Computers in Biology and Medicine 113 (2019) 103399

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

# Biologically-inspired image processing in computational retina models☆

Nikos Melanitis*,[1], Konstantina S. Nikita[1]

*Biomedical Simulations and Imaging Laboratory, School of Electrical and Computer Engineering, National Technical University of Athens, Athens, Greece*

ARTICLE INFO

*Keywords:*
RGC functions
Image processing
Feature extraction
Retina model
Retinal prosthesis

ABSTRACT

Retinal Prosthesis (RP) is an approach to restore vision, using an implanted device to electrically stimulate the retina. A fundamental problem in RP is to translate the visual scene to retina neural spike patterns, mimicking the computations normally done by retina neural circuits. Towards the perspective of improved RP interventions, we propose a Computer Vision (CV) image preprocessing method based on Retinal Ganglion Cells functions and then use the method to reproduce retina output with a standard Generalized Integrate & Fire (GIF) neuron model. "Virtual Retina" simulation software is used to provide the stimulus-retina response data to train and test our model. We use a sequence of natural images as model input and show that models using the proposed CV image preprocessing outperform models using raw image intensity (interspike-interval distance 0.17 vs 0.27). This result is aligned with our hypothesis that raw image intensity is an improper image representation for Retinal Ganglion Cells response prediction.

## 1. Introduction

Retinal Prosthesis (RP) is an approach to restore vision in blind people affected by degenerative retina diseases, where, despite the damage to retina cells, at least some Retinal Ganglion Cells (RGCs) remain functional. RP could potentially benefit a great number of individuals with vision problems, as in the case of Retinitis Pigmentosa which has a prevalence of approximately 1/4000 [1]. Essential steps to transfer RP technology to standard medical care have been taken through clinical trials [2]. Implantees with Retinitis Pigmentosa have been able to detect luminous sources and direction of motion while experiencing an overall improvement in their orientation and mobility [3].

RP devices consist of [4]: (i) a camera, to capture images of the scene, (ii) a processing unit, to process the camera images and compute the proper retina stimulation pattern, (iii) a telemetry system to transfer information and power between the external device and the implant, and (iv) an implanted electrode array to stimulate the retina. ARGUS II and Alpha IMS RP devices have received approval for medical use [4,5]. Photovoltaic RPs that do not require a camera to capture the scene are studied in ongoing clinical trials [6]. Currently, implants process the images in a simple intensity-based manner, translating proportionally image intensity to stimulation intensity [7]. Progress in vision restoration by RP systems depends on accurate retina's input-to-output mapping.

Retina performs complex processing of visual information, as brightness computation, motion and edge detection [8,9], that has been explored in neurobiological studies [8–14]. Around twenty RGC types transmit visual information to the brain [10,13] and typically, each RGC type is associated with a distinct transformation of the visual scene. However, recent studies contrast this assumption and present examples of specific RGCs with different response across different stimulation conditions [15]. Although nonlinear RGC responses have been described early in the study of retina function [12,16], they initially received less attention, as linear RGCs [17] were initially assumed to have the central role in vision [13].

Many functional models of the retina are comprised of linear filters followed by a static non-linearity (linear-nonlinear or LN models) and a spike generation mechanism that can be either probabilistic or deterministic [18–20]. In models that mimic information processing in the retina, model parameters may be deduced from retina physical parameters [21]. A general approach that simplifies model development is to rely on data-fitting methods to estimate model parameters [18,20,22]. Initial fitting methods constrained training stimuli to be drawn from a spherically symmetrical distribution [23], while more recent methods [24] can be used with arbitrary stimuli, as natural image sequences [20,25–27]. Additionally, recent studies have revealed the inefficiency of linear RGC models developed with non naturalistic

☆ N. Melanitis is funded by General Secretariat for Research and Technology (GSRT) and Hellenic Foundation for Research and Innovation(HFRI).
* Corresponding author.
*E-mail addresses:* nmelan@biosim.ntua.gr (N. Melanitis), knikita@ece.ntua.gr (K.S. Nikita).
[1] School of Electrical and Computer Engineering, National Technical University of Athens, Athens, Greece.

https://doi.org/10.1016/j.compbiomed.2019.103399
Received 10 May 2019; Received in revised form 20 August 2019; Accepted 20 August 2019

stimuli to reliably reproduce response to natural stimuli [15,26,28].

Different approaches have been used to test and evaluate models of visual processing by the retina. The most direct approach is to reproduce retina response [18,29]. Artificial data, i. e data generated by a model, serve the initial validation of new research approaches [18,29,30]. The improvement of artificial datasets has further motivated the development of accurate retina models [31]. An alternative validation option is to reproduce retina properties, as contour discrimination, level of cellular activity and sensitivity to light changes [32]. Additionally, evaluation has been based on establishing desired method properties, such as showing real-time feasibility of image processing methods without, however, testing the modeling assumptions [33].

Image processing in RP has been used to enhance image features and improve the achieved visual acuity [34,35]. Computer Vision (CV) methods are employed to interpret the scene and then provide cues, of visual or other form, to the user. Examples include strengthening the image edges, detecting important regions [34], contrast and brightness enhancement, face and object recognition [35]. However, incorporating image processing methods in models used to generate the retina spiking pattern, has not been adequately explored. A recent approach proposed to preprocess the images using the Direct Cosine Transform, which efficiently encodes and compresses the visual information [36].

In this paper, we introduce RGC models that integrate the current understanding of RGC functions in a preprocessing feature extraction step. We address issues of existing computational model concerning: readily model refinement and validation against experimental data of any RGC type, incorporation of nonlinear RGC functions, straightforward model enhancement and upgrade through fusion with more features or advanced spike generation mechanisms, and the computational cost that can be even further reduced through hardware implementation of CV algorithms used [37,38]. We examine RGC response characteristics by cell type to extract image features that correspond roughly to RGC functions. These novel features are subsequently fed to spiking neuron models trained with a maximum likelihood algorithm. Thus, the proposed approach can be flexibly applied to model different types of biological RGCs. The ability of our model to reproduce retina response to naturally occurring visual stimuli is enhanced by using natural images in training. We examine properties shared between response of nonlinear RGCs and the extracted features. We assess the performance of our image preprocessing pipeline by comparing the obtained results with those of RGC models fed with unprocessed (raw) images. To examine performance variation across RGC types, we model linear and nonlinear RGCs. The effect of image content and resolution on performance is examined through appropriately designed studies.

## 2. RGCs function and feature extraction

### 2.1. Retinal organization

RGCs spiking response encodes the visual information that is transmitted through the optic nerve to the brain. RGCs form parallel information streams, each encoding a distinct representation of the visual scene [8–11,13,14]. Each information stream is encoded by a different RGC type, with distinctive morphology, response properties and central projections [10]. Studies on animals (mouse, rabbit, cat, monkey) have shown that on average around twenty different RGC types can be identified on mammalian retinae [13]. When we examine the Receptive Fields (RFs) of any particular RGC type collectively, we observe that the RFs form a mosaic that covers the entire visual field, with little overlap between individual RFs. Local retina circuits transmit the signal generated by photoreceptors through Bipolar Cells, Horizontal cells and Amacrine cells to RGCs. Each RGC type has been associated with a unique retina circuit [9].

### 2.2. RGC types, functions and corresponding image features

The linear RGCs display a center-surround antagonism and thus "ON" type cells are excited while "OFF" inhibited when illumination at the center is increased relative to the surrounding [17]. "ON-OFF" cells are excited by "ON" and "OFF" type stimuli [39]. Cat X cells are representative linear RGCs [9].

Local edge detectors respond to edges that fall within the center of their RF. They respond to changes in stimulation and not to static stimuli [14]. These cells detect a contrasting border confined to a small region of the visual field [40].

Motion sensitive cells are excited when specific motion patterns in the visual scene fall within their RF. Different types of motion sensitive cells have been identified in the retina which may display center-surround antagonism or selectivity in motion direction [11,40].

The simplest motion sensitive cells are general motion detectors and respond to all moving textures regardless of other parameters [11]. Representative cells in this class are cat Y cells [13], mouse a-RGCs and monkey Parasol cells [14].

Object motion sensitive (OMS) cells respond to differential motion. They have a center-surround structure and are excited by center motion but suppressed when a motion occurs simultaneously in both center and surround regions. These cells do not have any preference regarding the motion direction. OMS cells can detect moving objects, but will not fire spikes when the visual scene drifts on the retina because of ocular movements [11]. Mouse W3 cell is a representative OMS cell [13].

