A Density-Driven Method for the Placement of Biological Cells Over Two-Dimensional Manifolds

Nicolas P. Rougier

A density-driven method for the placement of biological cells over two-dimensional manifolds

Nicolas P. Rougier*

February 23, 2018

Abstract

We introduce a graphical method originating from the computer graphics domain that is used for the arbitrary placement of cells over a two-dimensional manifold. Using a bitmap image whose luminance provides cell density, this method guarantees a discrete distribution of the positions of the cells respecting the local density. This method scales to any number of cells, allows one to specify arbitrary enclosing shapes and provides a scalable and versatile alternative to the more classical assumption of a uniform spatial distribution. The method is illustrated on a discrete homogeneous neural field, on the distribution of cones and rods in the retina and on the neural density of a flattened piece of cortex.

Keywords: Stippling, Voronoi, spatial distribution, density, cells, neurons, retina, cortex.

*INRIA Bordeaux Sud-Ouest, Talence, France – Nicolas.Rougier@inria.fr

1 Introduction

The spatial localization of neurons in the brain plays a critical role since their connectivity patterns may depend on their type and their position relatively to nearby neurons and areas (Ivenshitz & Segal, 2010). In the cortex, the probability of a connection existing between any two given areas declines sharply with distance (Markov et al., 2013), following an exponential decay with distance according to (Ercsey-Ravasz et al., 2013). For more local connections, such as interneurons, they generally have localized axonal arbors and interact mostly with close neighbours, depending on the distance (Jiang et al., 2015) from which a Gaussian probability of connection as a function of lateral distance can be derived (Potjans & Diesmann, 2012). Interestingly enough, whereas the neuroscience literature provides many data about the spatial distribution of neurons in different areas and species (e.g. (Pasternak & Woolsey, 1975) about the spatial distribution of neurons in the mouse barrel cortex, (McCormick, DeVaul, Shankle, & Fallon, 2000) about the neuron spatial distribution and morphology in the human cortex, (Blazquez-Llorca et al., 2014) about the spatial distribution of neurons innervated by chandelier cells), the computational literature exploiting such data is rather scarce and the spatial localization is hardly taken into account in most neural network models (be it computational, cognitive or machine learning models). One reason may be the inherent difficulty in describing the precise topography of a population such that most of the time, only the overall topology is described in terms of layers, structures or groups with their associated connectivity patterns (random, one to one, one to all, receptive fields, etc.). One can also argue that such precise localization is not necessary because for some models, it is not relevant (machine learning) while for some others, it may be subsumed into the notion of cell assemblies (Hebb, 1949) that represent the spatiotemporal structure of a group of neurons wired and acting together. Considering cell assemblies as the basic computational unit, one can consider local interactions to be subsumed into such assemblies and consequently, the exact spatial position of the neurons is not relevant. However, if cell assemblies allow to greatly simplify models, they also bring implicit limitations of which some have been highlighted in (Nallapu, Surampudi, & Rougier, 2017), such as for example the impossibility of having ambiguous representations (if such representations are identified with a single cell assembly) or to have topographic projections between two different groups. To overcome such potential limitations, we think the spatial localization of neurons is an important criterion worth to be studied because it could induce original connectivity schemes from which new computational properties can be derived as illustrated in Figure 2. However, before studying the

Figure 1: **Stippling.** According to Wikipedia[2], *Stippling is the creation of a pattern simulating varying degrees of solidity or shading by using small dots. Such a pattern may occur in nature and these effects are frequently emulated by artists.* The pair of boots (left part) have been first converted into a gray-level image and processed into a stippling figure (right part) using the weighted Voronoi stippling technique by (Secord, 2002) and replicated in (Rougier, 2017). Image from (Rougier, 2017) (CC-BY license).

influence of the spatial localization of neurons, it is necessary to first design a method for the arbitrary placement of neurons. This article introduces a graphical and scalable method for the automatic

placement of neurons (or any other type of cells actually) enforcing a user-provided density map. This graphical method is based on a stippling technique originating from the computer graphics domain for non-photorealistic rendering as illustrated in Figure 1.

2 Methods

Blue noise (Ulichney, 1987) is *an even, isotropic yet unstructured distribution of points* (Mehta, Wang, & Ramamoorthi, 2012) and has *minimal low frequency components and no concentrated spikes in the power spectrum energy* (Zhang et al., 2016). Said differently, blue noise (in the spatial domain) is a type of noise with intuitively good properties: points are evenly spread without visible structure (see figure 3 for the comparison of a uniform distribution and a blue noise distribution). This kind of noise has been extensively studied in the computer graphics domain and image processing because it can be used for object distribution, sampling, printing, half-toning, etc. One specific type of spatial blue noise is the Poisson disc distribution that is a 2D uniform point distribution in which all points are separated from each other by a minimum radius (see right part of figure 3). Several methods have been proposed for the generation of such noise, from the best in quality (dart throwing (Cook, 1986)) to faster ones (rejection sampling (Bridson, 2007)), see (Lagae & Dutré, 2008) for a review. An interesting variant of the Poisson disk distribution is an anisotropic distribution where local variations follow a given density function as illustrated in Figure 1 where the density function has been specified using the image gray levels. On the stippled image on the right, darker areas have a high concentration of dots (e.g. soles of the boots) while lighter areas such as the background display a sparse distribution of dots. There exist several techniques for computing such stippling density-driven patterns (optimal transport (Mehta et al., 2012), variational approach (Chen, Yuan, Choi, Liu, & Wang, 2012), least squares quantization (Lloyd, 1982), etc.) but the method proposed by (Secord, 2002) is probably the most straightforward and simple and has been replicated in (Rougier, 2017).

