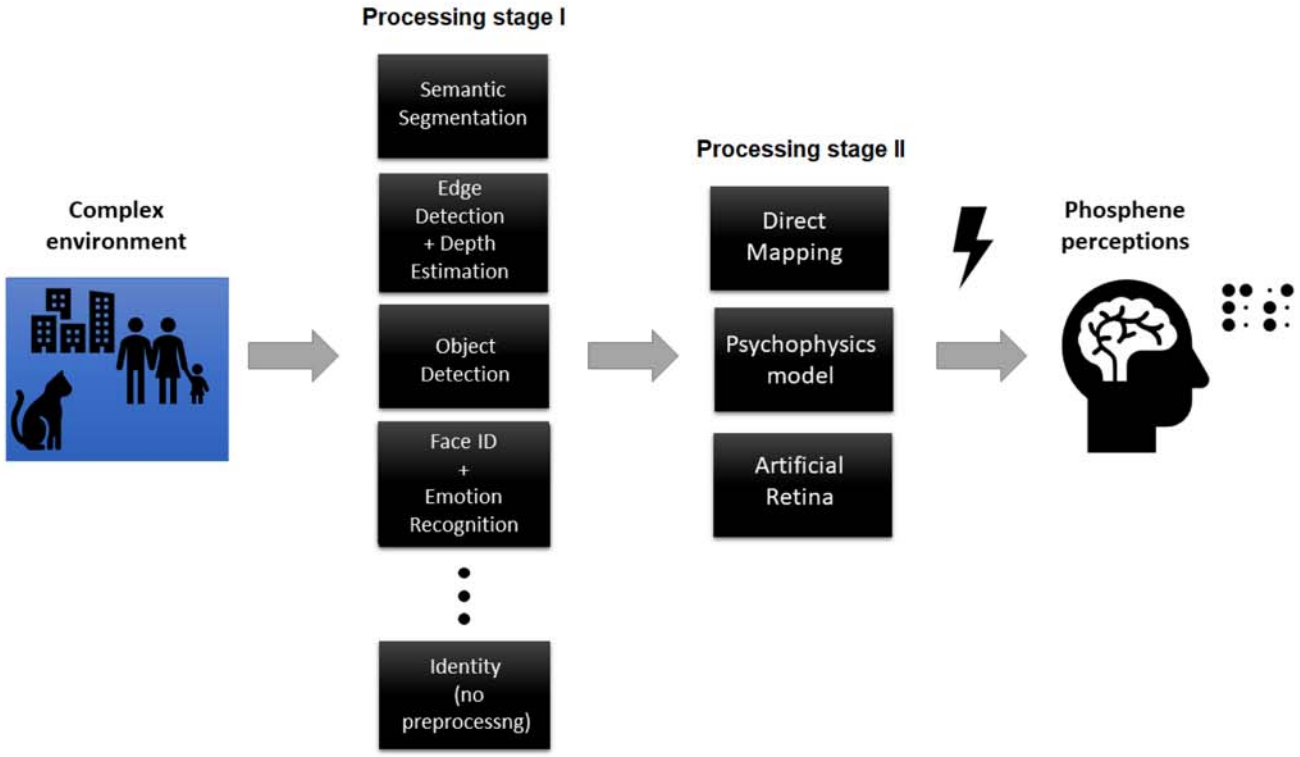

Fig. 3. In order to create a meaningful visual perception by electrical stimulation, we propose a two-stage image-processing and encoding system that follows a first optional task-oriented module and a second neural encoding module.

through phosphene perceptions, this is, visual phenomena experienced which is not caused by the natural processing of light by the visual system starting in the eyes, but from activation of those visual areas by other causes of such nature as mechanical, chemical or electrical.[12] Our goal is to elucidate image processing approaches to generate a series of phosphene perceptions which can maximize the usefulness of the prosthesis.

Processing stage I (task-oriented module). In order to preprocess the visual information in an intelligent way, extracting the most relevant visual features for specific user needs like navigation in different contexts (e.g. city streets or inside home), detecting and recognizing the identity of the people in a room, and so on, existing advanced computer vision models or custom adaptations of them could be used. Some of the possibilities that DL offers for task-oriented image preprocessing are.

Object detection. Visual entities are localized in space and classified in an image/video by object-detection models, which provides a bounding box and a label for multiple objects in parallel. In this way, a blind user with just a few electrodes and a well-defined phosphene map could potentially identify and track specific objects in a room, let's say, to know how many people are present at the time and where are they moving towards.

These models are frequently created with real-time performance in mind and are trained on standardized datasets with a variety of environments.[64,65] Besides, they can be easily retrained to adapt their functioning to new classes/deployment environments. Some of the most successful models at present time are YOLOv3[66] and variants of the SSD.[67]

Edge detection. Although traditional computer vision edge detection techniques are ubiquitous, the parameters of the image-processing algorithm must be properly tuned for the specific application and environmental conditions in which they are going to be applied. However, Convolutional Neural Network-based models, such as the one proposed in *Holistically-Nested Edge Detection*,[68] are able to perform multiscale and multilevel feature learning and to "approach the human ability to resolve the

challenging ambiguity in edge and object boundary detection". These DL models are able to work in a robust way under difficult light conditions and ambiguity and to discard irrelevant information present in the image (e.g. to extract the main contours of a human face, a dog's silhouette, or an open door without adding unnecessary details such as hair texture), significantly reducing the number of electrodes to be activated in the prosthesis, resulting in less injected current, fewer inter-electrode interactions, and a simplified phosphene generation that would make more sense to a blind person, since the model will generate shapes that are comparable to the edges that a human being has previously annotated when generating the model's training dataset.

Semantic/instance segmentation. Semantic segmentation is the ability to classify meaningful — categorical — visual entities at the pixel level. It goes beyond edge detection in the sense that the outputs of these segmentation models contain the edges of a selection of conceptually well-defined objects — for example, a car, door, or person — and thus, in the context of a visual prosthesis for the blind, it is especially relevant with both advantages and disadvantages (see Sec. 7). Instance segmentation models such as Mask R-CNN[69] go one step further and differentiate instances of individual objects at the pixel level; that is, the classified pixels can be grouped into objects in a coherent way.

Face detection and identification. Face detection and face identification models are potentially useful applications for visual prostheses. They have been proven to be fast and accurate, with some of the developments achieving impressive inference time and accuracy results (18 ms while running on a mobile device).[70] Some of the frameworks with developed real-time mobile applications, such as OpenFace,[71] which uses FaceNet[72] for feature extraction and a triplet loss function for face identification, are open and designed to be easy to use and retrain for custom applications.

The output of these and other task-oriented models can be deployed in parallel in a sufficiently capable yet light computer device and combined in a logical way, for example, performing both semantic segmentation and depth estimation[73,74] or salient object detection[75] and thus encoding semantic categories and depth, information that will be carried and transformed into electrical pulses in the next processing phase described below.

Processing stage II (encoding module). One of the main challenges that visual-prosthesis researchers face is the encoding problem: how do we map the desired visual features into stimulation patterns that will generate meaningful perception? While an accurate solution is still lacking, researchers will benefit from having a set of tools that will allow them to hypothesize and test in a clinically relevant environment in a highly flexible way. Several possible approaches for the encoding module are as follows.

Direct mapping. The most straightforward way of performing stimulation is to match the preprocessed (and normalized) image to electrical pulse trains, mapping the intensity (brightness) of the desired image to different levels of pulse amplitude or train frequency (see Fig. 1 for stimulus modulation strategies). Even in this simpler case, a previous electrode remapping procedure will be necessary (see Fig. 6 for remapping).

