Optimization-based Pairwise Interaction Point Process (O-PIPP): A Precise and Universal Retinal Mosaic Modeling Approach

Liuyuan He[1,2], Wenyao Wang[2*], Lei Ma[2, 3, 4*], Tiejun Huang[1, 2, 4]

Supplementary Text 1: Technical Details of Models

1. The d_{min} Model

In the process of the d_{min} model (Figure 1A), mosaic modeling was initialized with an empty area. During the updating stage, cells were arranged one by one with a random position and a radius (d_{min}). The d_{min}, representing the minimal distance between the cell and other settled cells, also known as the "exclusion zone", ensures that no other cells should be inside this zone. Specifically, during each update, one cell was placed at a random location. If the cell's nearest neighbor was farther than the d_{min}, the cell would be settled down in the location. Otherwise, we would need to attempt more times. The modeling process continued until all cells were settled in the mosaic, or the model couldn't find an eligible position for the waiting cell (we had set the maximum number of attempts to 10,000 times).

The number for each cell type in the mosaic was determined by the actual cell density and the area of the modeling region. The value of d_{min} was randomly sampled from a Gaussian distribution with a mean value (d_{mean}) and a standard deviation (d_{std}). We used a grid search to find the optimal parameters (d_{mean} and d_{std}). Taking horizontal cells as an example, candidates for d_{mean} ranged from 10 to 40 μm and candidates for d_{std} ranged from 1 to 10 μm, covering the observed statistics of horizontal cells (mean 23.57 μm and standard deviation 4.75 μm). We performed 50 simulations for

each combination of d_{mean} and d_{std}, and finally selected the combination that showed the least loss to the observed mosaic. Supplementary Table 1 displays the performance of the d_{min} model with different combinations of d_{mean} and d_{std} in the simulation for horizontal cells. In this case, the pair, $d_{mean} = 39\ \mu m$ and $d_{std} = 7\ \mu m$, yielded the lowest loss value (the red bold number) and we used them as the final parameters.

2. The PIPP Model

The PIPP model initialized all cells with random positions (Figure 1B). At each update step, the model selected a cell and attempted to reinsert it into the mosaic at a random position. The decision of whether to accept the new position followed the pairwise interaction point process, known as PIPP. Specifically, the interaction function $h(u)$ (1, 2) followed

$$h(u) = \begin{cases} 0, & u < \delta \\ 1 - \exp\left(-\left(\frac{u-\delta}{\phi}\right)^{\alpha_h}\right), & u \geq \delta \end{cases}, \quad (1)$$

where u is the distance between two cells. δ, ϕ, and α_h are the hyperparameters. The acceptance probability $p(x_i)$ of the i-th cell was

$$p(x_i) = \prod_{j=1, j\neq i}^{N} h\left(\left\|x_i - x_j\right\|\right), \quad (2)$$

where x_i denotes the position of the i-th cell, x_j denotes the position of others excluding the i-th cell, and N denotes the total number of cells. Following this formation, the acceptance probability $p(x_i)$ would be smaller if the cell was closer to neighbors. The PIPP model updated all cells in an iteration and terminated if the number of iterations reached a preset value (we had set the number to 20 iterations, as described in (2)). When compared with other methods, we chose the optimal mosaic with the lowest loss value from all iterations instead of relying on the result of the final iteration.

When we determined the hyperparameters of PIPP, taking horizontal cells as an example (Supplementary Figure 5), we first used the *spatstat* package in R to accomplish the non-parametric estimation (the red curve in Supplementary Figure 5) as described in (2). Then, we used non-linear least squares fitting in Python to find the optimal values of δ, ϕ, and α_h based on the Equation 1 (the black curve in Supplementary Figure 5). Detailed instructions are available in the GitHub tutorials (https://github.com/heliy/OPIPP/blob/main/tutorial/estimate_inter_ps.md).

3. Parameters in the O-PIPP Model

We conducted experiments on the horizontal cell mosaic to show the impact of O-PIPP parameters (T_0, α, P_r) on performance. For T_{min}, we fixed its value at 0.0001 to ensure enough optimization iterations. Each set of parameters underwent 50 simulation runs. We used the performance (the averaged L_F) and the time consumption (the averaged simulation time or the iteration numbers) as evaluation metrics.

In the first experiment, we assessed the impact of $T_0 \in [0.1, 0.3, 0.5, \ldots, 3.9, 4.1]$, and $\alpha \in [0.9, 0.95, 0.99]$ on the simulation results. We fixed $P_u = 0.01$ for all cases. In Supplementary Figure 3, it is evident that under these experimental conditions, there was a trade-off between performance and time consumption. Specifically, the impact of T_0 on performance saturated around 2.0 across different α values. While larger α values could further enhance performance, they also resulted in higher computational costs. Considering the balance between performance and time consumption, we chose this parameter set ($T_0 = 2.0$ and $\alpha = 0.95$) for our study.