2.1 Centroidal Voronoi Tesselation

Considering a set of n points $P = \{P_i\}_{i\in[1,n]}$ on a finite domain $D \in \mathbb{R}^2$, the Voronoi tesselation $V(P) = \{V_i\}_{i\in[1,n]}$ of P is defined as:

$$\forall i \in [1,n], V_i = \{x \in D \mid \|x - P_i\| \leq \|x - P_j\|, \forall j \neq i\} \tag{1}$$

Reciprocally, the (unique) Delaunay triangulation $T(P) = \{T_i\}_{i\in[1,n]}$ of P is the dual graph of the Voronoi diagram and defined such that no point in P is inside the circumcircle of any triangles in $T(P)$. The centers of the circumcircles are equivalent to the Voronoi diagram, i.e. a partition of D into Voronoi cells. For each of the cell V_i, we can compute its centroid C_i which is the center of mass of the cell. A Voronoi tesselation is said to be centroidal when we have $\forall i \in [1,n], C_i = P_i$ (see figure 3).

For an arbitrary set of points, there is no guarantee that the corresponding Voronoi tesselation is centroidal but different methods can be used to generate a centroidal tesselation from an arbitrary set of points. One of the most straightforward and iterative methods is the Lloyd relaxation scheme (Lloyd, 1982):

1. The Voronoi diagram of the n points is computed
2. The centroid of each of the n Voronoi cell is computed.
3. Each point is moved to the corresponding centroid of its Voronoi cell
4. The method terminates if criterion is met (see below), else go to 1