Psychophysics model. "Simple" models of phosphene brightness/size/persistence are created by performing a set of psychophysical experiments. Psychophysics makes it possible to define a relationship between physical stimulus variables — such as frequency, train duration, and pulse amplitude — and subjective, phenomenological perception. If we can create a sufficiently accurate psychophysical model which takes into account enough perception phenomena (e.g. phosphene brightness and shape, flickering, phosphene fusion, spatiotemporal perceptive interactions due to activation of simultaneous electrodes), a direct mapping between stimuli and perception will be possible. An alternative to the traditional psychophysics models could be the creation of a pulse2percept-like[29] model for a cortical prosthesis. This model would map electrical stimuli to spatiotemporal phosphene patterns experienced by the neuroprosthesis user. In order to convert the desired perceptions (coming from the task-oriented module or directly from the camera input after simple preprocessing) into the desired electrical stimulation commands, an inverse pulse2percept which maps desired perceptions to necessary stimulator commands model should be developed.

Retinal encoding. One of the proposed solutions to the encoding problem is the creation of a bioinspired artificial retina which will tackle both the feature extraction and the electrical stimulus modulation task.[17] The hypothesis holds that neural plasticity will allow the brain to adapt to the incoming signal in a more natural way if this signal is streamed into the brain in a biomimetic way. It is to notice that, when sufficiently understood, other models of visual encoding, including lateral geniculate nucleus and visual cortex computational models can be interfaced with the proposed functional pipeline architecture described in this work. This can be done by simply replacing the retinal encoder described in the following section, which we use as a descriptive demonstration of our interface as done in Refs. 76–78. In any case, the suitability of visual encoding models with any level of complexity, must be assessed functionally in clinical trials.

In the following sections, we describe two of the developed implementations of our proposed framework for visual neural interfaces that incorporates a neural encoding module (Sec. 5): an artificial retina and a task-oriented DNN (Sec. 6): object detection.

5. Interfacing Computational Neural Models with the Brain: An Artificial Retina

In this section, we will describe the ability of Neurolight to be deployed as a neural interface by creating a functional stimulation pipeline which incorporates a biological data-driven retinal ganglion cell model as a retinal image encoder. This model processes the image stream coming from a commercial camera and predicts neural firing rates, which will drive the configuration of the electrical pulse trains sent through the Utah array to the cortical neural tissue.

5.1. *Computational neural models*

Neurolight's Vision Models module makes it possible to define simple and complex retina-like visual preprocessing models, which are defined as Keras Sequential models or TensorFlow Graphs. These models can be deployed using a CPU, GPU, or specialized architectures such as FPGAs. In this first design version, we prepared two modalities: spatial processing models and spatiotemporal processing models, where the defined filters are 2D and 3D,

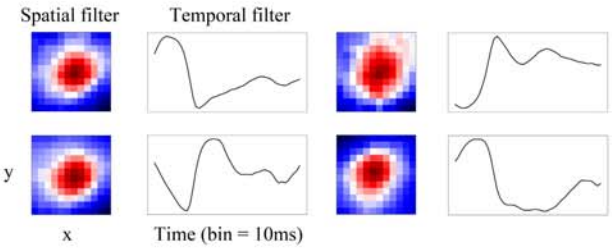

Fig. 4. (Color online) Spatiotemporal filters can be defined and used as part of the pipeline. In the image, a selection of four of the linear part of the LN models after a rank-one decomposition performed with pyret,[79] with the colored box representing the retrieved normalized center-surround spatial filter profiles and the plot traces corresponding to the temporal profiles. The ganglion cell's retinal recordings were performed as mentioned in Ref. 30.

respectively see Fig. 4 for a spatiotemporal visualization of several instances.

Linear–Nonlinear (LN) 2D/3D filters can either be custom-defined filters e.g. spatiotemporal center-surround receptive fields or Gabor-based models or created using data-driven machine-learning techniques. An LN model is equivalent to a 1-layered Convolutional Neural Network (see Fig. 5) and, in a simplistic view, CNNs can be seen as a hierarchical stack of LN models of variable complexity.

In order to demonstrate the naturalistic stimulation control capability of the system, we fitted several ganglion cells' firing responses to light patterns

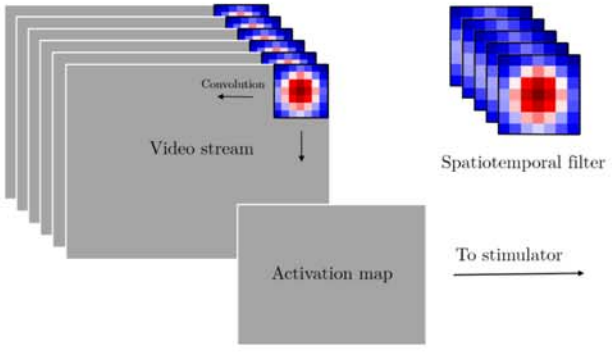

Fig. 5. Illustration of a spatiotemporal filter (LN model) applied to a video stream. The output of the model is an array of activation values that can be mapped to the stimulator's electrodes, after a proper normalization and matching with the pre-configured stimulus configuration values.

of different types full-field light flashes, checkerboard patterns, moving bars, and natural scenes, following procedures similar to those described in Ref. 30. The images of natural scenes were obtained from **Ref. 80 and the rest of the stimuli were created by code scripts.

The ganglion cell's firing rates were fitted by means of a two-stage iterative process in which each single neuron is modeled by means of an L2 regularized spatiotemporal LN process,[81] whose parameters are obtained with the Adam optimizer,[82] which is a variant of Gradient Descent optimization. The loss function utilized was a weighted sum of the mean squared error and the cross entropy between the biological retina's responses and the model's output.

In the first modeling stage, the input of the model consists of the flattened spatiotemporal visual stimulus that is projected into the retina during each time bin, and the output is the smoothed firing rate of the neuron as a response to the input.[30] The discrete time binning is 10 ms.

Due to the high dimensionality of the input (50 pixels×50 pixels×30 frames), the LN models created struggle to converge and are usually suboptimal. In order to tackle this, in the first stage, we used a high regularization factor. This will help the model's parameters to tend to zero in the spatial pixels which are outside the ganglion cell's receptive field, which is convenient in order to figure out which parts of the image are being encoded by the neuron. In the second modeling stage, we centered the model's target around the most relevant $15 \times 15 \times 30$ spatiotemporal pixels for each neuron found by the model (higher nonzero parameter density), thus decreasing the number of parameters from 75,000 to 6750, this is, 11 times fewer parameters, leading the model to a more robust and faster convergence.

Once the LN models of the neurons are created, they can be loaded as the weights of a one-layer Convolutional Neural Network into either a Keras sequential model (for quick deployment in the working pipeline) or a TensorFlow graph, which will make it possible to compile them into a specific graph format to be used by a NCS (see Sec. 5.3).

Both the Keras and TensorFlow libraries allow for application-specific customization by selecting the convolution stride or making spatial or spatiotemporal predictions over batches of images.

5.2. *Neurostimulator control and stimulation strategy*

We developed and implemented a Python version of the Blackrock Microsystems' API for the control of the CereStim96 neurostimulation device (briefly described in Sec. 3.1). This device allows for 16 simultaneous active channels and 15 different pulse train configurations, which can be pre-configured or dynamically created (by overriding previous configurations on demand). After checking the correct operation of the device's current modules, a base pulse train configuration is defined. Then, modified versions of the base pulse trains are created, with different values of pulse intensity, train frequency, and pulse width. These configurations, which shape the pulse trains that will be delivered through the corresponding channels, are defined by the following parameters: Amplitude 1, Amplitude 2, PulseWidth 1, PulseWidth 2, InterPhase, and InterPulse (see Fig. 1). Another key parameter to be taken into consideration is the InterTrain, that is, the time between train pulses.