Direction selective (DS) cells respond selectively to motion along their preferred axis (up, down, right, left) and exhibit center-surround antagonism, similarly to OMS cells [13].

Uniformity detectors (UD), also known as suppressed-by-contrast cells, maintain a constant firing rate which is suppressed by all kinds of non-uniform stimulus [40].

In order to model various RGC functions [8,9,11,13,14] we apply CV feature extraction methods in each image region $I_R$ with size matched to the cell's RF.

#### 2.2.1. Linear RGCs

The RF weighting function of linear RGCs has been well approximated by the Difference of two concentric Gaussians (DoG) [17]. To extract a DoG feature, we convolve ($*$) the image $I$ with two concentric Gaussian filters, a center $G_c$ and a surround $G_s$ with spatial scales (standard deviations $\sigma_c$ and $\sigma_s$, respectively) corresponding to the RF size of the cell:

$$\mathrm{DoG} = (G_c - G_s) * I = G_c * I - G_s * I \tag{1}$$

We sample the value of the DoG filtered image at the cell's center position. We normalize DoG response in the range [0,1] by noting that DoG is positive in a region inside a properly defined $r_c$, $DoG(r < r_c) > 0$, and is negative outside $r_c$, $DoG(r > r_c) < 0$. We can find $r_c$ by conditioning $DoG(r) = 0$:

$$r_c = \sqrt{\frac{4\sigma_c^2\sigma_s^2 \ln\frac{\sigma_s}{\sigma_c}}{\sigma_s^2 - \sigma_c^2}} \tag{2}$$

Since image intensity is in [0,1], we get the maximum DoG response $M$, as the integral of the DoG filter in the region $r < r_c$,

$$M = \exp\left(\frac{-r_c^2}{2\sigma_s^2}\right) - \exp\left(\frac{-r_c^2}{2\sigma_c^2}\right) \tag{3}$$

Finally, the integral of a DoG filter in $\mathbb{R}^2$ is 0 and so the minimum DoG response is $m = -M$.

We also extract binary (true or false) features which are conceived as rough estimations of whether a RGC will be activated or not, as follows:

- $DoG > T$ :ON

- $DoG < 1 - T$:OFF
- We use the logical OR (|) of the two previous binary values: $(ON|OFF)$

The threshold $T$ is set at $T = 0.51$ so that the condition for either ON or OFF case is easily satisfied by any input pattern that matches the RF weighting function.

#### 2.2.2. Nonlinear RGCs: local edge detectors

We extract image edges using the Canny edge detector response [41]:

$$\nabla^2(G * I) = 0 \tag{4}$$

Similarly to the linear RGCs, we introduce, apart from the Canny response, a binary feature stream that is set to true whenever an edge falls within the image area that defines the cell's RF ($I_R$).

#### 2.2.3. Nonlinear RGCs: motion sensitive cells

In order to add motion sensitivity to the feature based representation, we use the optical flow (OF) to produce a dense velocity field from an image sequence [42]. We describe the OF field using the horizontal ($v$) and vertical ($u$) components of the flow vectors. The OF implementation relies on the brightness constancy constraint (BCC), which states that a moving point has constant brightness across two frames of the image sequence,

$$B(p) = I_t + vI_x + uI_y = 0 \tag{5}$$

where $p$ denotes a point and the subscripts $t$, $x$, $y$ denote temporal and spatial derivatives at $p$. Eq. (5) is solved with respect to the unknown $v$, $u$ flow components following the standard Lukas-Kanade (LK) method [43]. To avoid numerical instability issues, Tikhonov regularization is used [44].

Research in biological vision has attempted to explain motion detection in sensory systems comprised by photoreceptors through neural circuits with time-delayed signals that are compared by a special neuron (Reichardt detectors [45]). Biological grounding for such circuits have been found in starburst amacrine cells-bipolar cells interactions [46]. We observe that BCC and Reichardt detector are based on the same principle- that motion is a phenomenon in which an object at one location appears elsewhere after a time delay. We use LK to computationally represent motion since it is a robust and widely used method that serves the purpose of a simple and general feature to incorporate motion sensitivity.

We get the mean flow vectors in $I_R$ ($\overrightarrow{f_m}$), in center ($\vec{C}$) and surround ($\vec{S}$) subregions, to extract additional binary features, for general motion detectors, DS and OMS cell types. For DS cells specifically, we average flow's projection on direction $d$ ($C_d$, $S_d$), where $d$ can be either up, down, right or left. In each case, we perform cell type specific operations on the computed flow and then threshold the result to get a binary value:

- General Motion Detectors: $\|\overrightarrow{f_m}\| > T$
- OMS cells: $|\|\vec{C}\| - \|\vec{S}\|| > T$
- DS cells: $C_d - S_d > T$

By using low threshold ($T$) values ($T = 0.1$) we can detect any motion that occurs in accordance to the motion pattern detected by each cell type.

#### 2.2.4. Nonlinear RGCs: uniformity detectors

We measure uniformity in the spatial and temporal domains. We introduce variance [47] and entropy [48] features to measure spatial uniformity. We use image variance in $I_R$, $var(I_R)$ and we define the variance feature as $f_v = \frac{var(I_R)}{A}$, where $A = \left(\frac{r(I_R)}{2}\right)^2 = \left(\frac{1}{2}\right)^2$, with $r(I_R)$ being the range of intensity values of $I_R$, and $r(I_R) = 1$ assuming intensity values are expressed in the range [0,1]. We observe that $f_v$ takes values in [0,1]. We evaluate entropy using 256 intensity bins as

$$f_e = \sum_{i=1}^{256} - p_i \log_2(p_i) \tag{6}$$

The image region $I_R$ with maximum entropy is characterized by a uniform distribution ($p_i = \frac{1}{256}$, $i = 1...256$), over the intensity bins. Thus, $f_e$ takes values in the range [0,8].

Moreover, we quantify the temporal uniformity by the mean, absolute, pixel-wise, difference of $I_R$ in two consecutive frames, $I_{R,n}$, $I_{R,n+1}$,

$$f_t = \frac{1}{N} \sum_{i=1}^{N} |I_{R,n+1}(\vec{r_i}) - I_{R,n}(\vec{r_i})| \tag{7}$$

where $N$ denotes the number of pixels in $I_R$. Consequently, $f_t$ takes values in the range [0,1].

Again, we complement the extracted uniformity features with a binary valued feature by thresholding $f_v$, $f_e$, $f_t$. The thresholds $T_v$, $T_e$, $T_t$, are set on the 10% of the maximum feature values. We declare $I_R$ not uniform if any of the features surpasses its respective threshold, that is, if the following condition holds:

$$(f_v > T_v)|(f_e > T_e)|(f_t > T_t) \tag{8}$$

### 2.3. Proposed feature vector

The modeled RGC functions and the corresponding CV methods are summarised in Fig. 1. Each image region ($I_R$) that corresponds to a cell RF is represented by a 18-dimensional feature vector that contains:

- the DoG response, sampled at the cell position (one value)
- the ON, OFF and ON/OFF binary streams (three values)
- Canny edge response averaged in $I_R$ region (one value)
- edge detector binary stream (one value)
- flow components in $x-$ and $y-$ image axes averaged in $I_R$ region (two values)
- binary streams for general motion detection, DS (four values) and OMS
- variance, entropy and temporal uniformity of region $I_R$ (three values)
- uniformity detector binary stream (one value)

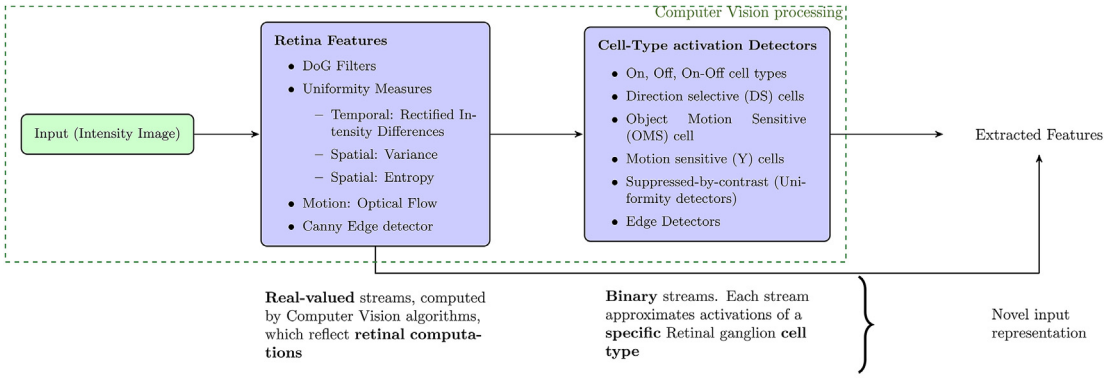
## 3. Using RGC feature extraction in retina simulations