[2] Stippling Wikipedia entry at `https://en.wikipedia.org/wiki/Stippling`

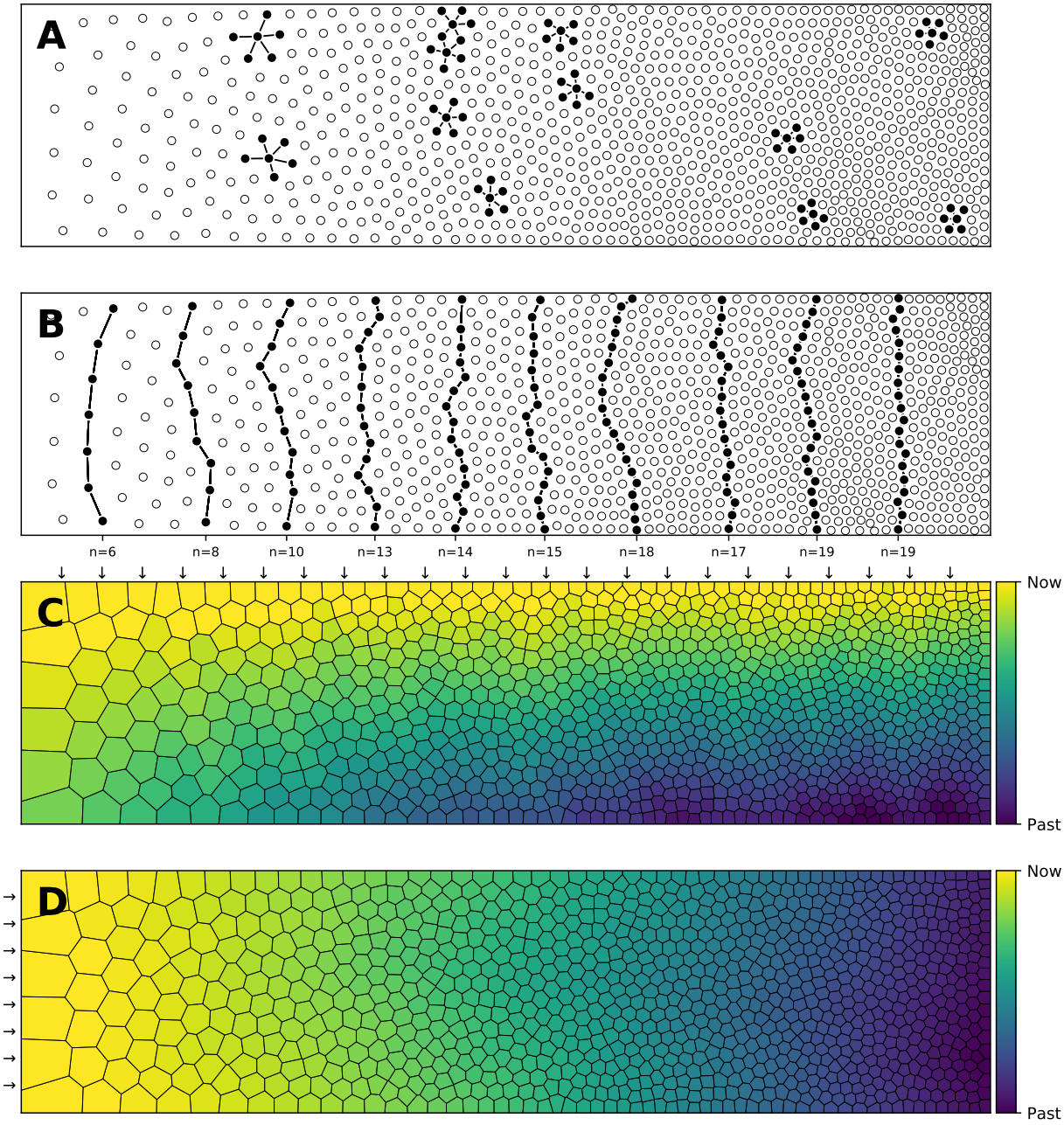

Figure 2: **Influence of spatial distribution on signal propagation. A.** A k-nearest neighbours (k=5) connectivity pattern shows mid-range connection lengths in low local density areas (left part) and short-range connection lengths in high density areas (right part). **B.** Shortest path from top to bottom using a k-nearest neighbours connectivity pattern (k=5). The lower the density, the shorter the path and the higher the density, the longer the path. On the far left, the shortest path from top to bottom is only 6 connections while this size triples on the far right to reach 19 connections. Said differently, the left part is the fast pathway while the right part is the slow pathway given some input data that would feed the architecture from the top. **C.** Due to the asymmetry of the cell positions, a signal entering on the top side (indicated with small arrows) travels at different speeds and will consequently reach the bottom side at different times. This represents a spatialization of time. Color represents time. **D.** Due to the asymmetry of the cell positions, a signal entering on the left side (indicated with small arrows) slows down while traveling before reaching the right side. This represents a compression of time and may serve as a short-term working memory. Color represents time.

The algorithm finishes when the maximum distance between points and centroids is less than a given threshold as illustrated in Figure 3. It is to be noted that because of numerical imprecisions, there is no guarantee that an arbitrary small threshold can be reached.

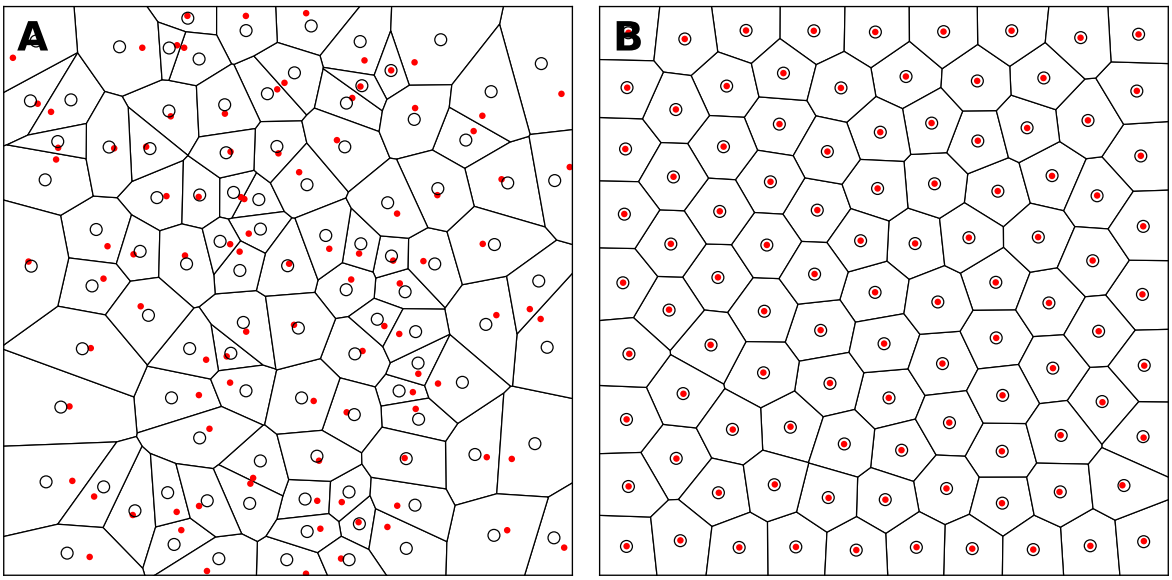

Figure 3: **Centroidal Voronoi Tesselation. A.** Voronoi diagram of a uniform distribution (n=100) where red dots represent the uniform distribution and white circles represent the centroids of each Voronoi cell. **B.** Centroidal Voronoi diagram where the point distribution matches the centroid distribution which constitutes a blue noise distribution (i.e. *a distribution that is roughly uniformly random with no preferred inter-point directions or distances* according to the definition of (Ebeida et al., 2014)). This figure has been obtained from the initial distribution on the left after 50 iterations of the Lloyd relaxation algorithm.