In this experiment, we select a list of electrodes which will be activated and map them into the actual channels to which the device connects.

After this, the camera configuration parameters are set, taking into consideration the dimensions of the input image and the number of frames to buffer for the spatiotemporal processing. The retina model is defined by loading, reshaping, and normalizing the ganglion cell's LN or any other customized filter based model's weights into a Keras or TensorFlow model which will handle the convolution operations and strides. In the case of using a hardware acceleration device like the NCS for offloading the model's computations, we compile the desired model and load it into the device.

Once the main configurations are set, the main thread starts. Each input frame from the camera device is handled by openCV, resized and normalized, and stored in a buffer variable of the desired length. This buffer will be processed by the retina model, either in the main computer or in the acceleration device. The model's predictions are then normalized between 0 and 1 and matched to the closest of the 15 configurations selected for each electrode, which vary either in the intensity of frequency, depending on the desired train modulation strategy.

Then, a single group stimulation sequence is generated for the corresponding electrodes and configurations and the command is sent to the stimulation device. If desired, a prompt window will be updated, showing the camera input and stimulation information. After checking that the stimulation operation has been properly performed, the next frame is obtained from the camera and the whole process is repeated.

The stimulation strategies must be shaped by decisions of diverse nature related to the actual knowledge of the psychophysics, computational modeling needs, software design decisions, hardware features and limitations. Two main strategies are contemplated currently: amplitude modulation and frequency modulation (see Fig. 1), where the model's predictions for each electrode are mapped to the closest matching configurations, which are previously set and loaded into the stimulator device. Clinical studies will reveal the most optimal way of operating.[40,41,83–85]

In order to generate meaningful perceptions with closely spaced electrode arrays, phosphene mapping techniques may be needed.[86,87] Several mapping strategies are reviewed in Ref. 87.

We programmed a simple routine similar to[88] that allows us to remap the desired spatial patterns to the electrodes that are most likely to produce spatially similar perceptions, based on the Euclidean distances from the desired patterns to the phosphenes that are possible to elicit.

In order to do this, a phosphene map must have been previously generated. For each of the pattern's constituent elements, the electrode that elicited a phosphene that is near the desired one is selected for activation.

An example of the ideal outcome is shown in Fig. 6, where a random phosphene map is generated and the electrodes are selected in such a way that they will create a phosphene pattern similar to the desired one.

This remapping procedure which may be needed in order to enhance the spatial coherence of the elicited phosphenes, can be used both in the case of trying to reproduce the input scene in a reliable way and in the case of using task-oriented DL processing modules for specific goals. As an example of a high-level semantic task, in an object detection task, the user could want to locate an object in space. For the system to elicit a particular phosphene pattern in the spatial location corresponding to the centroid of that detected object, electrode remapping would be used.

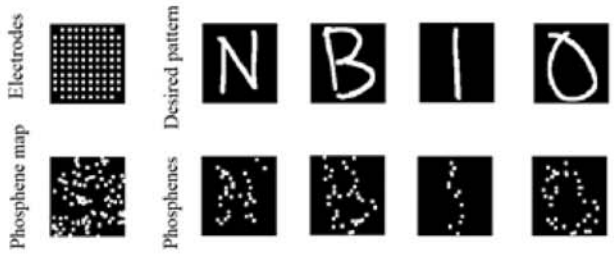

Fig. 6. Phosphene remapping will be needed in order to produce meaningful phosphenes. In the figure, a randomly generated phosphene map is created as a test corresponding to the worst situation possible no coherence between cortical electrode position and phosphene position is created and several meaningful patterns are generated by selecting the electrodes which elicited phosphenes that are the closest to each pattern's element.

5.3. *Hardware implementation: NCS*

In order to explore the possibilities that dedicated hardware acceleration offers, we deployed the vision models in an Intel NCS,[38] a low-power-consumption edge-computing device that is designed to deploy DL models for inference and incorporates an Intel® Movidius^{TM} VPU. This device is connected to the main computer; the preprocessed image/video data are passed to it through a USB port, and the model's predictions are returned after the processing.

After creating LN models of single ganglion cells, they were loaded into a TensorFlow graph. The LN models, as described in Sec. 5.1, consist of a spatial or spatiotemporal filter with a nonlinear activation function which is convoluted through the image/video input, returning the predicted ganglion cell's firing rates, and these predictions can be used for stimulator control after proper normalization and configuration-matching (see Fig. 5). The graph created is prepared for inference-only mode and compiled into a compatible format to be used in the device, up to 14 frames per second in the prepared setup.

6. Interfacing a Deep Neural Network with the Brain: Object Detection

The success of a visual prosthesis will be ultimately measured by the improvement in the quality of life of the people using it. In this line, the proposed task-oriented image processing in Sec. 4, reflected in Fig. 3 could play a major role in the system's usefulness by incorporating relevant information related to the user needs and specific environment, for example, border detection and depth estimation for navigation on streets versus face identification in a familiar environment. In this section, we implement a processing pipeline which incorporates a state-of-the-art DL model used for object detection and interface it with and interface it with 96 intracortical microelectrodes (Utah Electrode Array).

6.1. *A DL model interface*

Neurolight allows the use of a wide array of visual processing models and stimulation control strategies. In Sec. 5, we created an artificial retina and used it to encode visual information. In this section, we can further the potential of Neurolight by interfacing the visual brain with a task-oriented DNN. By incorporating such diverse image-processing capabilities, a blind user could dynamically change the model in order to perform the task needed for each specific context, for example, semantic segmentation, border detection, or depth estimation for navigation or face identification and emotion recognition or pose detection for assistance in social interactions.

Here, we show this concept by using an object detection model able to detect and classify 21 different entities — classes — when trained on the COCO dataset.[89] We then use its last layer's predictions to activate the Utah array's electrodes corresponding with the center of the detected object's position in order to allow the future user to be aware of the presence of two distinct categories person and dog.

The model used was a Single Shot Detector[90] SSD with MobileNetV2[91] as a feature extractor. This feature extractor processes the input image through a series of efficient depth-separable convolution layers with residuals (see the original MobileNetV2 paper for details), which makes it suitable for real-time applications with a competent accuracy comparable to heavier models in several applications such as object detection and semantic segmentation.

The SSD method allows the prediction of "category scores and box offsets for a fixed set of default bounding boxes using small convolutional filters applied to feature maps", which in this case are provided by the MobileNetV2 convolutional layers.

The model is used to detect the presence and position of selected categories present in the training set and can be retrained to detect new categories transfer learning[92] and adapted to the needs to blind users. We describe the use of the model predictions in Sec. 6.2.

6.2. *Neurostimulator control and stimulation strategy*

In this task, two different train pulse configurations were defined: the first one relating to the category "person" and the second relating to "dog". In a clinical setup, these configurations would be first tested with a 2-Alternative Forced Choice paradigm, in which the user discriminates between the two stimuli.