In the second experiment, we evaluated how the update probability $P_u \in [0.001, 0.01, 0.05, 0.1, 0.2, 0.5, 1.0]$ alter the simulation results. This time we fixed $T_0 = 2.0$, and $\alpha = 0.95$ for all cases. Supplementary Figure 4 shows that when $P_u \in [0.001, 0.01]$, the O-PIPP yielded similar good performances and fast simulation. And we finally chose $P_u = 0.01$ in our study.

Supplementary Text 2: Details of Network Models

1. The neural dynamic system

The neural dynamic system of the model (Supplementary Figure 6) was based on the hierarchical Linear-Nonlinear (LN) model for primate GCs (3), which used subunits to present BCs in retinal circuits. We simplified the LN block with a temporal filer into a Weight-Nonlinear block, where a cell summed inputs with corresponding weights and then turned them into outputs nonlinearly. Specifically, the response of the j-th bipolar cell at time t was

$$o_{bc}^{j}(t) = f_{bc}\left(\sum_{i=1}^{n} \frac{x_c^i(t)}{n}\right), \tag{3}$$

where $f_{bc}(\blacksquare)$ denotes the BC nonlinearity, n denotes the number of cones providing inputs, and $x_c^i(t)$ denotes the stimuli for the i-th cone. Similarly, the firing rate of the ganglion cell at time t was

$$o_{gc}(t) = f_{gc}\left(\sum_{j=1}^{m} \frac{o_{bc}^{j}(t)}{m}\right), \tag{4}$$

where $f_{gc}(\blacksquare)$ denotes the GC nonlinearity, m denotes the number of BCs providing inputs. The nonlinearities f_{bc} and f_{gc} were sigmoid functions as

$$f(x) = \frac{1}{1 + \exp\left(-\frac{x-h}{s}\right)}, \tag{5}$$

where $h = 0$, $s = 0.1$ in f_{bc} and $h = 0.5$, $s = 0.01$ in f_{gc}. For simplification, all bipolar cells and ganglion cells were ON-type.

2. The Receptive Field Measurement

The RGC's RF in a network model was measured by the STA (Spike-Triggered Average), method(4) (Supplementary Figure 9). The STA method typically uses discrete spike trains, to sum up stimulus and yield the receptive field of a neuron. The firing rate of the ganglion cell is the number of spikes in a single time bin and can represent the weight of the time bin when we do an average alongside a spike train. Therefore, in our network model, we referred to the previous work(5) and used the firing rates as the weights, to sum up the stimulus and got the receptive field.

The stimulus spreaded out 20 x 20 pixels(6) (1 pixel covers a 23μm x 23μm area) and included 500,000 steps. Pixels were independently chosen from a white-noise distribution in [-0.5, 0.5] with mean value = 0. As the discrete-neuronal network model didn't include temporal filters, the RF was a spatial array without temporal information. Then we used the convexity index (CI) (6) to quantify spatial RF shapes. We labeled pixels whose weights were higher than the mean weight value by at least half the standard deviation and binarized them together into a convex hull. The CI value is the ratio of the pixel number of the convex hull over its area. We used the Gaussian kernel density estimation to infer the probability density of CI values from a population of RFs.

Supplementary Tables

Supplementary Table 1: Losses of d_{min}-simulated mosaics with different parameters.

d_{mean} \ d_{std}	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
10	2.334	2.376	2.539	2.617	2.784	2.842	2.741	2.788	3.016	2.908
11	2.117	2.188	2.279	2.427	2.329	2.499	2.639	2.621	2.811	2.665
12	1.883	1.882	2.025	2.093	2.198	2.282	2.406	2.571	2.637	2.593
13	1.76	1.725	1.775	1.806	1.945	2.035	2.194	2.353	2.433	2.693
14	1.459	1.576	1.591	1.658	1.747	1.799	2.006	2.241	2.369	2.35
15	1.226	1.284	1.295	1.448	1.531	1.763	1.973	2.087	2.181	2.231
16	1.092	1.081	1.178	1.271	1.363	1.617	1.753	1.935	2.131	2.178
17	0.96	0.886	1.015	1.195	1.257	1.44	1.577	1.721	1.945	2.055
18	0.83	0.802	0.922	0.972	1.015	1.273	1.383	1.612	1.815	1.962
19	0.745	0.737	0.708	0.79	0.905	1.196	1.288	1.468	1.52	1.815
20	0.663	0.65	0.628	0.697	0.826	0.962	1.07	1.269	1.517	1.656
21	0.636	0.578	0.586	0.562	0.726	0.86	0.986	1.181	1.375	1.581
22	0.595	0.506	0.521	0.525	0.637	0.757	0.916	1.067	1.332	1.586
23	0.601	0.492	0.438	0.431	0.521	0.654	0.819	0.916	1.005	1.403
24	0.665	0.524	0.459	0.408	0.47	0.605	0.738	0.798	1.039	1.186
25	0.743	0.558	0.458	0.4	0.392	0.476	0.642	0.801	0.965	1.078
26	0.873	0.608	0.428	0.378	0.361	0.456	0.555	0.676	0.849	1.146
27	0.946	0.677	0.475	0.371	0.343	0.406	0.507	0.623	0.845	1.052
28	1.07	0.74	0.524	0.369	0.333	0.374	0.443	0.578	0.814	0.938