2.2 Weighted Centroidal Voronoi Tesselation

The weighted centroidal Voronoi tesselation, as illustrated in Figure 4, has been proposed in (Secord, 2002) and replicated in (Rougier, 2018). It is based on the Lloyd relaxation scheme with the notable difference that the centroids are now weighted according to the local density. This density information is provided using a bitmap image that represents the domain $D \in \mathbb{R}^2$. Any of the RGB channels of the image can be used to provide the density information at a specific integer coordinate position. By arbitrary convention, we'll consider the darker color (e.g. black) to have the the higher density. The method is then as follows:

1. The density image is resized if necessary (no interpolation)
2. The Voronoi diagram of the n points is computed
3. Each Voronoi cell is rasterized as a set of pixels
4. The weighted centroid is computed over each of the rasterized cell
5. Each point is moved to the corresponding centroid of its Voronoi cell
6. The method terminates if criterion is met, else go to 2

A different criterion for the termination is to use a fixed number of iterations as we did for all the examples introduced in this article (n=25).

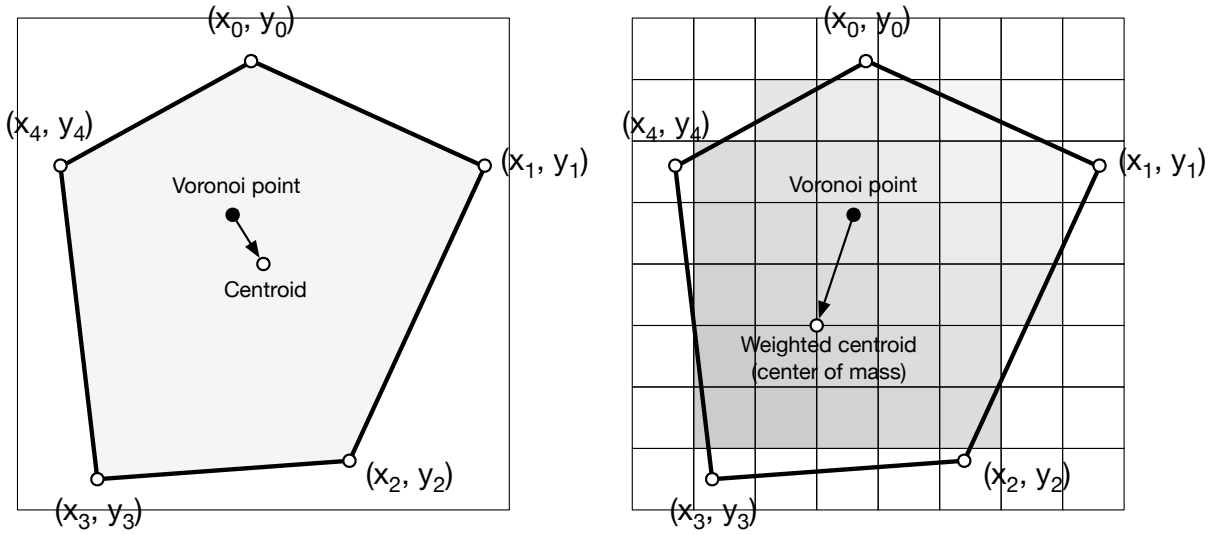

Figure 4: **Weighted centroid** The weighted centroid of a Voronoi cell is the center of mass computed over the rasterized cell.

Figure 4 illustrates the main difficulty in the method, that is, the rasterization of the cells and the computation of the weighted centroids. Since we use a bitmap image providing the density information and because the weighted centroids are computed over rasterized cells, it is quite obvious that the precision of the method is heavily dependent on the number of points and the size of the image. We estimated that a good precision can be reached if the mean number of pixels of a rasterized Voronoi cell is around 100 pixels (see figure 5). For example, if we have initially 1000 points to distribute and use a 100×100 input image, we would have only 10 pixels ($100 * 100/1000$) to compute the weighted centroid. Resizing first the image to 400×400 (without interpolation) makes this number to grow to 160 ($400 * 400/1000$). To obtain this 100 pixels estimation, we generated several polygons at different resolutions and compared the actual centroid (using its geometric definition) with the estimated centroid, considering a uniform density (whose center of mass is equal to the geometric centroid in such case).

Figure 6 shows the distribution of four populations with respective size 1000, 2500, 5000 and 10000 cells, using the same linear gradient as input. The local density is approximately independent of the total number of cells.

3 Results

We'll now illustrate the use of the proposed method on three different cases.