Once the experiment starts, the incoming frames from the camera are preprocessed resized and normalized and fed to the Single Shot Detection model, which returns predictions regarding to the bounding boxes of the detected objects along with the classification of this objects. These predictions are sorted by a confidence threshold, e.g. 80% model's prediction confidence. The middle points of the remaining predicted bounding boxes are calculated and situated in a 10×10 pixel image corresponding to the Utah array electrodes position, see Fig. 7. Whenever an electrode overlapping occurs, that is, two distinct high confidence detections are assigned to the same final electrode, the class with a higher priority person in this case would be selected. After the following electrode remapping, the corresponding stimulator channels are activated with the pre-configured train parameters frequency, pulse width, interphase according to the each object class, and with a pulse intensity corresponding to the phosphene perception threshold corresponding to each electrode that would assure at least a 90% of probability of phosphene perception in single trials -preliminary results in a blind human volunteer can be found in Ref. 93.

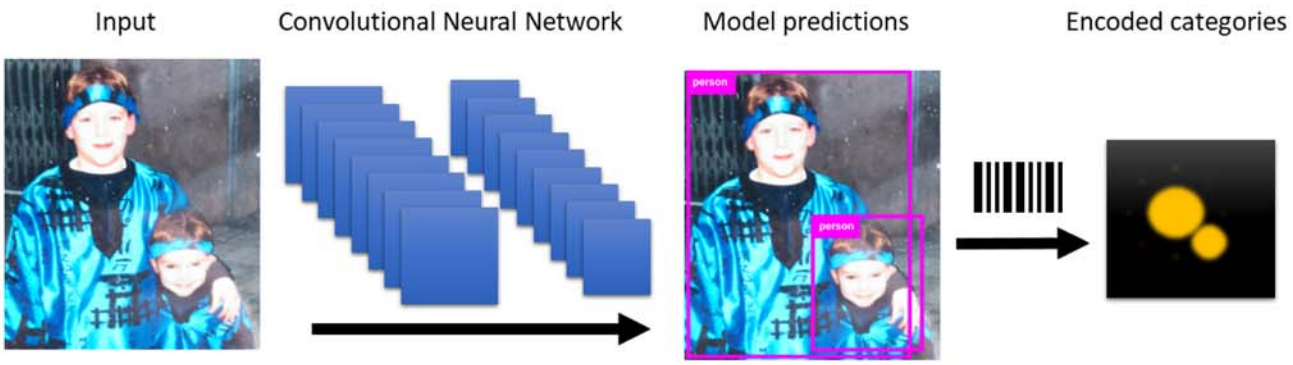

Fig. 7. By using object detection DL models and a simple category coding strategy, a visual prosthesis can convey useful information for a blind user in daily life situations. In the figure, a simple example where the category "person" is encoded as an elementary shape with the size of the generated pattern being proportional to the size of the predicted bounding box.

6.3. *Real-time inference on Nvidia Jetson Nano*

In this implementation, we used the Nvidia Jetson Nano System on Module (SOM), which incorporates a Nvidia Maxwell™ GPU and is able to deliver 472 GFLOPs consuming 5 to 10 W.

This edge-computing device was designed to efficiently perform DL model inference (see Ref. 94 for more technical specifications).

The SSD-MobileNetV2 backend neural network is able to run at 39 FPS on this hardware.[95]

In order to take advantage of the device's high inference speed in these conditions, we implemented the camera stream plus model's detections in parallel to the configuration sending of the commands to the neurostimulator. This way, the electrode activations and the pulse train delivery can be refreshed at 20 Hz in our setup. To elucidate the optimal use of the developed system and its improvement, an experimental framework and specific clinical trials must be carefully designed, which constitutes the next step in our work.

7. Discussion and Future work

Since many questions regarding the psychophysics of the phosphene generation are either still unanswered or in need of more extensive exploration, cortical neurostimulator control tools should be smooth and easy to use and at the same time capable of adaptation to the experimental scenarios.

In this work, a cortical prosthesis control framework prototype endowed with DL capabilities is developed, having at its core the design principles of robustness and flexibility, allowing custom adaptation to the needs of clinical research. This functional working pipeline also makes it possible to incorporate both simple and complex computational neural encoding models of visual inputs (such as data-driven or custom LN or CNN models of retinal ganglion cells) for prosthesis control and to define different stimulation strategies, such as amplitude and frequency modulation, based on the implemented model's output. This framework has been designed and implemented in an experimental setup with a commercial neurostimulator that can be used in both animal and human research. The core pipeline modules, such as the image capture and preprocessing, the DL models for task-oriented processing, the neural encoding module, and the stimulator control API, are based on open-source Python libraries commonly used by the scientific community, which we believe is a fundamental feature for scientific tools and knowledge sharing.

As a proof of concept for our proposed visual neuroprosthesis conceptual framework, which incorporates two signal processing stages a task oriented, DL assisted image preprocessing and a neural encoding stage, we implemented two of these systems as real prototypes.

Our first implementation consists of an artificial retina consisting of a spatiotemporal LN ganglion cell model with a Rectifier Linear Unit as an activation function, which is equivalent to a 1 layered Convolutional Neural Network with temporal channels instead of RGB channels spatiotemporal processing present in retina models developed in Refs. 30–33 and others.

The weakness of this implementation is the fact that it mainly uses a sequential processing flow: every new image/video stream is preprocessed and the model's inference is performed before encoding the commands and sending them to the stimulator. After that, the stimulator sends the pulse trains, and then finally a new image can be fetched. With this operation mode, the system is able to change the running stimulation parameters at 14 FPS with short pulse trains.

The fact that the electrical pulse trains are sent after the image processing-model prediction stage leads to a blinking stimulation strategy: train pulses are interleaved with an inter-train resting period. This type of stimulation of the visual cortex has been tested previously[84] and prevents the neural tissue from being permanently under the influence of external electrical fields, although the implications of this stimulation strategy have yet to be elucidated. In this matter, the inter-train interval necessary to generate a separated or continuous phosphene will be one of the main features of study in the clinical phases, along with the effects of temporal summation, phosphene size, and brightness dynamics.

This bottleneck caused by the sequential model's inference and electrode configuration setting can be avoided by using a *multiprocessing* approach, allowing for independent and parallel processing of every stage (image processing-encoding and stimulation commands), which is the strategy adopted in our second implementation, which consists of a DNN (a Single Shot Detector) that allows for automatic object detection in real time. This allows Neurolight to obtain a better performance in terms of FPS we were able to re-configure and reallocate and electrode activations up to the rate of 20 Hz at the present moment. The two signal processing and stimulator control algorithms were implemented with different hardware platforms (NCS with a Raspberry Pi 3B+ and a Nvidia Jetson Nano), in order to show the flexibility of the proposed framework.

One of the main challenges faced by a visual-prostheses designer is how to convey the most useful information through the neural prosthesis: Neurolight incorporates a set of tools that could help elucidate the effect of different image-encoding strategies and the usefulness of several task-oriented image processing techniques in both clinical environments and daily life activities when portable neurostimulation hardware (or virtual reality setups) are used.

Since future extensive clinical trials are yet to come, we deliberately designed and kept this framework flexible enough so that clinicians and researchers would be able to experiment with or use different combinations of its processing modules.

Therefore, if the research team (the user, in the future) aims to elicit phosphene perceptions without any task-oriented support, skipping this way any semantic preprocessing, Processing Stage I (see Fig. 3) can be ignored, which is equivalent to using the "Identity" block shown in the figure. Then, Preprocessing Stage II neural encoding and stimulator control will be the main active modules, and the optimal outcome of this way of functioning would be to generate a visual perception which resemblances to the real environment with as high fidelity as possible, with optional image enhancement preprocessing models deployment such as Edge Detection or Depth estimation.