29	1.115	0.796	0.531	0.406	0.325	0.343	0.469	0.535	0.742	0.806
30	1.25	0.846	0.615	0.393	0.363	0.343	0.344	0.487	0.528	0.76
31	1.484	0.9	0.648	0.42	0.36	0.311	0.361	0.437	0.599	0.731
32	1.906	1.054	0.655	0.472	0.351	0.329	0.348	0.417	0.499	0.668
33	2.366	1.268	0.727	0.495	0.38	0.319	0.352	0.424	0.499	0.638
34	3.12	1.556	0.789	0.537	0.419	0.35	0.321	0.362	0.468	0.634
35	4.269	1.984	0.988	0.559	0.376	0.356	0.33	0.356	0.416	0.537
36	5.253	2.866	1.261	0.619	0.41	0.346	0.329	0.355	0.419	0.473
37	7.08	3.747	1.668	0.689	0.428	0.366	0.32	0.294	0.357	0.487
38	8.845	4.817	2.114	0.845	0.462	0.334	0.307	0.331	0.374	0.449
39	10.124	6.496	3.01	1.154	0.502	0.378	**0.291**	0.311	0.354	0.439
40	11.687	7.993	3.831	1.635	0.613	0.4	0.322	0.308	0.342	0.404

Supplementary Table 2. Mean and standard deviation (STD) of metrics from O-PIPP-simulated mosaics.

Cell Types	**KL divergence of NN distances**		**KL divergence of VD areas**		**L_F**		**NNRI**		**VDRI**	
	Mean	STD	Mean	STD	Mean	STD	Mean	STD	Mean	STD
HC	0.024	0.004	0.009	0.001	0.034	0.004	4.688	0.181	4.595	0.139

Cholinergic AC	0.016	0.002	0.012	0.001	0.028	0.002	4.047	0.127	3.895	0.141
VGluT3 AC	0.034	0.002	0.026	0.003	0.060	0.004	3.578	0.137	3.243	0.104
AII AC	0.008	0.002	0.007	0.002	0.014	0.002	3.018	0.073	2.714	0.052
T2 BC	0.008	0.002	0.010	0.003	0.018	0.002	3.027	0.061	2.326	0.084
T3b BC	0.003	0.001	0.008	0.002	0.011	0.002	2.879	0.064	2.167	0.073
T4BC	0.024	0.003	0.032	0.007	0.056	0.009	2.489	0.082	2.210	0.062
WT Cone	0.003	0.001	0.039	0.015	0.042	0.015	4.209	0.043	2.830	0.412
RP Cone	0.015	0.005	0.029	0.007	0.044	0.008	2.618	0.088	0.810	0.101

Supplementary Table 3. Mean and standard deviation (STD) of metrics from PIPP-simulated mosaics.

Cell Types	**KL divergence of NN distances**		**KL divergence of VD areas**		L_F		**NNRI**		**VDRI**	
	Mean	STD	Mean	STD	Mean	STD	Mean	STD	Mean	STD