3.1 Case 1: Retina cells

The human retina counts two main types of photoreceptors, namely rods and cones (L-cones, M-cones and S-cones). They are distributed over the retinal surface in a non-uniform way, with a high concentration of cones (L-cones and M-cones) in the foveal region while the rods are to be found mostly in the peripheral region with a peak density at around 18-20° of foveal eccentricity. Furthermore, the respective size of those cells is different, rods being much smaller than cones. The distribution of rods and cones in the human retina has been extensively studied in the literature and is described precisely in a number of works (Curcio, Sloan, Kalina, & Hendrickson, 1990; Ahnelt & Kolb, 2000). Our goal here is not to fit the precise distribution of cones and rods but rather to give a generic procedure that can be eventually used to fit those figures, for a specific region of the retina or the whole retina. The main

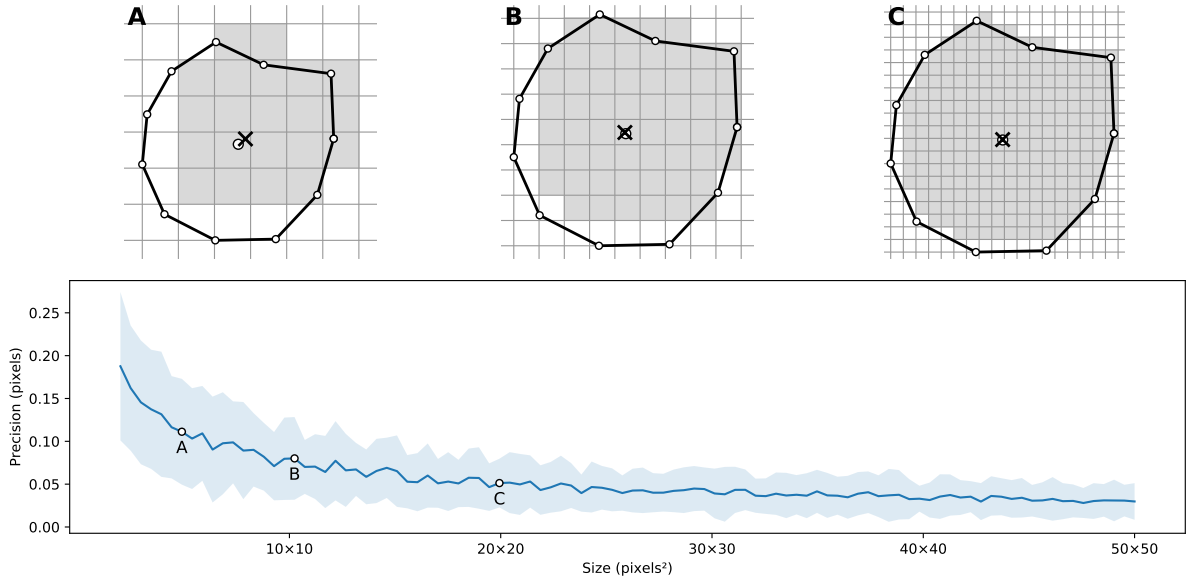

Figure 5: **Rasterized centroid precision.** The difference between the geometrical centroid (circle) and the centroid computed over the rasterized polygon (cross) is dependent on the size of the polygon.

difficulty is the presence of two types of cells having different sizes. Even though there exist blue-noise sampling procedures taking different sizes into account (Zhang et al., 2016), we'll use instead the aforementioned method using a two stages procedure as illustrated in Figure 7.

A first radial density map is created for the placement of 25 cones and the stippling procedure is applied for 15 steps to get the final positions of the 25 cones. A linear rod density map is created where discs of varying (random) sizes of null density are created at the positions of the cones. These discs will prevent the rods from spreading over these areas. Finally, the stippling procedure is applied a second time over the newly built density map for 25 iterations. The final result can be seen in Figure 7C where rods are tightly packed on the left, loosely packed on the right and nicely circumvent the cones.

3.2 Case 2: Neural field

Dynamic neural fields (DNF) describe the dynamics of a large population of neurons by taking the continuum limit in space, using coarse-grained properties of single neurons to describe the activity (Wilson & Cowan, 1972, 1973; Amari, 1977; Coombes, beim Graben, Potthast, & Wright, 2014). In this example, we consider a neural field with activity u that is governed by an equation of the type:

$$\tau \frac{\partial u(\mathbf{x},t)}{\partial t} = -u(\mathbf{x},t) + \int_{-\infty}^{+\infty} w(\mathbf{x},\mathbf{y}) f(u(\mathbf{y},t)) d\mathbf{y} + I(\mathbf{x}) + h$$

The lateral connection kernel w is a difference of Gaussians (DoG) with short range excitation and long range inhibition that reads:

$$w(\mathbf{x}) = I_e \exp^{\frac{-\mathbf{x}^2}{\sigma_e}} - I_i \exp^{\frac{-\mathbf{x}^2}{\sigma_i}}$$