### 3.1. Reference model and dataset generation

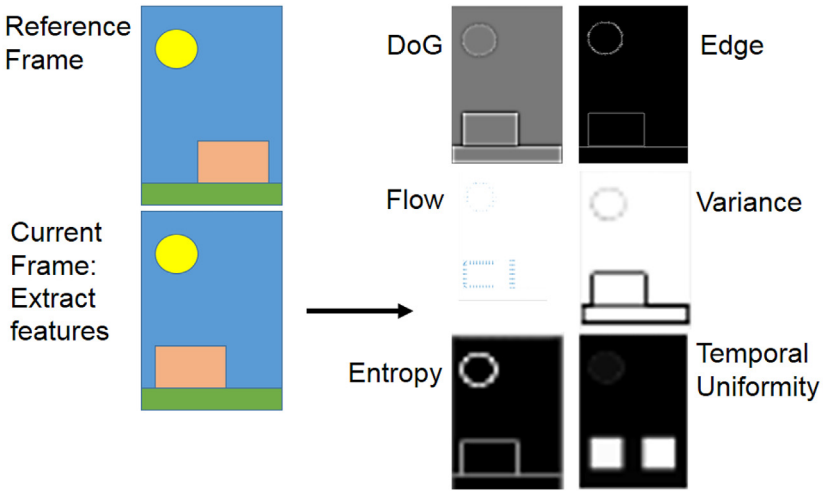
To train and evaluate our approach, we use an artificial dataset of retina spiking responses to natural image stimuli generated using Virtual Retina (VR) simulator software (reference model) [21].

VR is a layered model in three stages ((i)outer plexiform layer, (ii) contrast gain control in bipolar cells (BCs), (iii)inner plexiform layer and RGCs) that follows retina's organization in cellular layers. At several locations of the model, linear spatio-temporal shaping is implemented through temporal exponentials or exponential cascades and spatial two-dimensional Gaussians. Additionally, a first order differential equation with time-dependent coefficients (i.e. “time-constant”) is implemented in BC layer. VR is validated by reproducing experimental recordings (responses to gratings, multi-sinus analysis) of biological RGCs (cat X and cat Y cells). Model parameters' values are either taken from the literature or are manually set to produce good results on all the examined experiments. Cat X, cat Y, parvo and magno cell parameters are provided [21].

In Fig. 2, we present three frames of the used input sequence, which shows two persons walking down a road, provided by VR software release. The sequence is comprised of 56 frames of size 160 × 128 pixels.




(a)




(b)

**Fig. 1.** Top: Pipeline to transform an Image to more relevant information streams. The graph shows current implementation. Features (stage 1) and Detectors' outputs (stage 2) are fed to models for RGC spiking pattern computation Bottom: Schematic representation of feature extraction. The input is transformed to feature maps which are then filtered, sampled at RGC positions or manipulated through boolean operators (see Section 2) to produce the final representation by 18 scalar features.

In order to improve model training and testing, we increase the dataset size by tiling each frame with $n_c$ RGCs with non-overlapping RFs, thus increasing $n_c$-times the sequence length. The reference data takes the form of $n_c$ spike streams, each one associated with an $I_R$ that corresponds to the RF of a VR-simulated RGC. We use VR to simulate two different RGC types (Section 4.2). In all cases we use 70% of the input sequence as training set and 30% as testing set. Some features (e.g flow) cannot be extracted on the first frame of a sequence, which is discarded.

RF size in VR is determined by the spatial extent of the Gaussian filters used. To establish the spatial correspondence between reference (VR) cells and our models, we parse VR parameters and define RF size as a square with edges of $6\sigma_{VR}$ ($\pm$ $3\sigma_{VR}$ around the center), where $\sigma_{VR}$ is the maximum standard deviation used in VR Gaussian filters. Details on RF size and dataset size are provided in Section 4.

The reference model (VR) used is stochastic and we quantify its intrinsic variability (IV) with the following procedure: for any given input image sequence $V$, we run VR with Input $V$ ten times and get the spiking trains (output) $S_1$, ...,$S_{10}$. Then, we estimate Virtual Retina IV for sequence $V$ as the mean pairwise distance of spike trains

$$IV = \frac{1}{N}\sum_{i\neq j} d(S_i, S_j) \tag{9}$$

**Fig. 2.** The input sequence we use in our simulations. We take the sequence from Ref. [21]. The sequence has 56 frames. We tile each frame with $n_c$ non overlapping RGCs, thus producing a total of $56{\cdot}n_c$ input "points" ($n_c$ depends on RGC RF size).

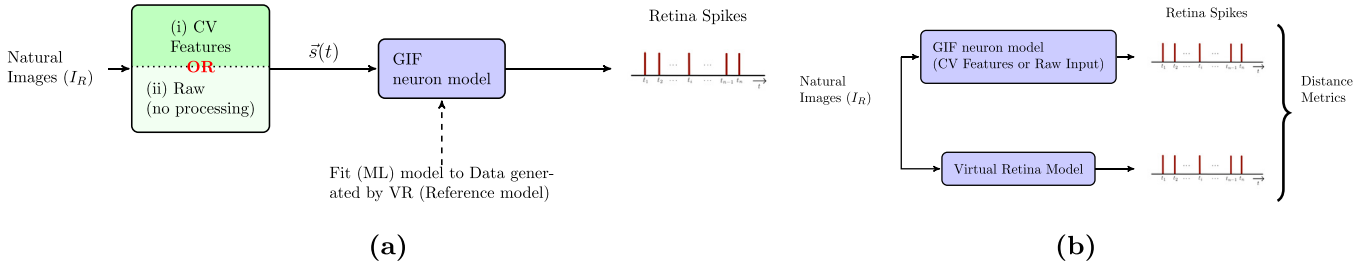



**Fig. 3.** Left: Training Pipeline. (i) The images are processed to extract features (Section 2). These features are used as input to a GIF model (Section 3.2) which produces the retina spiking output. To determine GIF model parameters, we use data fitting methods. (ii) Alternatively, raw images are directly fed to a GIF model Right: Performance evaluation is based on distance metrics between GIF model spiking response and VR spiking response.

### 3.2. Spiking neuron model: description and training

Retina response is obtained using a Generalized Integrate & Fire (GIF) spiking neuron model [49] fed with the extracted CV features. We train the model using a Maximum-Likelihood (ML) estimation algorithm [24] using artificial data generated as described in Section 3.1. The GIF neuron response is given by the following equations [49]:

$$\frac{dV}{dt} = -\frac{1}{\tau}(V(t) - V_l) + I_{stim}(t) + I_{sp}(t) + I_{nse}(t) \tag{10}$$

$$I_{stim}(t) = \vec{k} * \vec{s}(t) \tag{11}$$

$$I_{sp} = \vec{h} * \vec{r}(t) \tag{12}$$

where $\vec{k}$, $\vec{h}$ are convolutional filters, which are expressed in raised cosines bases, $\vec{s}(t)$ is the stimulus (input) vector, $\vec{r}(t)$ is a vector of recent spiking history matched to the size of $\vec{h}$, $I_{sp}$ is a feedback term and $I_{nse}$ a random input, which consists of Gaussian noise with standard deviation $\sigma_n$. Spikes occur whenever $V(t) > 1$ and then $V(t)$ is reset to zero. $V$ is an internal, real valued model parameter which corresponds to cell's membrane potential. By adjusting $\vec{k}$, $\vec{h}$ extent in time, we can model RGCs that display either transient or sustained temporal responses.