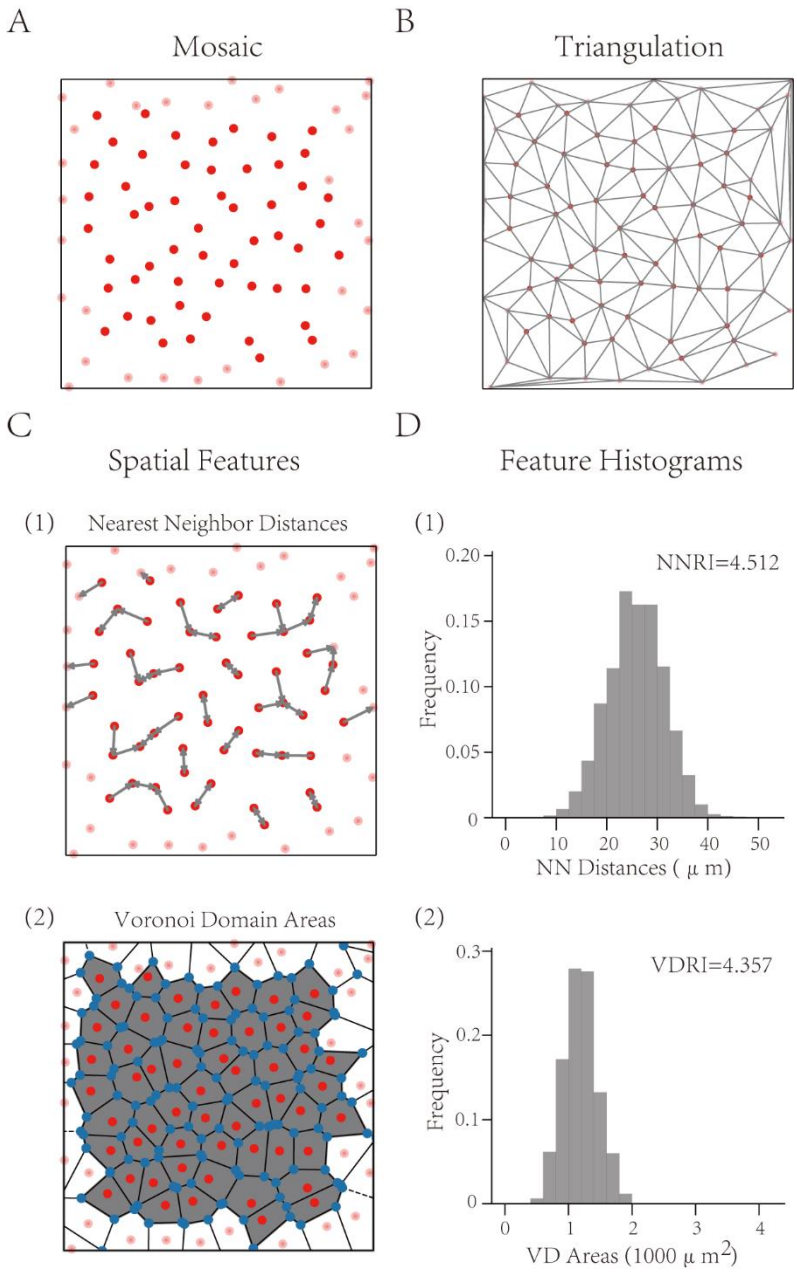
HC	0.228	0.077	0.100	0.057	0.328	0.106	4.582	0.555	4.705	0.511
Cholinergic AC	0.140	0.042	0.077	0.021	0.217	0.046	4.620	0.488	4.951	0.363
VGluT3 AC	0.245	0.067	0.325	0.060	0.571	0.095	3.786	0.406	3.741	0.361
AII AC	0.110	0.056	0.174	0.043	0.284	0.076	3.031	0.245	2.775	0.305
T2 BC	0.093	0.038	0.100	0.034	0.192	0.050	3.354	0.289	2.558	0.395
T3b BC	0.187	0.052	0.131	0.037	0.318	0.075	4.123	0.256	3.204	0.458
T4BC	0.256	0.077	0.359	0.128	0.615	0.145	2.825	0.373	2.768	0.406
WT Cone	0.018	0.017	0.055	0.023	0.073	0.037	4.126	0.140	2.867	0.335
RP Cone	0.186	0.173	0.358	0.330	0.544	0.502	2.982	0.376	1.444	0.668

Supplementary Table 4. Mean and standard deviation (STD) of metrics from d_{min}-simulated mosaics.

Cell Types	**KL divergence of NN distances**	**KL divergence of VD areas**	L_F	**NNRI**	**VDRI**

	Mean	STD	Mean	STD	Mean	STD	Mean	STD	Mean	STD
HC	0.190	0.073	0.134	0.072	0.322	0.094	6.135	0.798	4.601	0.519
Cholinergic AC	0.154	0.055	0.158	0.040	0.318	0.068	5.320	0.559	4.470	0.485
VGluT3 AC	0.237	0.073	0.525	0.080	0.790	0.124	4.855	0.676	3.844	0.484
AII AC	0.094	0.036	0.183	0.071	0.197	0.070	3.361	0.309	2.079	0.500
T2 BC	0.101	0.038	0.136	0.053	0.235	0.058	3.416	0.293	2.139	0.486
T3b BC	0.121	0.040	0.095	0.030	0.230	0.053	3.528	0.288	2.342	0.390
T4BC	0.151	0.059	0.362	0.210	0.336	0.084	2.650	0.292	1.694	0.384