The input $I(\mathbf{x})$ is a scaled white noise that reads:

$$I(\mathbf{x}) = I_s * uniform(noise)$$

and the function f is a clamped linear function between 0 and 1 such that:

$$f(x) = max(min(x,1),0)$$

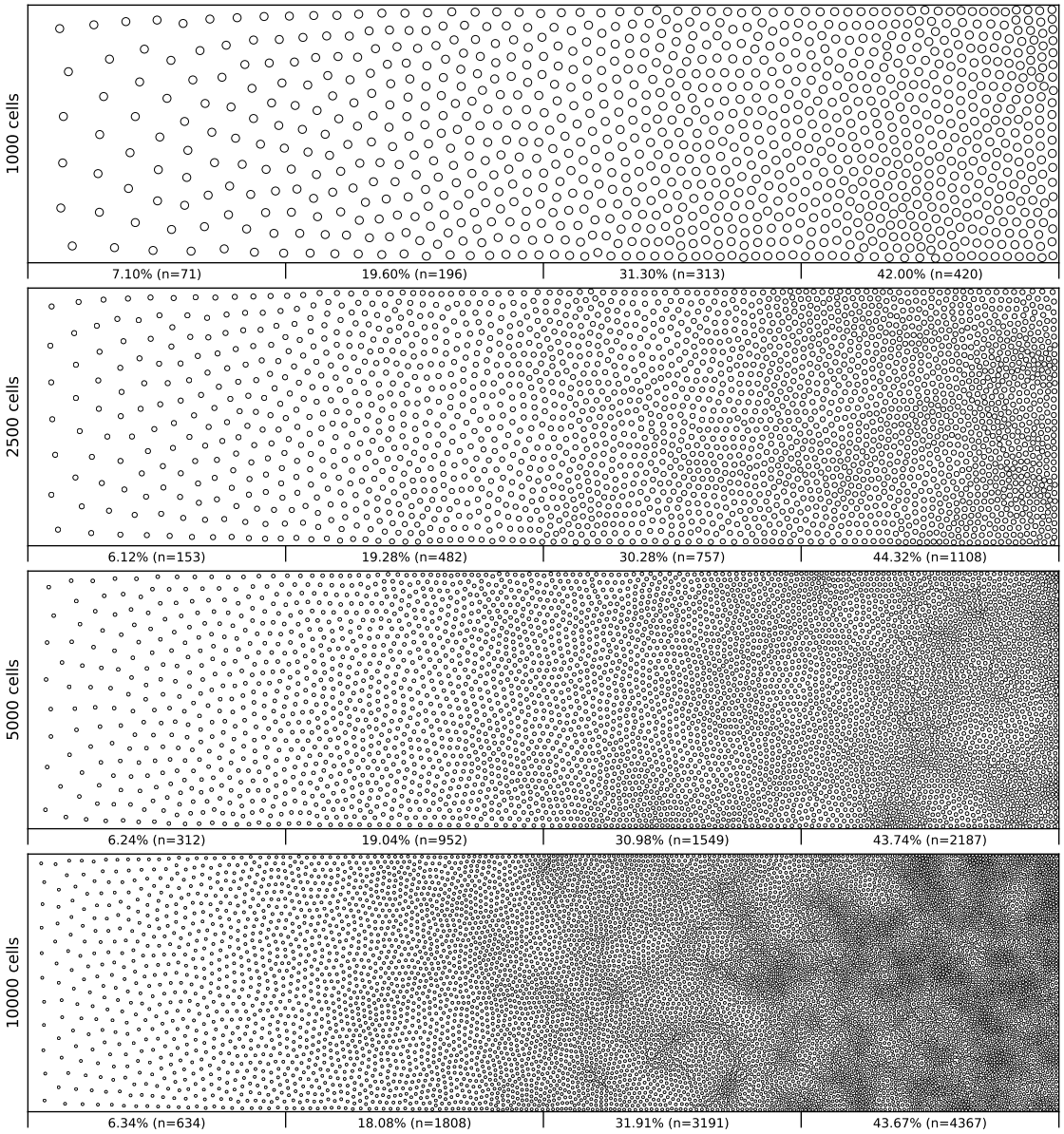

Figure 6: **Non-uniform distribution (linear gradient).** Different population distributions (size of 1000, 2500, 5000 and 10000 cells) using the same linear gradient as input have been computed. Each distribution has been split into four equal areas and the respective proportion and number of cells present in the area is indicated at the bottom of the area. The proportion of cells present in each area is approximately independent ($\pm 2.5\%$) of the overall number of cells.