If task-driven is chosen, both Processing Stage I and Processing Stage II will be active. As an example for, let's say, object detection, the desired categories of objects will be recognized and detected in real time by a DNN to be encoded and appear as pre-defined phosphene patterns with differentiable characteristics in space and time which will allow the user to recognize and locate them. The most effective way to encode an object category to allow a maximum number of patterns to differentiate will be one of the challenges that we face in the clinical trials stage.

One of the challenges in this work was to show how the proposed general framework can be deployed in devices which are accessible to any laboratory (see NCS, Raspberry Pi or Nvidia Jetson Nano which can be purchased easily and for a price within reach of any research team). These details are also important to show that the computing power needed is not extremely high for the current technology, and hardware devices designed with DL model inference in mind, such as the NCS and the Nvidia Jetson SOCs, allow us to deploy these vision systems in real time with low power consumption and high flexibility in terms of model architectures. We believe these are useful features in order to build a neuroprothesis that aims to be useful in the future with the current technology.

The strategy proposed in the DL approach is to simplify the transmitted visual information in a meaningful way such that the complexity of the environment is diminished without disregarding the important information necessary for understanding the scene, improved navigation, and social interactions, thus augmenting the capabilities of a neural interface independently of its number of electrodes.

Among of the limitations of this approach is that, by using task-driven models (i.e. semantic segmentation or object detection) that can augment the capabilities of the visual prosthesis to be useful in certain environments and situations of daily life, we are at the same time imposing a "sensory" and a "conceptual" bottleneck, that is, limiting the information received by the user from the external world. In addition, DL models which are insufficiently robust to generalize in different environments and conditions could fail to be useful, for example by providing false positives or negatives or dynamically unstable predictions. Objects or categories which are unrecognizable by the neural networks will not be encoded by the system. This issue is only one of many that imposes a big responsibility on the use of AI in neural interfaces, and ethical debate becomes not only essential but mandatory. For a related comment on the implications of the actual and future developments in neurotechnology, see Ref. 96.

The advantage of this object detection system is that it detects categorically perfectly defined objects, so the electrical stimulus conveyed to the brain can be modified (e.g. a certain blinking frequency or current) in a way that it gives a hint to the prosthesis wearer of what he/she is seeing and thus helping object recognition.

Another important matter is that DNNs are developed for tasks that do not necessarily directly match our needs, but for other research or commercial applications. This observation is far from pointing towards an unsolvable challenge, since vision models are relatively easy to retrain on custom datasets to solve specific tasks by transfer learning. A recent example of advances in this direction is in Ref. 97, where several Convolutional Neural Neworks are combined "*...for extracting and conveying relevant information about the scene such as structural informative edges of the environment and silhouettes of segmented objects*", and their performance is evaluated in sighted subjects using simulated prosthetic vision. The ground truth for the models and their evaluation procedures should be designed to address blind people's needs in collaboration with experts in rehabilitation and navigation training. The ability of different image-processing strategies should be assessed according to established procedures such as that in Ref. 98, although a consensus on standardization must be achieved in the field. In addition to the assessment of the system's performance, new rehabilitation paradigms should be developed along with the different task-oriented and encoding frameworks.

Related to this goal-oriented use of DL, great work in the line of assistive navigation systems for visually impaired people in this case, peripheral vision loss have been done in Ref. 99, where a real time object detector is used along with a Kalman Filter based object tracker and a motion model to feed a hazard estimation Neural Network. Then, "*the provided hazard type is then translated into a smart notification to increase the user's cognitive perception using the healthy vision within the visual field*". Although real world evaluation with human subjects it's still on its way, this constitutes a good example of how this technologies are potentially useful for blind people in real, daily life environments. In addition, the use of virtual reality,[100] visual prostheses phosphene simulation see Ref. 101 for an emotion recognition simulation application along with log-polar for filter based feature extraction strategies and log-polar aware model of complex motion integration across the visual field being produced.[102,103] Together with signal processing advances, the design of efficient neuromorphic hardware chips[104] and their applications on the control robotic motor and cognitive systems[105,106] are fields worth exploring when considering the design of a future prosthesis.

Along with the use of DL models to augment the capabilities of a prosthesis with a limited electrode count and artificial retinas to encode the visual information in a more bioinspired way or data-driven psychophysical models adapted to each user, another element into the cortical prosthesis has been proposed recently, named neural co-processors[107] which would perform motor prosthetic device control by means of a combination of neural encoding and decoding models.

Although this encoding–decoding approach seems not to be feasible at the present moment at least regarding to cortical visual prostheses it

constitutes an interesting theoretical concept to contemplate.

An interesting example of the study of the conscious report of visual percepts by awake monkeys by means of measuring neural activity at the level of different brain stages including the dorsolateral prefrontal cortex (dlPFC) can be found at Ref. 108.

In future works, we expect to implement several psychophysics modules which complement the tool and allow for a better fine-tuning of the whole system. Regarding the visual-encoding models, it is hypothesized that retina-like image preprocessing could be beneficial for visual prostheses[17] by performing a bioinspired feature extraction of visual information, although this remains unanswered. Along with the technical achievements made, new experiments need to be designed accordingly to provide answers. In this way, more complex CNN-RNN-based retina models which are proven to mimic the retinal encoding will be compiled and tested and a tradeoff between model complexity and overall system performance in terms of computing speed will be extensively studied.

In addition, regarding future clinical studies and the evaluation of the usefulness of the task-oriented image processing modules proposed, a conceptual framework for clinical experiments design must be performed along with rehabilitation procedures that will be key in the human-neuroprosthesis interaction improvement.

Finally, although more research is still need, we hope that this work constitutes a step forward to integrating knowledge from many scientific and engineering fields towards the development of a cortical visual neuroprosthesis for the blind. It is carried out under the aim that many neural engineers dream of: to help people achieving a level of neural function recovery sufficient to improve their quality of life.

Acknowledgments

This work is partly supported by Fundación Séneca Agencia de Ciencia y Tecnología de la Región de Murcia, under projects 20041/GERM/16 and 21091/PDC/19; by grant RTI2018-098969-B-100 from the Spanish Ministerio de Ciencia Innovación y Universidades; by grant PROMETEO/2019/119 from the Generalitat Valenciana and by the Bidons Egara Research Chair of the University Miguel Hernández.