Supplementary Figures



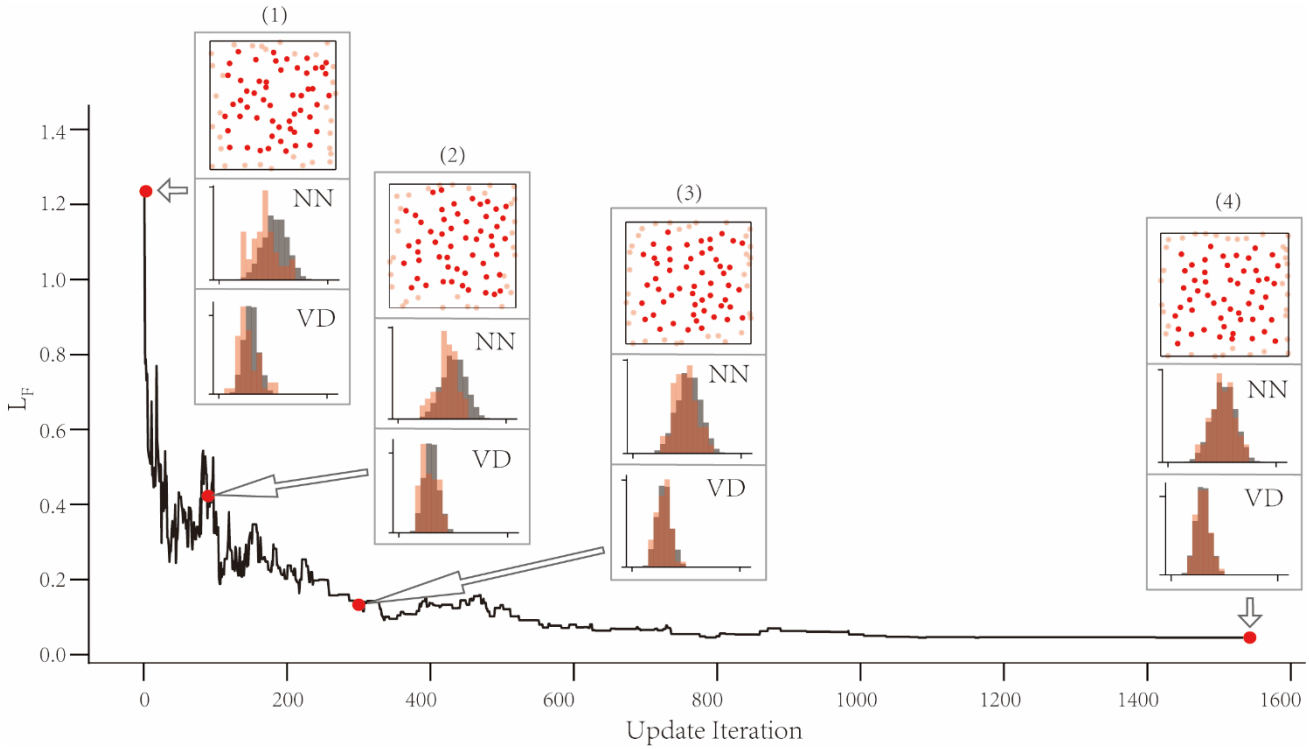
Supplementary Figure 1. Triangulation-related spatial features on a horizontal cell mosaic.

(A) A horizontal cell mosaic from (7). Transparent points denote cells close to the boundary of the area with a $200\mu m$ side length.

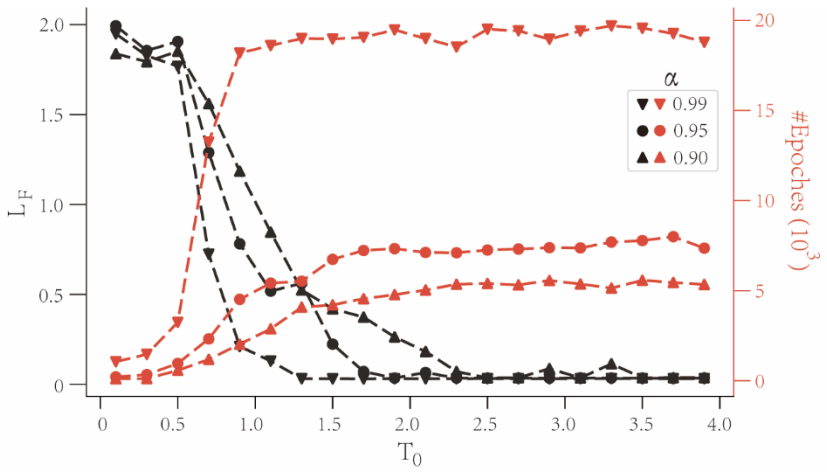
(B) The Delaunay triangulation results. Gray edges denote the cellular neighborhoods.

(C) The spatial features without boundary cells. (1) Nearest Neighbor (NN) distances, (2) Voronoi Domain (VD) areas.

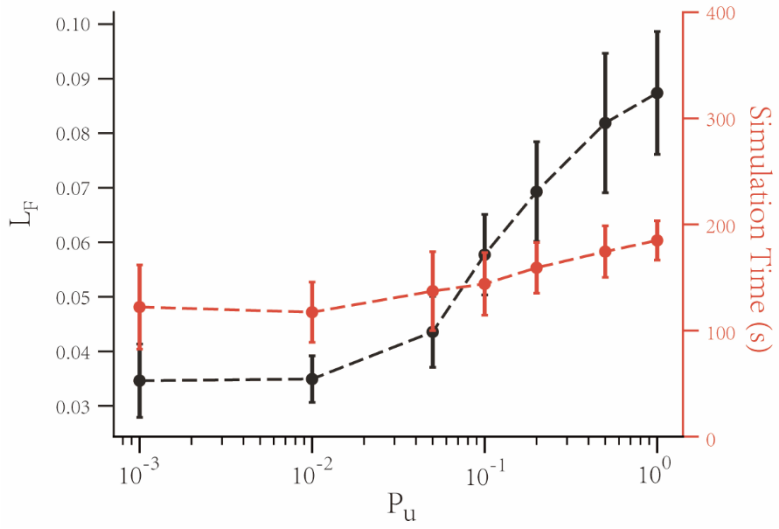
(D) Distributions of the population of features and related regularity indices (RI). (1) NN distances and the NNRI, (2) VD areas and the VDRI.