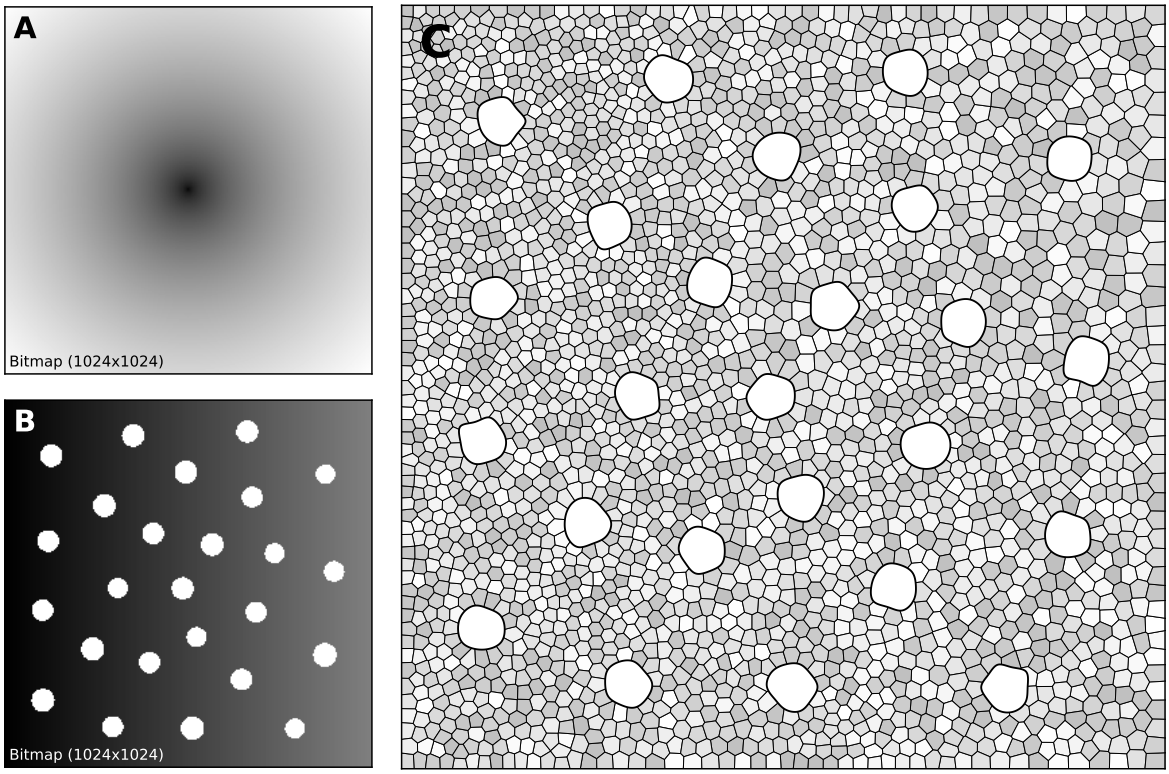

Figure 7: **Cones and rods distribution. A.** The density map for the placement of cones (n=25) is a circular and quadratic gradient with highest density in the center. **B.** The density map for the placement of rods (n=2500) is built using the rods distribution. Starting from a linear density, *holes* with different sizes are created at the location of each cone and prevent rods from spreading over these areas during the stippling procedure. **C.** Final distribution of cones and rods. Cones are represented as white blobs (splines) while rods are represented as Voronoi regions using random colors to better highlight the covered area.

In order to solve the neural field equation, the spatial domain was discretized into a 40×40 grid, the temporal resolution was set to $dt = 100ms$ and the simulation was run for $t = 10$ seconds. Relevant parameters are given in table 1. In Figure 8A, one can see the characteristic Turing patterns

Parameter	Name	Value
Grid size	n	40
Timestep	dt	100ms
Duration	t	10s
Time constant	τ	750ms
Resting potential	h	0
Input scaling	I_s	0.1
Noise level	N	0.1
Scaling factor	s	$40^2/n^2$
Sigma excitatory	σ_e	0.05
Scale excitatory	I_e	$0.15 \times s$
Sigma inhibitory	σ_i	0.085
Scale inhibitory	I_i	$0.05 \times s$

Table 1: Parameters for the neural fields

that have formed within the field. The number and size of clusters depend on the lateral connection kernel. Figure 8B shows the discretized and homogeneous version of the DNF where each cell has been assigned a position on the field, the connection kernel function and the parameters being the same as in the continuous version. The result of the simulation shown in Figure 8B is the normalized histogram of cell activities using 40×40 regular bins. One can see the formation of the Turing patterns that are similar to the continuous version. In Figure 8C however, the positions of the cells have been changed (using the proposed stippling method) such that there is an annulus of higher density. This is the only difference with the previous model. While the output can still be considered to be Turing patterns, one can see clearly that the activity clusters are precisely localized onto the higher density regions. Said differently, the functional properties of the field have been modified by a mere change in the structure. This suggests that the homogeneous condition of neural fields (that is the standard hypothesis in most works because it facilitates mathematical analysis) is actually quite a strong limitation that constrains the functional properties of the field.

3.3 Case 3: Cortical density

It has been shown in (C. E. Collins, Airey, Young, Leitch, & Kaas, 2010; Young, Collins, & Kaas, 2013; Christine E. Collins et al., 2016) that the neural density varies across and within cortical areas with an inverse relationship to the average neuron size: larger neurons take up more space and thus cannot be as densely packed as smaller neurons. (C. E. Collins et al., 2010) have studied the neural density in a cortical hemisphere of five primates and provided all the relevant data in the supplementary information. They dissected the flat hemisphere into a grid of 5×5mm piece and used an isotropic fractionator method to estimate the number of cells (neurons and non-neurons). To illustrate the method, we'll use the data from one of the two galagos that have been studied in order to produce a discrete distribution of sites enforcing the local measured density.