References

1. J. R. Wolpaw, N. Birbaumer, W. J. Heetderks, D. J. McFarland, P. H. Peckham, G. Schalk, E. Donchin, L. A. Quatrano, C. J. Robinson and T. M. Vaughan, Brain-computer interface technology: A review of the first international meeting, *IEEE Trans. Rehabil. Eng.* **8**(2) (2000) 164–173.
2. T. S. Davis, H. A. C. Wark, D. T. Hutchinson, D. J. Warren, K. O'Neill, T. Scheinblum and G. A. Clark, Restoring motor control and sensory feedback in people with upper extremity amputations using arrays of 96 microelectrodes implanted in the median and ulnar nerves, *J. Neural Eng.* **13**(3) (2016) 036001.
3. P. Nuyujukian, J. S. Albites, J. Saab, C. Pandarinath, B. Jarosiewicz, C. H. Blabe, B. Franco, S. T. Mernoff, E. N. Eskandar, J. D. Simeral, L. R. Hochberg, K. V. Shenoy and J. M. Henderson, Cortical control of a tablet computer by people with paralysis, *PLoS One* **13**(11) (2018) 1–16.
4. W. F. House, Cochlear implants, *Otol. Rhinol. Laryngol.* **85**(27) (1976) 1–93.
5. J. D. Weiland, W. Liu and M. S. Humayun, Retinal prosthesis, *Annu. Rev. Biomed. Eng.* **7**(1) (2005) 361–401.
6. H. S. Mayberg, A. M. Lozano, V. Voon, H. E. McNeely, D. Seminowicz, C. Hamani, J. M. Schwalb and S. H. Kennedy, Deep brain stimulation for treatment-resistant depression, *Neuron* **45**(5) (2005) 651–660.
7. A. Sengupta, A. Chaffiol, E. Macé, R. Caplette, M. Desrosiers, M. Lampič, V. Forster, O. Marre, J. Y. Lin, J. A. Sahel, S. Picaud, D. Dalkara and J. Duebel, Red-shifted channel rhodopsin stimulation restores light responses in blind mice, macaque retina, and human retina, *EMBO Mol. Med.* **8**(11) (2016) 1248–1264.
8. P. Fattahi, G. Yang, G. Kim and M. R. Abidian, A review of organic and inorganic biomaterials for neural interfaces, *Adv. Mater.* **26**(12) (2014) 1846–1885.
9. S. Vassanelli and M. Mahmud, Trends and challenges in neuroengineering: Toward intelligent neuroprostheses through brain inspired systems communication, *Front. Neurosci.* **10** (2016) 438.
10. L. da Cruz, D. J. Dorn, M. S. Humayun, G. Dagnelie, J. Handa, P. O. Barale, J. A. Sahel, P. E. Stanga, F. Hafezi, A. B. Avinoam, J. Salzmann, A. Santos, D. Birch, R. Spencer, A. V. Cideciyan, E. de Juan, J. L. Duncan, D. Eliott, A. Fawzi, L. C. Olmos de Koo, A. C. Ho, G. Brown, J. Haller, C. Regillo, L. V. Del Priore, A. Arditi and R. J. Greenberg, Five-year safety and performance results from

the argus II retinal prosthesis system clinical trial, *Ophthalmology* **123**(10) (2016) 2248–2254.
11. R. Hornig, M. Dapper, E. L. Joliff, R. Hill, K. Ishaque, C. Posch, R. Benosman, Y. LeMer, J. A. Sahel and S. Picaud, *Pixium Vision: First Clinical Results and Innovative Developments* (Springer, Cham, 2017).
12. E. Fernandez, F. Pelayo, S. Romero, M. Bongard, C. Marin, A. Alfaro and L. Merabet, Development of a cortical visual neuroprosthesis for the blind: The relevance of neuroplasticity, *J. Neural Eng.* **2**(4) (2005) 1–12.
13. P. M. Lewis, H. M. Acklandm, A. J. Lowery and J. V. Rosenfeld, Restoration of vision in blind individuals using bionic devices: A review with a focus on cortical visual prostheses, *Brain Res.* **1595** (2015) 51–73.
14. G. Dagnelie, Visual prosthetics 2006: Assessment and expectations, *Expert Rev. Med. Devices* **3**(3) (2006) 315–325.
15. A. Barriga-Rivera, L. Bareket, J. Goding, U. A. Aregueta-Robles and G. J. Suaning, Visual prosthesis: Interfacing stimulating electrodes with retinal neurons to restore vision, *Front. Neurosci.* **11**(620) (2017) 620.
16. R. M. Mirochnik and J. S. Pezaris, Contemporary approaches to visual prostheses, *Mil. Med. Res.* **6**(1) (2019) 1–9.
17. R. A. Normann, B. A. Greger, P. House, S. F. Romero, F. Pelayo and E. Fernandez, Toward the development of a cortically based visual neuroprosthesis, *J. Neural Eng.* **6**(3) (2009) 35001.
18. W. H. Dobelle, Artificial vision for the blind by connecting a television camera to the visual cortex, *ASAIO J.* **46**(1) (2000) 3–9.
19. P. Troyk, M. Bak, J. Berg, D. Bradley, S. Cogan, R. Erickson, C. Kufta, D. McCreery, E. Schmidt and V. Towle, A model for intracortical visual prosthesis research, *Artif. Organs* **7**(11) (2003) 1005–1015.
20. A. J. Lowery, Introducing the Monash vision group's cortical prosthesis, in *Proc. IEEE Int. Conf. Image Processing*, eds. D. Taubman and M. Wu, Melbourne, Australia, 2013, pp. 1536–1539.
21. Early Feasibility Study of the Orion Visual Cortical Prosthesis System. ClinicalTrials.gov. Identifier: NCT03344848, https://clinicaltrials.gov/ct2/show/NCT03344848.
22. J. R. Golden, C. Erickson, N. P. Cottaris, N. Parthasarathy, F. Rieke, D. H. Brainard, B. A. Wandell and E. J. Chichilnisky, Simulation of visual perception and learning with a retinal prosthesis, *J. Neural Eng.* **16**(2) (2019) 025003.
23. L. H. Jepson, P. Hottowy, G. A. Weiner, W. Dabrowski, A. M. Litke and E. J. Chichilnisky, High-fidelity reproduction of spatiotemporal visual signals for retinal prosthesis, *Neuron* **83**(1) (2014) 87–92.
24. Development of a Cortical Visual Neuroprosthesis for the Blind (CORTIVIS). ClinicalTrials.gov. Identifier: NCT02983370, https://clinicaltrials.gov/ct2/show/NCT02983370.
25. R. V. Shannon, A model of threshold for pulsatile electrical stimulation of cochlear implants, *Hear. Res.* **40**(3) (1989) 197–204.
26. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. Corrado, A. Davis, J. Dean, S. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, W. Jia, R. Jó Zefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu and X. Zheng, TensorFlow: Large-scale machine learning on heterogeneous distributed systems, in *Computer Science — Distributed, Parallel, and Cluster Computing, Computer Science — Learning*, 2015, https://www.tensorflow.org/about/bib.
27. F. Chollet, *Keras* (2015), https://github.com/fchollet/keras.
28. N. P. Shah, S. Madugula, E. J. Chichilnisky, J. Shlens and Y. Singer, Learning a neural response metric for retinal prosthesis (2018), doi:10.1101/226530.
29. M. Beyeler, G. Boynton, I. Fine and A. Rokem, pulse2percept: A Python-based simulation framework for bionic vision, in *Proc. of the 16th Python in Science Conf.*, eds. K. Huff, D. Lippa, D. Niederhut and M. Pacer, Austin, United States, 2017, pp. 81–88.
30. A. Lozano, C. Soto-Sanchez, J. Garrigós, J. J. Martínez, J. M. Ferrández and E. Fernández, A 3D convolutional neural network to model retinal ganglion cell's responses to light patterns in mice, *Int. J. Neural Syst.* **28**(10) (2018) 1850043.
31. R. Crespo-Cano, A. Martínez-Álvarez, A. Díaz-Tahoces, S. Cuenca-Asensi, J. M. Ferrández and E. Fernández, On the automatic tuning of a retina model by using a multi-objective optimization, in *Artificial Computation in Biology and Medicine*, Elche, Spain, 2015, pp. 108–118.
32. L. Mcintosh, N. Maheswaranathan, A. Nayebi, S. Ganguli and S. Baccus, Deep learning models of the retinal response to natural scenes, in *Advances in Neural Information Processing Systems*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon and R. Garnett (eds.), Vol. 29, Barcelona, Spain, 2016, pp. 1369–1377.
33. Q. Yan, Y. Zheng, S. Jia, Y. Zhang, Z. Yu, F. Chen, Y. Tian, T. Huang and J. K. Liu, Revealing fine structures of the retinal receptive field by deep learning networks, arXiv:1811.02290.
34. E. M. Maynard, C. T. Nordhausen and R. A. Normann, The utah intracortical electrode array: A recording structure for potential brain-computer