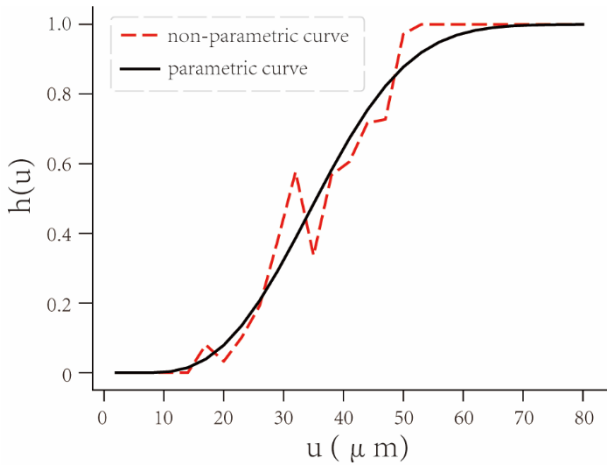
Supplementary Figure 2. The track of loss values along the optimization process of an O-PIPP-simulated HC mosaic.



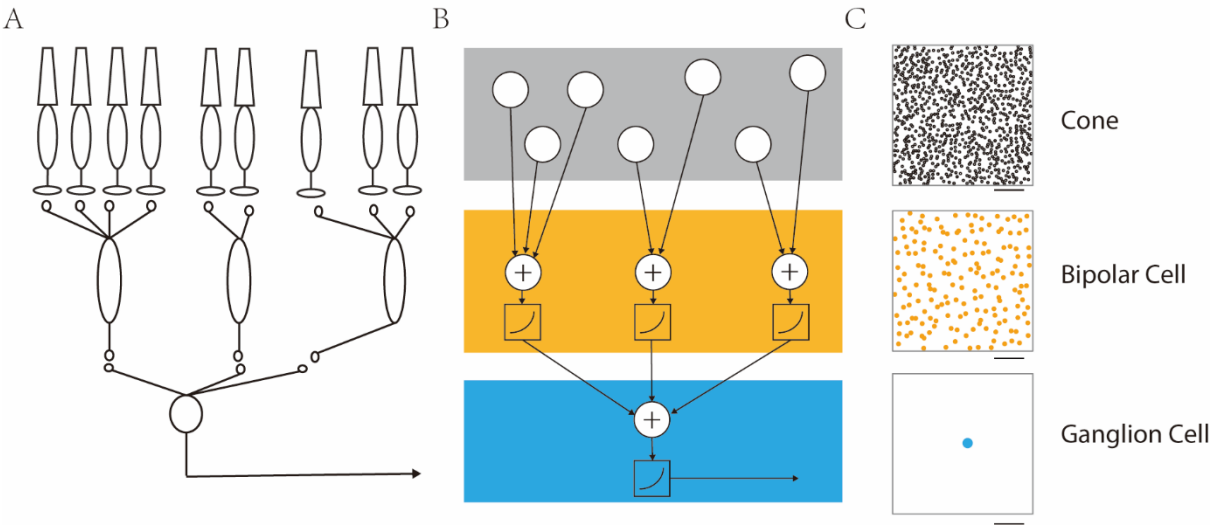
Supplementary Figure 3. Effect of T_0 and α on the result of O-PIPP.



Supplementary Figure 4. Effect of P_u on the result of O-PIPP.



Supplementary Figure 5. The interaction function $h(u)$ curves of horizontal cells.