The parameters that are adjusted in training are the spatio-temporal filter weights ($\vec{k}$, $\vec{h}$ in eqs. (11) and (12) and neuron firing parameters: leakage voltage, integrator time-constant and noise standard deviation ($V_l$, $\tau$, $\sigma_n$ in eq. (10)). We evaluate the GIF model likelihood function, $P\left(\textbf{spikes}|\textbf{stim}, \left\{\vec{k}, \vec{h}, V_l, \tau, \sigma_n\right\}\right)$, that expresses the probability of observing a set of spike times (**spikes**) given a set of stimuli (**stim**) and a setting of parameters ($\vec{k}$, $\vec{h}$, $V_l$, $\tau$, $\sigma_n$). We fit model parameters by optimizing the likelihood of observed responses.

### 3.3. Simulation pipeline

Our pipeline is outlined in Fig. 3. In order to assess the performance we examine spike train similarity using two complementary measures [50,51], Interspike-Interval (ISI) and SPIKE distance [52,53]. We evaluate the distance of the two spiking responses generated by the introduced and reference model and compare to the reference model IV.

For comparison purposes, we train GIF spiking neuron models, using two different visual input representations (Fig. 3):

- Features: extracted from $I_R$ using the methods introduced in Section 2
- Raw intensity values

To compare the above "features" and "raw" approaches on equal grounds, we downsample the input images to achieve identical dimensionality of the two input representations, so that $\vec{s}(t)$ is a 18-dimensional vector that contains either feature values or intensities respectively.

We swept GIF parameters that are not adjusted in training, specifically the number of filters' ($\vec{k}$, $\vec{h}$) basis functions and filters' temporal extent, to create a population of models and then select the optimal-performing models. For each filter, we first determine its endpoints $e_1$, $e_2$. The raised cosine basis then has its first base function centered at $e_1$ and its last centered at $e_2$. We then sweep the number of base functions, $n_k$ and $n_h$ respectively, increasing them up to the maximum number of independent functions. For each dataset (see sets in Section 4.2-4.4) and cell type case (linear and nonlinear cells, see Section 4.2), sweeping $e_1$, $e_2$, $n_k$, $n_h$ values resulted in training approximately 90 models.

We need to choose endpoints so that the time dependence in the GIF model, which is expressed through filters $\vec{k}$, $\vec{h}$, corresponds to VR time dependence, which can be approximately estimated from the time constants of the successive temporal convolutions implemented in VR. Since, temporal filters that peak at $\tau > 0$, are also implemented in VR models, we additionally tested using $e_1 > 0$. We also tested using $e_1$ values in the feedback filter $\vec{h}$ that correspond to the VR refractory period. Raised cosines basis has the ability to represent fine temporal structure near $e_1$ but is smooth near $e_2$ [49].

We automate model selection by enforcing the following conditions:

- We reject models with performance worse than $T_{tr}$ in training set and $T_{te}$ in testing set. This step is intended to quickly discard poorly performing models
- For each trained model, we compare to an untrained model with the exact same filter size and base dimensionality but with randomly selected filter weights and other model parameters. We reject all trained models that perform equally or worse than their untrained counterpart, either in training or testing set. Overfitting or inappropriate parameters ($e_1$, $e_2$, $n_k$, $n_h$) may diverge training and lead to worse than random testing set performance. By applying this condition we detect such occurrences and discard such models
- Finally, we examine whether the models' spiking output carries information about the input sequence. We reject models that sustain a constant firing rate, regardless of the stimulus presented. We also compare to a "mean-rate" model that fires spikes at a constant rate, equal to the mean firing rate of the reference spiking train. We reject models that perform equally or worse than a mean-rate model

## 4. Results

### 4.1. Uniformity features exhibit properties of uniformity RGCs response

To establish the correspondence between our feature extraction procedure and biological retina processing, we examine properties described in an electrophysiological study of cat RGC responses to sinusoidal gratings [54]. We focus on CV uniformity features which have not been used in previous studies. We show that uniformity features display the three following RGC properties: the constant supression of

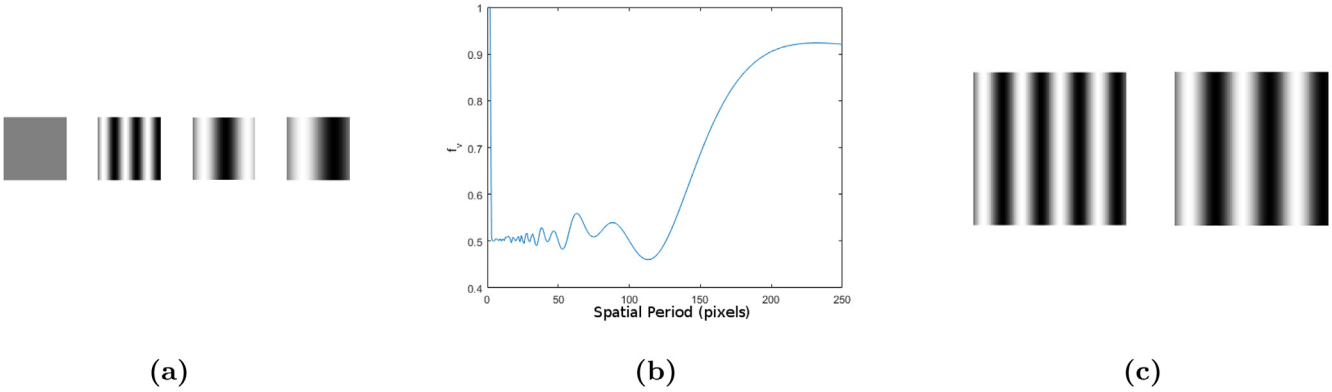



**Fig. 4.** Left: Sinusoidal gratings used as stimulus. The spatial period $T_s$ increases from $T_s = 1$ pixel (leftmost) to $T_s = 100$ pixels (rightmost). Center: Uniformity feature $f_v$ of static sinusoidal gratings with respect to gratings' spatial period. Right: The leftmost grating fits exactly in the RF area. Four full periods cover the RF area. For the rightmost spatial grating, approximately 3.5 periods cover the RF area.

cell's mean firing rate over a limited band of spatial frequencies, the monotonic dependence of suppression on contrast, where stronger contrasts induce stronger supression, and the sinusoidal modulation of response to gratings whose contrast reverses sinusoidally [54]. The stimuli used were sinusoidal spatial gratings. RF size was set at $100 \times 100$ pixels.

The relationship between gratings' spatial frequency and response is shown in Figs. 4b and 5. We observe constant uniformity feature values ($f_v$), in the spatial frequency band that can be sensed in our experimental setting, in agreement to the reduced firing rate of uniformity RGCs over a spatial frequency band [54]. The two spatial frequency cutoffs naturally arise, as when frequency increases, aliasing occurs by sampling the pattern at pixel positions, while when the frequency decreases, the pattern displays no variation within the extent of a single RF.

We present $f_v$ values of both static (Fig. 4b) and drifting (Fig. 5) sinusoidal gratings. We observe that $f_v$ values show small variation within the band the cell can sense (central region), but increase monotonically as $T_s$ surpasses RF's size. The small oscillations in Figs. 4b and 5 are caused by $f_v$ variations when RF contains a non-integer number of spatial pattern periods (Fig. 4c). In the case of static gratings, taking periods $T_s = 3...130$ pixels, so that the maximum $f_v$ value occurs in the central $T_s$ region, we get an average $f_v = 0.5153 \pm 0.0176$. Concerning drifting gratings, we examine $f_v$ for $T_s = 1...500$ pixels. We increased $T_s$ to 500 pixels to illustrate the upper plateau in $f_v$ values ($f_v \approx 0.9$) that occurs in larger $T_s$ values. We give the mean $f_v$ values at each $T_s$, which are relevant to the steady state mean firing rate in Ref. [54]. We similarly observe small variations in $f_v$, $f_v = 0.5063 \pm 0.0067$, as $T_s$ varies within the band the cell can sense, $T_s = 3...130$ pixels.

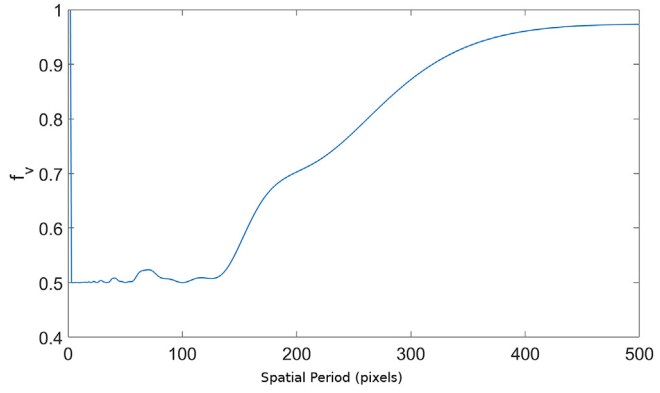



**Fig. 5.** Uniformity feature $f_v$ of drifting sinusoidal gratings as a function of spatial period. At each spatial period $T_s$, we drifted the grating over the cell's RF and averaged $f_v$ values.