interfaces, *Electroencephalogr. Clin. Neurophysiol.* **102**(3) (1997) 228–239.
35. G. Bradski, The openCV library, *Dr. Dobb's J. Softw. Tools* **25** (2000) 120–125.
36. E. Jones, T. E. Oliphant, P. Peterson *et al.*, *SciPy: Open Source Scientific Tools for Python* (2001), http://www.scipy.org/.
37. T. E. Oliphant, *Guide to NumPy* (Massachusetts Institute of Technology, 2006).
38. Intel's Neural Compute Stick, https://movidius.github.io/ncsdk/ncs.html.
39. M. L. Kringelbach, N. Jenkinson, S. L. F. Owen and T. Z. Aziz, Translational principles of deep brain stimulation, *Nat. Rev. Neurosci.* **8** (2007) 623–635.
40. T. S. Davis, R. A. Parker, P. A. House, E. Bagley, S. Wendelken, R. A. Normann and B. Greger, Spatial and temporal characteristics of V1 microstimulation during chronic implantation of a microelectrode array in a behaving macaque, *J. Neural Eng.* **9**(6) (2012) 65003.
41. E. Fernandez, Development of visual neuroprostheses: Trends and challenges, *Bioelectron. Med.* **4**(1) (2018) 12.
42. A. B. Schwartz, X. T. Cui, D. J. Weber and D. W. Moran, Brain-controlled interfaces: Movement restoration with neural prosthetics, *Neuron* **8**(1) (2006) 205–200.
43. A. B. Schwartz, Cortical neural prosthetics, *Annu. Rev. Neurosci.* **27** (2004) 487–507.
44. E. Musk, An integrated brain-machine interface platform with thousands of channels (2019), doi:10.1101/703801v4.
45. B. S. Wilson and M. F. Dorman, Cochlear implants: A remarkable past and a brilliant future, *Hear. Res.* **242**(1–2) (2008) 3–21.
46. P. C. Laizou, Signal-processing techniques for cochlear implants, *IEEE Eng. Med. Biol.* **18**(3) (1999) 34–46.
47. A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto and H. Adam, MobileNets: Efficient convolutional neural networks for mobile vision applications, arXiv:1704.04861.
48. Nvidia's Jetson AGX Xavier Module, https://developer.nvidia.com/embedded/jetson-agx-xavier.
49. Y. Cheng, D. Wang, P. Zhou and T. Zhang, A survey of model compression and acceleration for deep neural networks, arXiv:1710.09282.
50. R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár and K. He, *Detectron* (2018), https://github.com/facebookresearch/detectron.
51. M. Bojarski, D. D. Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao and K. Zieba, End to end learning for self-driving cars, arXiv:1604.07316.
52. J. Wang, Y. Ma, L. Zhang, R. Gao and D. Wu, Deep learning for smart manufacturing: Methods and applications, *J. Manuf. Syst.* **48**(1) (2018) 144–156.
53. A. Kamilaris and F. X. Prenafeta-Boldú, Deep learning in agriculture: A survey, *Comput. Electron. Agric.* **147** (2018) 70–90.
54. P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya, M. P. Lungren and A. Y. Ng, CheXNet: Radiologist-level pneumonia detection on chest X-rays with deep lLearning, arXiv:1711.05225.
55. D. A. Ragab, M. Sharkas, S. Marshall and J. Ren, Breast cancer detection using deep convolutional neural networks and support vector machines, *PeerJ* **7** e6201, doi:10.7717/peerj.6201.
56. C. Pandarinath, D. J. O'Shea, J. Collins, R. Jozefowicz, S. D. Stavisky, J. C. Kao, E. M. Trautmann, M. T. Kaufman, S. I. Ryu, L. R. Hochberg, J. M. Henderson, K. V. Shenoy, L. F. Abbott and David Sussillo, Inferring single-trial neural population dynamics using sequential auto-encoders (2017), doi:10.1101/152884.
57. L. T. McIntosh, N. Maheswaranathan, A. Nayebi, S. Ganguli and S. A. Baccus, Deep learning models of the retinal response to natural scenes, arXiv:1702.01825.
58. S. A. Cadena, G. H. Denfield, E. Y. Walker, L. A. Gatys, A. S. Tolias, M. Bethge and A. S. Ecker, Deep convolutional models improve predictions of macaque V1 responses to natural images, *PLOS Comput. Biol.* **15**(4) (2019) 1–27.
59. W. F. Kindel, E. D. Christensen and J. Zylberberg, Using deep learning to reveal the neural code for images in primary visual cortex, arXiv:1706.06208.
60. C. F. Cadieu, H. Hong, D. L. K. Yamins, N. Pinto, D. Ardila, E. A. Solomon, N. J. Majaj and J. J. DiCarlo, Deep neural networks rival the representation of primate IT cortex for core visual object recognition, *PLoS Comput. Biol.* **10**(4) (2014) 1–18.
61. A. H. Marblestone, G. Wayne and K. P. Kording, Toward an integration of deep learning and neuroscience, *Front. Comput. Neurosci.* **10**(94) (2016).
62. M. H. Histed, V. Bonin and R. C. Reid, Direct activation of sparse, distributed populations of cortical neurons by electrical microstimulation, *Neuron* **63**(4) (2009) 508–522.
63. A. W. Roe1, M. M. Chernov, R. M. Friedman and G. Chen, *In vivo* mapping of cortical columnar networks in the monkey with focal electrical and optical stimulation, *Front. Neuroanat.* **9** (2015) 135.
64. CocoDataSet. http://cocodataset.org/.
65. PascalVOCproject. http://host.robots.ox.ac.uk/pascal/VOC/.