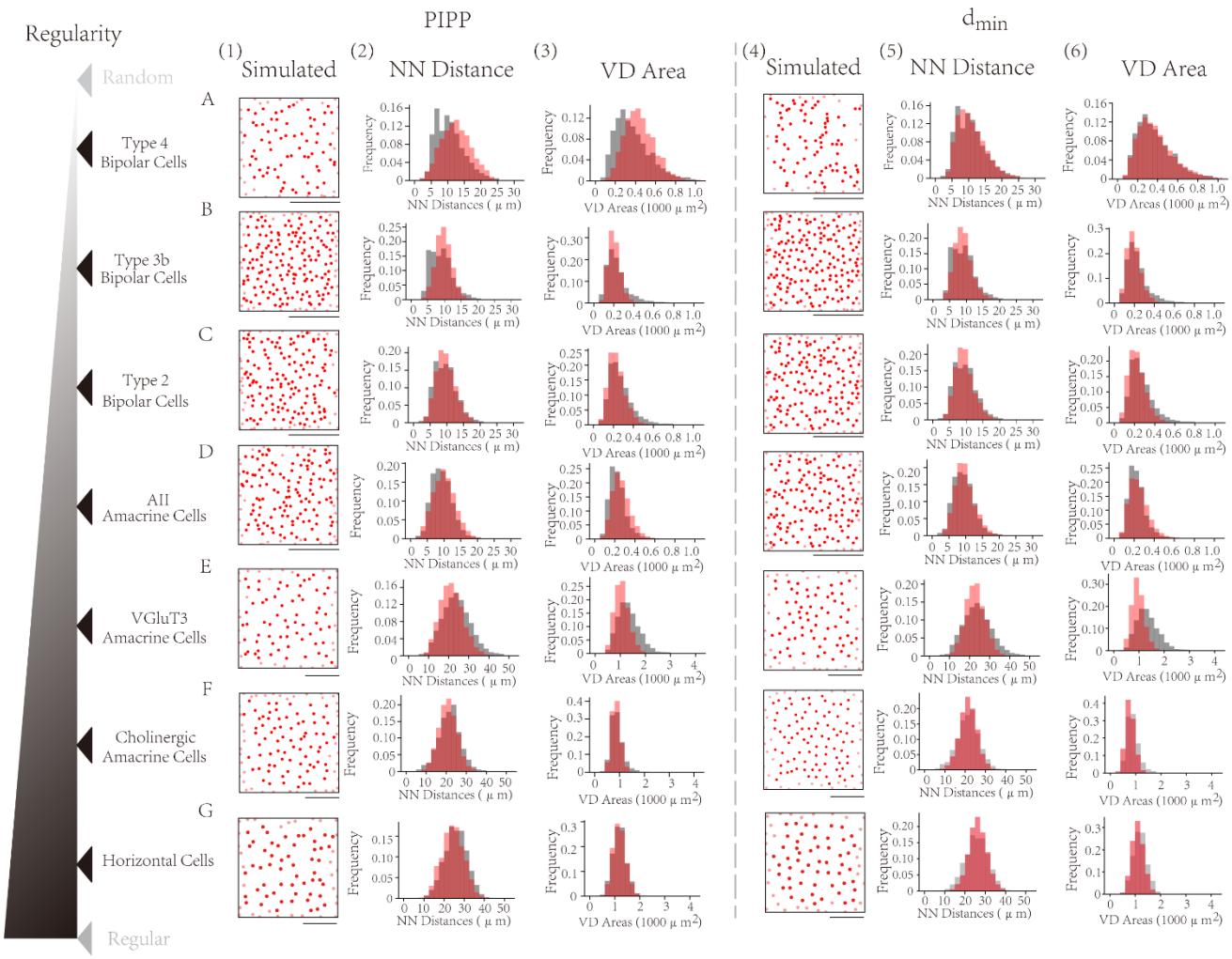


Supplementary Figure 6. The three-layer retinal discrete-neuronal network model with mosaics.

(A) Schematic of the retinal network with cone – bipolar cell – ganglion cell connections.

(B) The schema of the network model, modified from (3).

(C) The WT cone mosaic and O-PIPP-simulated BC mosaic in the network.



Supplementary Figure 7. Simulation results with the PIPP model and the d_{min} model on wild-type retinal mosaics.

(A to G) Cell types are ordered by the regularity index (NNRI) of the natural mosaic.

(1) The artificial mosaic generated by the PIPP approach.

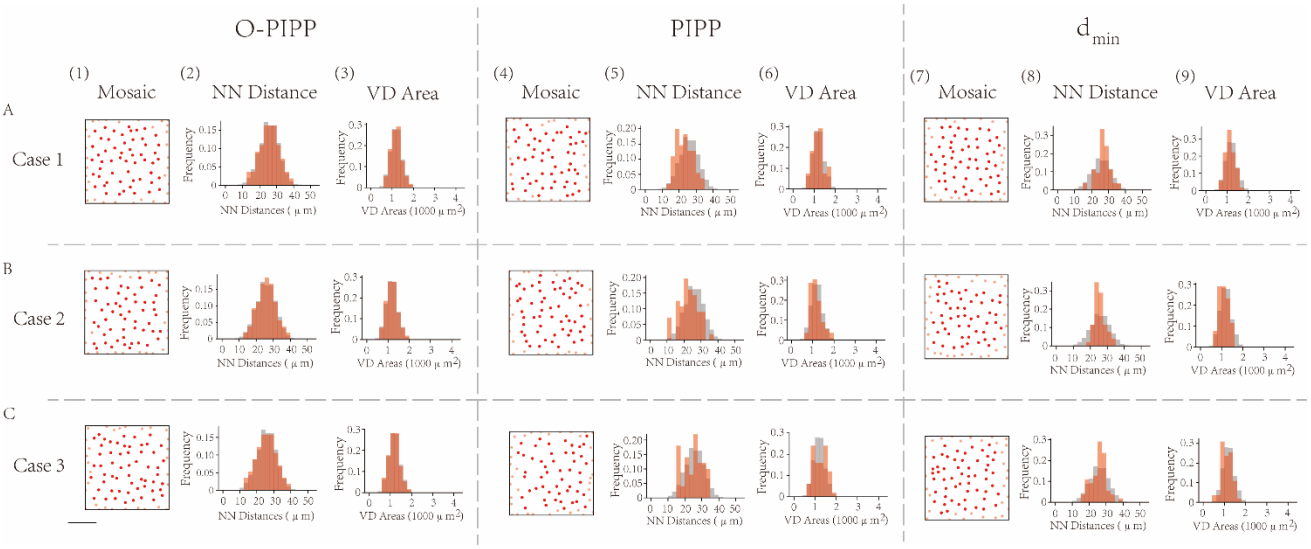
(2) The distribution of NN distances in natural mosaics (gray), overlaid by values extracted from PIPP-simulated mosaics (transparent red).