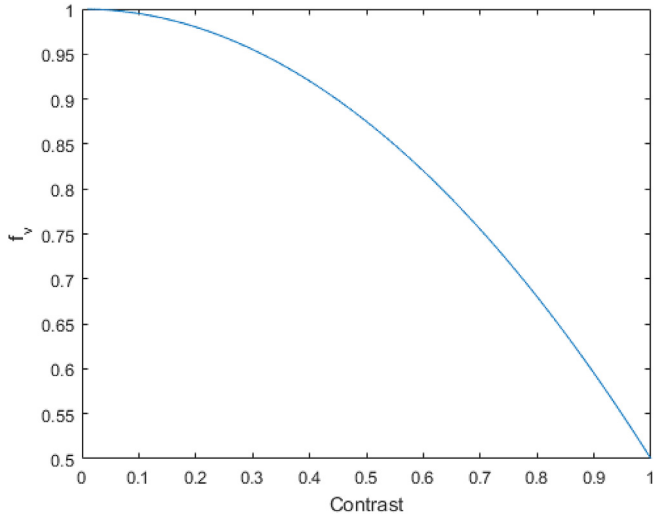



**Fig. 6.** Average $f_v$ values of drifting sinusoidal gratings with varying contrast. The sinusoidal grating has a spatial period $T_s = 10$ pixels.

We reproduce in Fig. 6 the monotonic decrease in mean FR as contrast increases [54]. We used a sinusoidal grating with period $T_s = 10$ pixels, however the result was independent of $T_s$ value. Again the pattern was shifted across the RF and $f_v$ averaged over a full pattern shift.

Concerning sinusoidal modulation of response, we turn to the temporal uniformity feature $f_t$, to examine the time profile of the response to sinusoidally contrast reversing static gratings and reproduce the sinusoidal response modulation observed in Ref. [54]. Previously, we focused on the mean response, while we now concentrate on the time evolution of the response.

Contrast is defined as $C = \frac{M-m}{M+m}$, where $M$, $m$ are the maximum and minimum intensities respectively. Thus, a sinusoidally contrast reversing grating is described by

$$S(x, t) = C(t) \cdot \sin x = \sin\left(t - \frac{\pi}{2}\right) \sin x \tag{13}$$

where $x$ denotes the spatial variable (Fig. 4a). Considering that $f_t$ is defined as the sum of the rectified differences over the RF area, we get

$$f_t = \sum_x |S(x, t) - S(x, t + \Delta T)| = A \cdot \left| \sin\left(t + \frac{\Delta T}{2}\right) \right| \tag{14}$$

where $A$ is a constant, the sum over $x$ covers the RF region and $t$, $t + \Delta T$ are the two time points compared in $f_t$ computation. Thus, $f_t$ time course assumes a rectified sine form, in agreement to the experimental

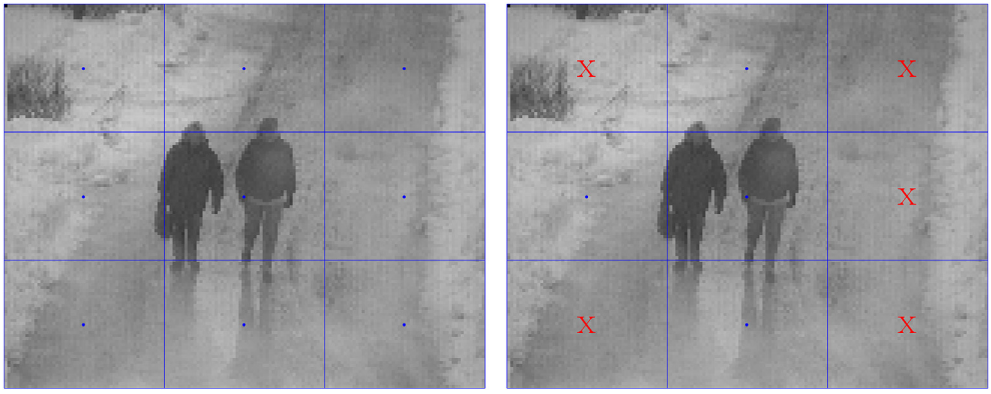



**Fig. 7.** Left: RGCs tile each frame. The RFs are noted as blue rectangles overlayed on the image frame. Each blue point, represents an RGC position. See Section 4.2. Right: An example of user selected RFs, in which we reject background-only RFs. Red "x" signs denote rejected RFs. See Section 4.4.

observations of [54].

### 4.2. Feature extraction evaluation on a natural image sequence

We examine the integration of CV processing pipeline to models of linear and nonlinear RGCs and compare to RGC models which use raw image intensity as their input. We use the input sequence of Fig. 2. In each frame, we create a grid of RGCs, so that neighboring RGCs have non-overlapping RFs that cover, like tiles, the total frame area. RGC tiling is illustrated in Fig. 7. We refer to the datasets of this section as "Full" datasets.

We model the two main cat RGC types, cat X and cat Y cells [17,21]:

- cat X (linear) cells whose response demonstrates linearity of summation within their RFs, as described in Section 2.2.1
- cat Y (nonlinear) cells, whose response exhibits motion sensitivity (Section 2.2.3), demonstrating nonlinear spatial summation, as indicated by experimentally identified response properties [21]. One such property is the strong excitation of cat Y cells whenever there is change in the stimulus, even if the stimulus contrast is reduced to low levels

Cat X and cat Y cells have been previously modelled in VR software [21].

In cat X cells "Full" set, RF is $18 \times 18$ pixels and $n_c = 35$ cells are instantiated in each frame which results in a total sequence length of 1925 RFs. In cat Y cells "Full" set, RF is $18 \times 18$ pixels and $n_c = 30$ which results in a total sequence length of 1650 RFs. We trained linear cell models for 35 training algorithm iterations and increased the iterations to 50 in nonlinear cell models, to capture their more complex response properties which are also indicated by the additional processing stages in VR cat Y cell model compared to cat X cell models.

Using the automatic model selection method (Section 3.3), we get the results shown in table 1. We set $T_{tr} = 0.30$, based on observed VR IV, and $T_{te} = 0.40$, as we anticipate worse testing set performance. We use these $T_{tr}$, $T_{te}$ values in all our simulations presented in this paper. In table 1, $n_s$ is the total number of models whose performance satisfied the conditions of the model selection method (see Section 3.3) and ISI, SPIKE values quantify the performance of the best performing (out of the $n_s$ selected) model in each cell type and dataset case. The reference model's IV is valuable in the interpretation of model performance. If GIF model spiking output has a distance to reference model's spike train lower than the model's IV, then the GIF model response is indistinguishable from reference model responses.

We present the top performing models trained for each cell type (cat X and cat Y) and input representation (CV features or raw image intensity). We additionally include in Table 1 the performance of 'Half' models, in which we reduce RF size (see Section 4.3), and of 'User' models, in which we modify the input sequence (see Section 4.4).

Regarding our results in Table 1, we note that the models that used CV features performed better than those that used raw intensity input. Furthermore, the improvement was more profound in the case of cat Y cells compared to cat X cells. This is an expected result, because the synaptic pooling of Y cells [21] increased nonlinearity of response and cannot be well approximated by fitting the GIF neuron's filter weights, when the input is the raw image intensity.

### 4.3. The effect of RF size on model performance

Next, we examine the effect of image resolution and RF size (in pixels) on the performance. We repeat the simulations of Section 4.2, but this time we reduce every cell's RF size by a scale factor of 0.5 ("Half" dataset). In cat X and cat Y "Half" sets, $n_c = 121$ cells are instantiated in each frame which results in a total sequence length of 6655 RFs. Again, we trained cat X cell models for 35 iterations and cat Y models for 50 iterations. In Table 1, we present the corresponding performance on the new "Half" dataset.