66. J. Redmon and A. Farhadi, YOLOv3: An incremental improvement, arXiv:1804.02767.
67. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu and A. C. Berg, SSD: Single shot multibox detector, arXiv:1512.02325.
68. S. Xie and Z. Tu, Holistically-nested edge detection, arXiv:1504.06375.
69. K. He, G. Gkioxari, P. Dollár and R. Girshick, Mask R-CNN, arXiv:1703.06870.
70. S. Chen, Y. Liu, X. Gao and Z. Han, MobileFaceNets: Efficient CNNs for accurate real-time face verification on mobile devices, in *Biometric Recognition* (Springer, Cham, 2018), pp. 428–438.
71. B. Amos, B. Ludwiczuk and M. Satyanarayanan, *Openface: A general-purpose face recognition library with mobile applications* (2016), http://cmusatyalab.github.io/openface/.
72. F. Schroff, D. Kalenichenko and J. Philbin, FaceNet: A unified embedding for face recognition and clustering, arXiv:1503.03832.
73. L. Liebel and M. Körner, MultiDepth: Single-image depth estimation via multi-task regression and classification, arXiv:1907.11111.
74. S. Gur and L. Wolf, Single image depth estimation trained via depth from defocus cues, *Proc. IEEE Computer Society Conf. Computer Vision Pattern Recognition*, California, USA, 2019, pp. 7683–7692.
75. W. Wang, Q. Lai, H. Fu, J. Shen and H. Ling, Salient object detection in the deep learning era: An in-depth survey, arXiv:1904.09146.
76. A. Lozano, J. S. Suarez, C. Soto-Sánchez, J. Garrigós, J. J. Martínez, J. M. Ferrández and E. Fernández, Neurolight alpha: Interfacing computational neural models for stimulus modulation in cortical visual neuroprostheses, *Lect. Notes Comput. Sci.* **11486** (2019) 1–12.
77. J. M. Ferrandez Vicente, J. R. Alvarez-Sanchez, F. de la Paz Lopez, J. Toledo Moreo and H. Adeli (eds.), Understanding the brain function and emotions, *Proc. of the 8th Int. Work-Conf. the Interplay Between Natural and Artificial Computation*, Part I (Springer, 2019).
78. J. M. Ferrandez Vicente, J. R. Alvarez-Sanchez, F. de la Paz Lopez, J. Toledo Moreo and H. Adeli (eds.), *From Bioinspired Systems and Biomedical Applications to Machine Learning, Proc. of the 8th Int. Work-Conf. the Interplay Between Natural and Artificial Computation*, Part II (Springer, 2019).
79. B. Naecker and N. Maheswaranathan, Pyret: Retinal data analysis in Python — pyret 0.6.0 documentation, https://pyret.readthedocs.io/en/master/.
80. J. Deng, W. Dong, R. Socher, L. Li, K. Li and L. Fei-fei, Imagenet: A large-scale hierarchical image database, *Proc. of IEEE Conf. Computer Vision and Pattern Recognition,* Florida, United States, 2009, pp. 248–255.
81. S. A. Baccus and M. Meister, Fast and slow contrast adaptation in retinal circuitry, *Neuron* **36**(5) (2002) 909–919.
82. D. P. Kingma and J. Ba, ADAM: A method for stochastic optimization, arXiv:1412.6980.
83. W. H. Dobelle and M. G. Mladejovsky, Phosphenes produced by electrical stimulation of human occipital cortex, and their application to the development of a prosthesis for the blind. *J. Physiol.* **243**(2) (1974) 553–576.
84. E. M. Schmidt, M. J. Bak, F. T. Hambrecht, C. V. Kufta, D. K. O'Rourke and P. Vallabhanath, Feasibility of a visual prosthesis for the blind based on intracortical micro stimulation of the visual cortex, *Brain* **119**(2) (1996) 507–522.
85. A. N. Foroushani, C. C. Pack and M. Sawan, Cortical visual prostheses: From microstimulation to functional percept, *J. Neural Eng.* **15**(2) (2018) 21005.
86. R. A. Normanna, D. J. Warren, J. Ammermuller, E. Fernandez and S. Guillory, High-resolution spatiotemporal mapping of visual pathways using multielectrode arrays, *Vision Res.* **41**(10) (2001) 1261–1275.
87. H. C. Stronks and N. H. Lovell, Phosphene mapping techniques for visual prostheses, in *Visual Prosthetics* (Springer, Boston, 2011), pp. 367–383.
88. S. Romero, C. Morillas, F. Pelayo and E. Fernandez, Computer-controlled neurostimulation for a visual implant, *Proc. of the First Int. Conf. Biomedical Electronics and Devices*, Portugal, Madeira, 2008, pp. 84–91.
89. T. Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick and P. Dollár, Microsoft COCO: Common objects in context, arXiv:1405.0312.
90. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu and A. C. Berg, SSD: Single shot multiBox detector, arXiv:1512.02325.
91. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L. C. Chen, MobileNetV2: Inverted residuals and linear bottlenecks, arXiv:1801.04381.
92. C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang and C. Liu, A survey on deep transfer learning, arXiv:1808.01974.
93. E. Fernandez, C. Soto, A. Alfaro, P. Gonzalez, A. Lozano, S. Peña, M. D. Grima, A. Rodil, A. Alarcon, J. Rolston, T. Davies and R. A. Normann, Development of a cortical visual neuroprosthesis for the blind: Preliminary results, *Invest. Ophthalmol. Vis. Sci.* **60**(9) (2019) 4021.
94. Nvidia's Jetson Nano Module, https://www.nvidia.com/en-gb/autonomous-machines/embedded-systems/jetson-nano/.
95. Jetson Nano Deep Learning Inference Benchmarking, https://devtalk.nvidia.com/default/topic/1050

377/jetson-nano/deep-learning-inference-benchmarking-instructions/.
96. P. R. Roelfsema, D. Denys and P. C. Klink, Mind reading and writing: The future of neurotechnology, *Trends. Cogn. Sci.* **22**(7) (2018) 598–610.
97. M. Sanchez-García, R. Martínez-Cantin and J. J. Guerrero, Semantic and structural image segmentation for prosthetic vision, *PLoS One* **15**(1) (2020) 1–22.
98. D. R. Geruschat, M. Flax, N. Tanna, M. Bianchi, A. Fisher, M. Goldschmidt, L. Fisher, G. Dagnelie, J. Deremeik, A. Smith, F. Anaflous and J. Dorn, FLORA^{TM}: Phase I development of a functional vision assessment for prosthetic vision users, *Clin. Exp. Optom.* **98**(4) (2015) 342–347.
99. O. Younis, W. Al-Nuaimy, F. Rowe and M. H. Alomari, A smart context-aware hazard attention system to help people with peripheral vision loss, *Sensors* **19**(17) (2019) 1–22.
100. H. Apfelbaum, E. Peli and A. Pelah, Heading assessment by 'tunnel vision' patients and control subjects standing or walking in a virtual reality environment, *ACM Trans. Appl. Percept.* **4**(1) (2007) 8, doi:10.1145/1227134.1227142.
101. C. J. M. Bollen, U. Guclu, R. J. A. Van-Wezel, M. A. J. Van-Gerven and Y. Gucluturk, Simulating neuroprosthetic vision for emotion recognition, *Proc. IEEE 8th Int. Conf. Affective Computing and Intelligent Interaction Workshops and Demos*, Cambridge, United Kingdom, 2019, pp. 85–87.
102. F. Solari, M. Chessa and S. P. Sabatini, Design strategies for direct multi-scale and multi-orientation feature extraction in the log-polar domain, *Pattern Recognit. Lett.* **3**(1) (2012) 41–51.
103. M. Chessa, G. Maiello, P. J. Bex and F. Solari, A space-variant model for motion interpretation across the visual field, *J. Vis.* **16**(2) (2016) 1–24.
104. G. Indiveri, B. Linares-Barranco, T. J. Hamilton, A. van-Schaik, R. Etienne-Cummings, T. Delbruck, S. Liu, P. Dudek, P. Häfliger, S. Renaud, J. Schemmel, G. Cauwenberghs, G. Arthur, K. Hynna, F. Folowosele, S. Saighi, T. Serrano-Gotarredona, J. Wijekoon, Y. Wang and K. Boahen, Neuromorphic silicon neuron circuits, *Front. Neurosci.* **5** (2011) 1–23.
105. A. Gibaldi, A. Canessa, M. Chessa, S. P. Sabatini and F. Solari, A neuromorphic control module for real-time vergence eye movements on the iCub robot head, *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, Bled, Slovenia, 2011, pp. 543–550.
106. E. Chicca, F. Stefanini, C. Bartolozzi and G. Indiveri, Neuromorphic electronic circuits for building autonomous cognitive systems, *Proc. IEEE* **102**(9) (2014) 1367–1388.
107. R. P. Rao, Towards neural co-processors for the brain: Combining decoding and encoding in brain-computer interfaces, *Curr. Opin. Neurobiol.* **55** (2019) 142–151.
108. B. V. Vugt, B. Dagnino, D. Vartak, H. Safaai, S. Panzeri, S. Dehaene and P. R. Roelfsema, The threshold for conscious report: Signal loss and response bias in visual and frontal cortex, *Science* **360**(6388) (2018) 537–542.