(3) The distribution of VD areas in natural mosaics (gray), overlaid by values extracted from PIPP-simulated mosaics (transparent red).

(4) The artificial mosaic generated by the d_{min} approach.

(5) The distribution of NN distances in natural mosaics (gray), overlaid by values extracted from d_{min}- simulated mosaics (transparent red).

(6) The distribution of VD areas in natural mosaics (gray), overlaid by values extracted from d_{min}- simulated mosaics (transparent red).



Supplementary Figure 8. Single simulation cases from three models on horizontal cell mosaics.

(A to C) Simulation cases on horizontal cell mosaics.

(1) Artificial mosaics generated by the O-PIPP approach.

(2) The distribution of NN distances in natural mosaics (gray), overlaid by values extracted from the O-PIPP-simulated mosaic (transparent red).

(3) The distribution of VD distances in natural mosaics (gray), overlaid by values extracted from the O-PIPP-simulated mosaic (transparent red).

(4) Artificial mosaics generated by the PIPP approach.

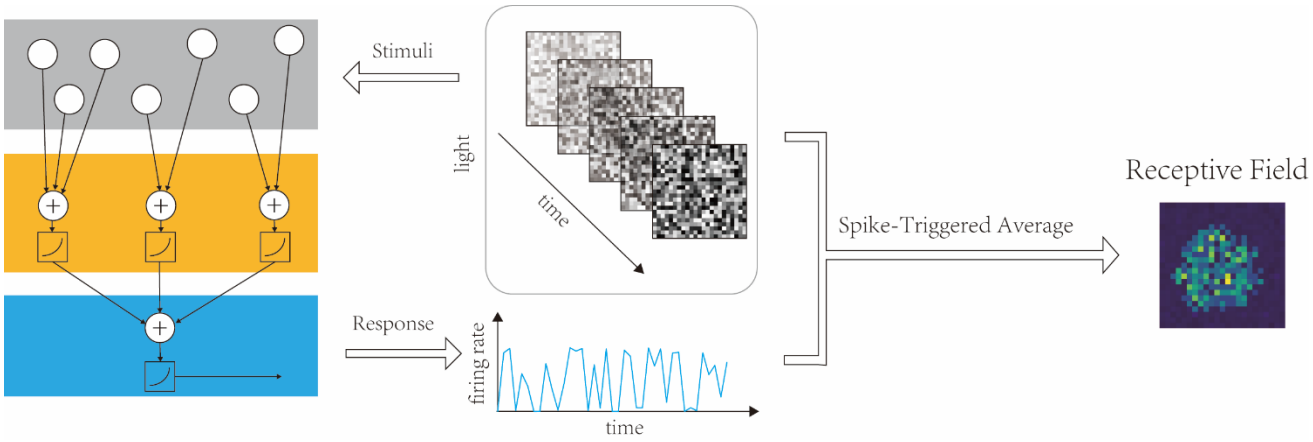
(5) The distribution of NN distances in natural mosaics (gray), overlaid by values extracted from the PIPP-simulated mosaic (transparent red).

(6) The distribution of VD distances in natural mosaics (gray), overlaid by values extracted from the PIPP-simulated mosaic (transparent red).

(7) Artificial mosaics generated by the d_{min} approach.

(8) The distribution of NN distances in natural mosaics (gray), overlaid by values extracted from the d_{min}-simulated mosaic (transparent red).

(9) The distribution of VD distances in natural mosaics (gray), overlaid by values extracted from the d_{min}-simulated mosaic (transparent red).



Supplementary Figure 9. Schematic of the Spike-Triggered Average (STA) method in the estimation of RFs.

References

1. Diggle PJ, Eglen SJ, Troy JB. Modelling the bivariate spatial distribution of amacrine cells. Case Studies in Spatial Point Process Modeling. 2006:215-33.
2. Eglen SJ, Diggle PJ, Troy JB. Homotypic constraints dominate positioning of on-and off-center beta retinal ganglion cells. Visual Neuroscience. 2005;22(6):859-71.
3. Freeman J, Field GD, Li PH, Greschner M, Gunning DE, Mathieson K, et al. Mapping nonlinear receptive field structure in primate retina at single cone resolution. Elife. 2015;4:e05241.
4. Chichilnisky, E. J. A simple white noise analysis of neuronal light responses. Network Computation in Neural Systems. 2001;12(2):199-213.
5. Schnitzer MJ, Meister M. Multineuronal firing patterns in the signal from eye to brain. Neuron. 2003;37(3):499-511.
6. Yu W-Q, Grzywacz NM, Lee E-J, Field GD. Cell type-specific changes in retinal ganglion cell function induced by rod death and cone reorganization in rats. Journal of neurophysiology. 2017;118(1):434-54.
7. Keeley PW, Eglen SJ, Reese BE. From random to regular: Variation in the patterning of retinal mosaics. Journal of Comparative Neurology. 2020;528(13):2135-60.