We compare the results on the datasets of this section ("Half" datasets), to those of Section 4.2 ("Full" datasets). In the case of cat X cells with "features" input, we have an improvement in "Half" set compared to "Full" set. This result can be attributed to the larger volume of data in "Half" set, which similarly leads to a slight performance improvement in both cat X and cat Y models with raw intensity input. In the case of cat Y cells with "features" input, we note that the "Half" set performance is slightly worse.

These results on cat X and cat Y performance show that cat Y cells are more dependent on CV features and that great reductions in image resolution compromise feature extraction. Image processing methods (e.g edge detection) did not improve the recognition of low resolution images (less than $25 \times 25$ pixels) in subjects with normal vision [55]. Moreover, edge detection methods in low resolution images perform worse than unprocessed images in picture recognition problems [56].

Below we summarize our findings on:

1. Linear and nonlinear cells: Linear (cat X) cells response depends more on image filters (DoG) and so is not as dependent on RF size as nonlinear (cat Y) cells, which depend on CV methods with documented resolution dependency [55,56].
2. Dataset Size: As each individual RF occupies a smaller area in "Half" sets, more non-overlapping RFs are used to cover the same image area. As a consequence, we instantiate a larger number of RGCs in different image coordinates. A larger dataset can contribute to improved model training
3. Raw input: The models that get raw input, are not affected by decreased image resolution, which is only relevant to feature extraction methods. However, the larger dataset (see also previous point

**Table 1**
Performance of the Best model trained for cat X (**linear**) or cat Y (**nonlinear**) cells and each input representation (Computer Vision features (Feature) or unprocessed image intensity (Raw)). In "Half" sets, RF is scaled by a factor of 0.5. In "User" sets, background RFs are rejected.

| Set | Cell | Input | ISI distance | | IV (ISI) | SPIKE distance | | IV (SPIKE) | Selected models ($n_s$) |
|---|---|---|---|---|---|---|---|---|---|
| | | | Training Set | Testing Set | | Training Set | Testing Set | | |
| Full | X | Feature | 0.17 | 0.24 | 0.05 | 0.09 | 0.11 | 0.07 | 12 |
| | | Raw | 0.17 | 0.30 | 0.05 | 0.09 | 0.13 | 0.07 | 10 |
| | Y | Feature | 0.12 | 0.17 | 0.04 | 0.08 | 0.09 | 0.08 | 13 |
| | | Raw | 0.22 | 0.27 | 0.04 | 0.11 | 0.12 | 0.08 | 12 |
| Half | X | Feature | 0.17 | 0.18 | 0.05 | 0.09 | 0.09 | 0.07 | 12 |
| | | Raw | 0.17 | 0.28 | 0.05 | 0.09 | 0.12 | 0.07 | 9 |
| | Y | Feature | 0.12 | 0.20 | 0.04 | 0.08 | 0.09 | 0.07 | 12 |
| | | Raw | 0.14 | 0.25 | 0.04 | 0.08 | 0.11 | 0.07 | 13 |
| User | X | Feature | 0.18 | 0.16 | 0.06 | 0.09 | 0.07 | 0.07 | 32 |
| | | Raw | 0.18 | 0.23 | 0.06 | 0.10 | 0.07 | 0.07 | 46 |
| | Y | Feature | 0.14 | 0.13 | 0.05 | 0.08 | 0.07 | 0.07 | 58 |
| | | Raw | 0.17 | 0.22 | 0.05 | 0.09 | 0.09 | 0.07 | 58 |

2), can provide performance gains for these models as well. Thus, we have a slight performance improvement in "Half" sets

### 4.4. Removing background-only instances from the dataset

In this section, we examine the relationship between model performance and training sequence content. We again use the dataset and RF size we used in Section 4.2, but reject RFs with static and background scenes and only keep RFs with actions, that is RFs with changing content. We call this new Dataset "User" (as it was interactively user-selected). The selection process which is illustrated in Fig. 7 results in a sequence length of 357 RFs in cat X "User" set and a length of 331 RFs in cat Y "User" set. We adapted to the smaller size of "User" datasets and trained all models for 35 iterations.

The obtained results are summarised in Table 1. For both cat X and cat Y cells, "User" datasets show significant performance improvements over "Full" sets, in terms of ISI distance and number of selected models.

## 5. Discussion

In this work, we show that CV algorithms can be integrated to retina models towards improving the models' performance. Instead of relying exclusively on data fitting methods to reliably reproduce retina response, we use studies on retina [8,9,11,13,14] to guide the selection of CV methods. We tested the effectiveness of CV processing in modeling the retina and demonstrated that performance degrades with low image resolution and improves when background input instances are rejected.

We verified the correspondence between uniformity features and properties of uniformity RGC response, for contrast dependence (Fig. 6), bandpass suppression in spatial frequency (Figs. 4b and 5) and response to sinusoidally contrast reversing gratings, and thus demonstrated the potential of our feature extraction pipeline to be incorporated in improved retina models that capture known retina response properties. We used both static gratings, to demonstrate the response variation to instantaneous inputs as spatial frequency varies, and drifting gratings, averaging feature values over a full drift across the RF, to capture the total input that is represented by the mean, steady-state, firing rate. To make distinct properties of feature extraction from properties of neuron models or other modules of retina models, we examined uniformity features directly and refrained from using the features in spike generation models. We highlight and analyze more thoroughly uniformity features in response to (i)the lack of literature correlating entropy, variance and intensity temporal variations to UD response and (ii)the availability of descriptions of UD response to various stimuli [54].

The introduced features reproduce the general principles of RGC functions and do not attempt to reproduce in detail the exact biological behaviour of a specific RGC. We do not attempt to use image processing towards detailed biophysical models of RGCs which relate membrane potentials to spikes, our aim is to build a computational model that translates images to RGC spikes. If the image representation using features mimics in great detail a specific biological function, it would be difficult to use this representation in models of different cells. Moreover, features are the input to our neuron model, not its final output, so features are not required to represent in great detail the biological functions.

The incorporation of our preprocessing pipeline leads to improved models for nonlinear and linear RGCs. The improvement is curbed in linear RGC models, as VR filters for linear RGCs can be approximated to an extent by GIF parameters. In VR models of nonlinear (Y) cells the internal variable that represents RGC input current is spatially Gaussian-averaged before the final spike generation layer, a process that is paralleled by the RF-wide averaging we apply after the various Image processing operators to extract the final features.

We introduce an automated model selection method which also takes into consideration biological plausibility and model generalization, by comparing trained neuron models to neurons firing at a constant rate or to randomly instantiated neuron models, respectively. The number of selected models ($n_s$) can be used as an additional measure of performance, as robust models are expected to sustain their performance in a neighborhood of GIF model parameters.

Following recent evidence that multiple computations are performed by RGCs of a single type, which challenge the "one function per RGC type" paradigm [15], we apply the same CV preprocessing pipeline in models of both linear and nonlinear RGCs. Since our approach is data-driven, it is straightforward to model different RGCs using appropriate datasets. In previous approaches (e.g VR [21]), the values of model parameters were determined manually requiring knowledge about both the RGC types studied and the methods used.

Our aim is to reproduce RGC image encoding to neural spikes. This diverges from efficient image coding which facilitates unique reconstruction of the encoded images. Retina output has been described both as performing efficient coding and as performing nonlinear lossy feature extraction [57]. Our model could be possibly improved towards faithful image reconstruction, by encoding the images to $m$ different spike streams, each associated with an appropriately selected RGC type, similarly to the way multiple RGC types tile the visual field.

Currently RP approaches stimulate RGCs in a non-selective manner and consequently our preprocessing pipeline may only be utilized towards more accurate models of a single RGC type. However, the non-selective RGC stimulation poses important limitations concerning the acuity of RP-attained vision and induces important perceptual

distortions [58]. Thus, selective RGC stimulation either by electrical pulses [59] or optogenetic stimulation strategies [60] is actively pursued and will, in the long term, make relevant the modeling of separate RGC populations in improved RP interventions. Additionally, improved retina models will allow for a deeper understanding of perceptual distortions in RP that are induced whenever retina is stimulated in an unnatural way [61].

While artificial data have been used to validate retina models [18,29] and neuron models [30], biological data can enhance the validity of the approach. In a study of prediction mechanisms on the retina, some results were only revealed with the use of a biological dataset [26]. Similarly, in a study investigating common retina population spiking patterns, or “codewords” [28], many more codewords (50) were found on the biological data, compared to only six “codewords” found on LN models fit to the same data.

In future steps, we may test our approach on different RGC cell types, and explore how well we can model each different cell type. In a recently published dataset [39], two-photon calcium imaging was used to record from more than 11000 individual RGCs, which are also assigned to specific cell types. The recent development and deployment of such advanced experimental techniques makes feasible the recording of RGC type specific cell responses to natural image stimuli.

Keeping the same feature extraction process, we may change the spike generation procedure using ad hoc, biologically-based modeling approaches or suitable, general approximation methods. The features themselves, or the retina spiking output, can also be used as inputs in different tasks (navigation [16,62], face detection [62]), as in biologically inspired computer vision [63,64] where biological solutions are mimicked in CV algorithms.

Finally, the introduced CV features could be combined with convolutional neural networks (CNNs) [65] to improve RGC response prediction. In the past, CNNs have been utilized in models of the primary visual cortex [66]. Image processing and deep learning have been combined in various biomedical [67–69] and CV applications applying feature fusion methods [70].

## 6. Conclusions

We have introduced a Computer Vision image preprocessing method to model RGC functions and reproduced retina spiking output with a GIF neuron model. We show that methods developed over the last decades in the Computer Vision field, can be transferred to the area of retinal implants to simulate retina computations. We have demonstrated that the use of features as input improves performance over raw image intensity, defending our hypothesis that raw image intensity is an improper visual input representation. Additionally, we have shown that low image resolution can degrade CV features performance and that model performance is improved when background-only inputs are rejected.

## Conflicts of interest

None Declared.

## References


[1] H. Permana, Q. Fang, I. Cosic, 3-layer implantable microstrip antenna optimised for retinal prosthesis system in mics band, Bioelectronics and Bioinformatics (ISBB), 2011 International Symposium on, IEEE, 2011, pp. 65–68.

[2] M.S. Humayun, J.D. Dorn, L. Da Cruz, G. Dagnelie, J.-A. Sahel, P.E. Stanga, A.V. Cideciyan, J.L. Duncan, D. Eliott, E. Filley, et al., Interim results from the international trial of second sight's visual prosthesis, Ophthalmology 119 (4) (2012) 779–788.

[3] K. Loizos, G. Lazzi, J.S. Lauritzen, J. Anderson, B.W. Jones, R. Marc, A multi-scale computational model for the study of retinal prosthetic stimulation, Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE, IEEE, 2014, pp. 6100–6103.

[4] J.D. Weiland, M.S. Humayun, Retinal prosthesis, IEEE Trans. Biomed. Eng. 61 (5) (2014) 1412.

[5] K. Stingl, K.U. Bartz-Schmidt, D. Besch, C.K. Chee, C.L. Cottriall, F. Gekeler, M. Groppe, T.L. Jackson, R.E. MacLaren, A. Koitschev, et al., Subretinal visual implant alpha ims–clinical trial interim report, Vis. Res. 111 (2015) 149–160.

[6] H. Lorach, G. Goetz, R. Smith, X. Lei, Y. Mandel, T. Kamins, K. Mathieson, P. Huie, J. Harris, A. Sher, et al., Photovoltaic restoration of sight with high visual acuity, Nat. Med. 21 (5) (2015) 476.

[7] S. Picaud, J.-A. Sahel, Retinal prostheses: clinical results and future challenges, Comptes Rendus Biol. 337 (3) (2014) 214–222 spotlight on vision https://doi.org/10.1016/j.crvi.2014.01.001 http://www.sciencedirect.com/science/article/pii/S163106911400002X.

[8] R.H. Masland, Cell populations of the retina: the proctor lecture, Investig. Ophthalmol. Vis. Sci. 52 (7) (2011) 4581.

[9] R.H. Masland, The neuronal organization of the retina, Neuron 76 (2) (2012) 266–280.

[10] G. Field, E. Chichilnisky, Information processing in the primate retina: circuitry and coding, Annu. Rev. Neurosci. 30 (2007) 1–30.

[11] T. Gollisch, M. Meister, Eye smarter than scientists believed: neural computations in circuits of the retina, Neuron 65 (2) (2010) 150–164.

[12] W. Levick, Receptive fields and trigger features of ganglion cells in the visual streak of the rabbit's retina, J. Physiol. 188 (3) (1967) 285–307.

[13] B. Roska, M. Meister, J. Werner, L. Chalupa, The retina dissects the visual scene into distinct features, New. Vis. Neurosci. (2014) 163–183.

[14] J.R. Sanes, R.H. Masland, The types of retinal ganglion cells: current status and implications for neuronal classification, Annu. Rev. Neurosci. 38 (2015) 221–246.

[15] S. Deny, U. Ferrari, E. Mace, P. Yger, R. Caplette, S. Picaud, G. Tkačik, O. Marre, Multiplexed computations in retinal ganglion cells of a single type, Nat. Commun. 8 (1) (2017) 1964.

[16] M.B. Herscher, T.P. Kelley, Functional electronic model of the frog retina, IEEE Trans. Mil. Electron MIL-7 (2 & 3) (1963) 98–103, https://doi.org/10.1109/TME.1963.4323057.

[17] C. Enroth-Cugell, J.G. Robson, Functional characteristics and diversity of cat retinal ganglion cells. basic characteristics and quantitative description, Investig. Ophthalmol. Vis. Sci. 25 (3) (1984) 250–267.

[18] R. Eckmiller, D. Neumann, O. Baruth, Tunable retina encoders for retina implants: why and how, J. Neural Eng. 2 (1) (2005) S91 http://stacks.iop.org/1741-2552/2/i=1/a=011.

[19] A. Martinez-Alvarez, A. Olmedo-Paya, S. Cuenca-Asensi, J.M. Ferrandez, E. Fernandez, Retinastudio: a bioinspired framework to encode visual information, Neurocomputing 114 (Supplement C) (2013) 45–53 searching for the interplay between neuroscience and computation https://doi.org/10.1016/j.neucom.2012.07.035 http://www.sciencedirect.com/science/article/pii/S0925231212007850.

[20] S. Nirenberg, C. Pandarinath, Retinal prosthetic strategy with the capacity to restore normal vision, Proc. Natl. Acad. Sci. 109 (37) (2012) 15012–15017.

[21] A. Wohrer, P. Kornprobst, Virtual retina: a biological retina model and simulator, with contrast gain control, J. Comput. Neurosci. 26 (2) (2009) 219–249.

[22] H. Lorach, R. Benosman, O. Marre, S.-H. Ieng, J.A. Sahel, S. Picaud, Artificial retina: the multichannel processing of the mammalian retina achieved with a neuromorphic asynchronous light acquisition device, J. Neural Eng. 9 (6) (2012) 066004.

[23] E. Chichilnisky, A simple white noise analysis of neuronal light responses, Netw. Comput. Neural Syst. 12 (2) (2001) 199–213.

[24] L. Paninski, Maximum likelihood estimation of cascade point-process neural encoding models, Netw. Comput. Neural Syst. 15 (4) (2004) 243–262.

[25] D.L. Ringach, M.J. Hawken, R. Shapley, Receptive field structure of neurons in monkey primary visual cortex revealed by stimulation with natural image sequences, J. Vis. 2 (1) (2002) 2–2.

[26] J.M. Salisbury, S.E. Palmer, Optimal prediction in the retina and natural motion statistics, J. Stat. Phys. 162 (5) (2016) 1309–1323.

[27] K.D. Simmons, J.S. Prentice, G. Tkačik, J. Homann, H.K. Yee, S.E. Palmer, P.C. Nelson, V. Balasubramanian, Transformation of stimulus correlations by the retina, PLoS Comput. Biol. 9 (12) (2013) e1003344.

[28] J.S. Prentice, O. Marre, M.L. Ioffe, A.R. Loback, G. Tkačik, M.J. Berry II, Error-robust modes of the retinal population code, PLoS Comput. Biol. 12 (11) (2016) e1005148.

[29] E.A. Real, Models of Visual Processing by the Retina, Ph.D. thesis Harvard University, 2012.

[30] L. Paninski, J.W. Pillow, E.P. Simoncelli, Maximum likelihood estimation of a stochastic integrate-and-fire neural encoding model, Neural Comput. 16 (12) (2004) 2533–2561, https://doi.org/10.1162/0899766042321797 http://doi.org/10.1162/0899766042321797.

[31] I. Bomash, Y. Roudi, S. Nirenberg, A virtual retina for studying population coding, PLoS One 8 (1) (2013) e53363.

[32] A. Olmedo-Paya, A. Martinez-Alvarez, S. Cuenca-Asensi, J.M. Ferrandez, E. Fernandez, Modeling the role of fixational eye movements in real-world scenes, Neurocomputing 151 (2015) 78–84.

[33] A. Asher, W.A. Segal, S.A. Baccus, L.P. Yaroslavsky, D.V. Palanker, Image processing for a high-resolution optoelectronic retinal prosthesis, IEEE Trans. Biomed. Eng. 54 (6) (2007) 993–1004.

[34] R.J. Greenberg, R.P. Williamson, J.H. Schulman, R.P. Rassool, L.J. Mandell, A.N. Seidman, Video processing methods for improving visual acuity and/or perceived image resolution, uS Patent 6 (Jul. 19 2005) 920 358.

[35] N. Barnes, An overview of vision processing in implantable prosthetic vision, Image Processing (ICIP), 2013 20th IEEE International Conference on, IEEE, 2013, pp. 1532–1535.

[36] R. Sarpeshkar, L. Turicchia, S. Mandal, Coding for visual prostheses, uS Patent 8 (Apr. 15 2014) 700 166.

[37] Z. Wei, D.-J. Lee, B. Nelson, Fpga-based real-time optical flow algorithm design and implementation, J. Multimed. 2 (5) (2007) 38–45.
[38] P. Possa, S. Mahmoudi, N. Harb, C. Valderrama, P. Manneback, A multi-resolution fpga-based architecture for real-time edge and corner detection, IEEE Trans. Comput. 63 (10) (2014) 2376–2388.
[39] T. Baden, P. Berens, K. Franke, M.R. Rosón, M. Bethge, T. Euler, The functional diversity of retinal ganglion cells in the mouse, Nature 529 (7586) (2016) 345.
[40] B. Cleland, W. Levick, Properties of rarely encountered types of ganglion cells in the cat's retina and on overall classification, J. Physiol. 240 (2) (1974) 457–492.
[41] J. Canny, A computational approach to edge detection, IEEE Trans. Pattern Anal. Mach. Intell PAMI -8 (6) (1986) 679–698, https://doi.org/10.1109/TPAMI.1986.4767851.
[42] D. Sun, S. Roth, M.J. Black, Secrets of optical flow estimation and their principles, Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE, 2010, pp. 2432–2439.
[43] B.D. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, Proc 7th Intl Joint Conf on Artificial Intelligence, (IJ CAI), 1981.
[44] A.N. Tikhonov, V.Y. Arsenin, Solutions of Ill-Posed Problems, Winston, 1977.
[45] W. Reichardt, Evaluation of optical motion information by movement detectors, J. Comp. Physiol. 161 (4) (1987) 533–547.
[46] J.S. Kim, M.J. Greene, A. Zlateski, K. Lee, M. Richardson, S.C. Turaga, M. Purcaro, M. Balkam, A. Robinson, B.F. Behabadi, et al., Space–time wiring specificity supports direction selectivity in the retina, Nature 509 (7500) (2014) 331.
[47] W. Ng, C. Lee, Comment on using the uniformity measure for performance measure in image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 18 (9) (1996) 933–934.
[48] S. Gilles, reportRobust Description and Matching of Images, Ph. D. thesis, Dept. Eng. Sci., Univ. Oxford.
[49] J.W. Pillow, L. Paninski, V.J. Uzzell, E.P. Simoncelli, E. Chichilnisky, Prediction and decoding of retinal ganglion cell responses with a probabilistic spiking model, J. Neurosci. 25 (47) (2005) 11003–11013.
[50] T. Kreuz, D. Chicharro, C. Houghton, R.G. Andrzejak, F. Mormann, Monitoring spike train synchrony, J. Neurophysiol. 109 (5) (2013) 1457–1472.
[51] M. Mulansky, N. Bozanic, A. Sburlea, T. Kreuz, A guide to time-resolved and parameter-free measures of spike train synchrony, International Conference on Event-Based Control, Communication, and Signal Processing, (EBCCSP), 2015, pp. 1–8, , https://doi.org/10.1109/EBCCSP.2015.7300693 2015.
[52] T. Kreuz, J.S. Haas, A. Morelli, H.D. Abarbanel, A. Politi, Measuring spike train synchrony, J. Neurosci. Methods 165 (1) (2007) 151–161 https://doi.org/10.1016/j.jneumeth.2007.05.031 http://www.sciencedirect.com/science/article/pii/S0165027007002671.
[53] T. Kreuz, D. Chicharro, M. Greschner, R.G. Andrzejak, Time-resolved and time-scale adaptive measures of spike train synchrony, J. Neurosci. Methods 195 (1) (2011) 92–106 https://doi.org/10.1016/j.jneumeth.2010.11.020 http://www.sciencedirect.com/science/article/pii/S0165027010006564.
[54] J. Troy, G. Einstein, R. Schuurmans, J. Robson, C. Enroth-Cugell, Responses to sinusoidal gratings of two types of very nonlinear retinal ganglion cells of cat, Vis. Neurosci. 3 (3) (1989) 213–223.
[55] L. Turicchia, M. O'Halloran, D. Kumar, R. Sarpeshkar, A low-power imager and compression algorithms for a brain-machine visual prosthesis for the blind, Biosensing 7035 (2008) 703510–703513.
[56] J.A. Dowling, A. Maeder, W. Boles, Mobility enhancement and assessment for a visual prosthesis, in: A.A. Amini, A. Manduca (Eds.), Medical Imaging 2004: Physiology, Function, and Structure from Medical Images, SPIE, 2004, , https://doi.org/10.1117/12.534866 https://doi.org/10.1117/12.534866.
[57] J. Lindsey, S.A. Ocko, S. Ganguli, S. Deny, The effects of neural resource constraints on early visual representations, International Conference on Learning Representations, 2019.
[58] M. Beyeler, A. Rokem, G. Boynton, I. Fine, Learning to see again: biological constraints on cortical plasticity and the implications for sight restoration technologies, J. Neural Eng. 14 (5).
[59] S. Sekhar, A. Jalligampala, E. Zrenner, D. Rathbun, Correspondence between visual and electrical input filters of on and off mouse retinal ganglion cells, J. Neural Eng. 14 (4).
[60] I. Fine, G. Boynton, Pulse trains to percepts: the challenge of creating a perceptually intelligible world with sight recovery technologies, Philos. Trans. R. Soc. Biol. Sci. 370 (1677).
[61] J. Golden, C. Erickson-Davis, N. Cottaris, N. Parthasarathy, F. Rieke, D. Brainard, B. Wandell, E. Chichilnisky, Simulation of visual perception and learning with a retinal prosthesis, J. Neural Eng. 16 (2).
[62] S. Nirenberg, I. Bomash, Retinal encoder for machine vision, uS Patent 9 (Jan. 17 2017) 547 804.
[63] J. Herault, Biologically Inspired Computer Vision: Fundamentals and Applications, John Wiley & Sons, 2015.
[64] N.K. Medathati, H. Neumann, G.S. Masson, P. Kornprobst, Bio-inspired computer vision: towards a synergistic approach of artificial and biological vision, Comput. Vis. Image Understand. 150 (2016) 1–30.
[65] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436.
[66] A. S. Ecker, F. H. Sinz, E. Froudarakis, P. G. Fahey, S. A. Cadena, E. Y. Walker, E. Cobos, J. Reimer, A. S. Tolias, M. Bethge, A rotation-equivariant convolutional neural network model of primary visual cortex, arXiv. URL https://arxiv.org/abs/1809.10504.
[67] Z. Lai, H. Deng, Medical image classification based on deep features extracted by deep model and statistic feature fusion with multilayer perceptron, Comput. Intell. Neurosci. 2018 (2018) 2061516.
[68] Z. Kong, T. Li, J. Luo, S. Xu, Automatic tissue image segmentation based on image processing and deep learning, J. Healthc. Eng. 2019 (2019) 2912458.
[69] L. Zou, J. Zheng, C. Miao, M.J. Mckeown, Z.J. Wang, 3d cnn based automatic diagnosis of attention deficit hyperactivity disorder using functional and structural mri, IEEE Access 5 (2017) 23626–23636.
[70] Y. Wang, B. Song, P. Zhang, N. Xin, G. Cao, A fast feature fusion algorithm in image classification for cyber physical systems, IEEE Access 5 (2017) 9089–9098.