Using the Inkscape software[3], we opened the supplementary information PDF file from (C. E. Collins et al., 2010) and isolated the top of the figure S3 (galago 07-104). We renamed each individual patch according to the patch number indicated in the figure and saved the result as a SVG file. We took the first datasheet (galago 07-104) of the S1 dataset (Excel format) and converted it to a CSV format. Using the matplotlib library (Hunter, 2007), we produced a bitmap file (size 1000×1000 pixels see

[3] https://inkscape.org/en/

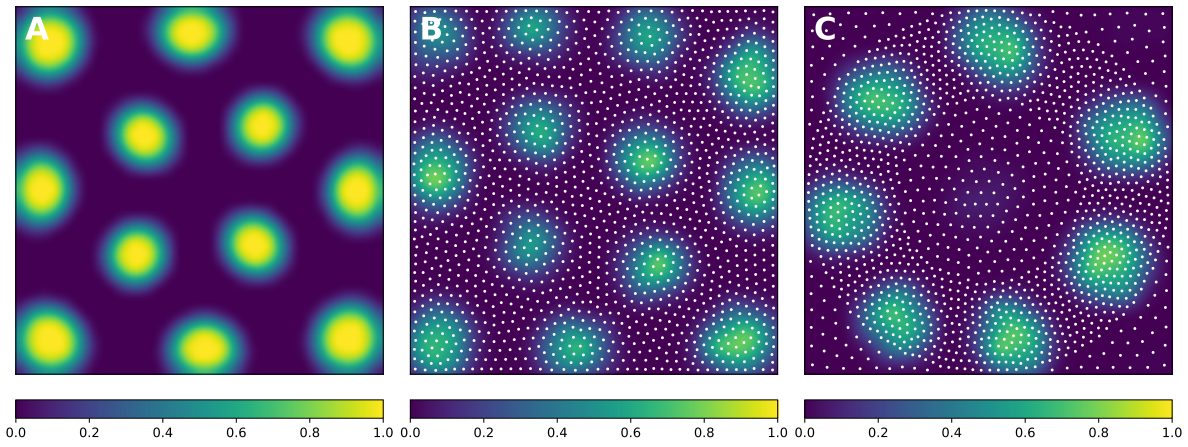

Figure 8: **Non-homogeneous discrete neural field**. Each plot has been smoothed using a bicubic filter. **A.** Turing patterns resulting from a continuous and homogeneous neural field with constant and noisy input. **B.** Turing patterns resulting from a discrete and homogeneous neural field with constant and noisy input. White dots indicate the position of the cells. Mean activity is computed from the histogram of neurons activity using 40×40 bins. **C.** Localized Turing patterns resulting from a discrete and non-homogeneous neural field with constant and noisy input. White dots indicate the position of the neurons. Mean activity is computed from the histogram of neuron activity using 40×40 bins.

figure 9-A) where each cortical patch was drawn using a gray level that corresponds to its normalized density, a density of 1.0 (black color) corresponding to the most densely populated area (area 2).

Using the Shapely library (Gillies et al., 2007), we computed the convex hull of the whole set of the 36 cortical patches as well as the centroid for each individual patch. The boundary of the convex hull was resampled such as to have 50 equidistant points along the outline. The density information for these points was computed using the density of the nearest centroid. A cubic two-dimensional interpolation was computed inside the convex hull using a Clough-Tocher differential scheme (Alfeld, 1984) and the result was saved as a bitmap file (size 1000×1000 pixels, see figure 9-C). We'll refer to this interpolation as the continuous case.

The two bitmap files were processed with the provided stippler script (Rougier, 2018) using the red channel for density information and run over 25 iterations using N=$\{$1000, 5000, 10000, 25000, 50000$\}$ sites. The result, for a single run, is a file with the 2-D coordinates of the N sites. From these coordinates, we computed the density for each of the original cortical patches by computing the patch area size and the number of sites inside. Results are indicated in table 2. Unsurprisingly, the accuracy of the distribution grows with the number of sites (with one exception in the continuous case). For N=50,000 sites, the difference between the actual density and the distribution is within a margin of 5%. In the continuous case however, it does not seem reasonable to expect a much higher accuracy than in the discrete (patch) case because the bitmap 1000×1000 has been interpolated using only 36 sites (patch centroids).

4 Discussion

We've introduced a graphical method for the placement of biological cells over a two-dimensional manifold enforcing a density distribution that is provided using a bitmap image and the method has been illustrated on three simple use cases. For a more realistic placement (i.e. actual three dimensional structures), the method could be adapted but it is to be noted that several methods have been recently proposed. Parametric anatomical modeling (Pyka, Klatt, & Cheng, 2014) allows one to model

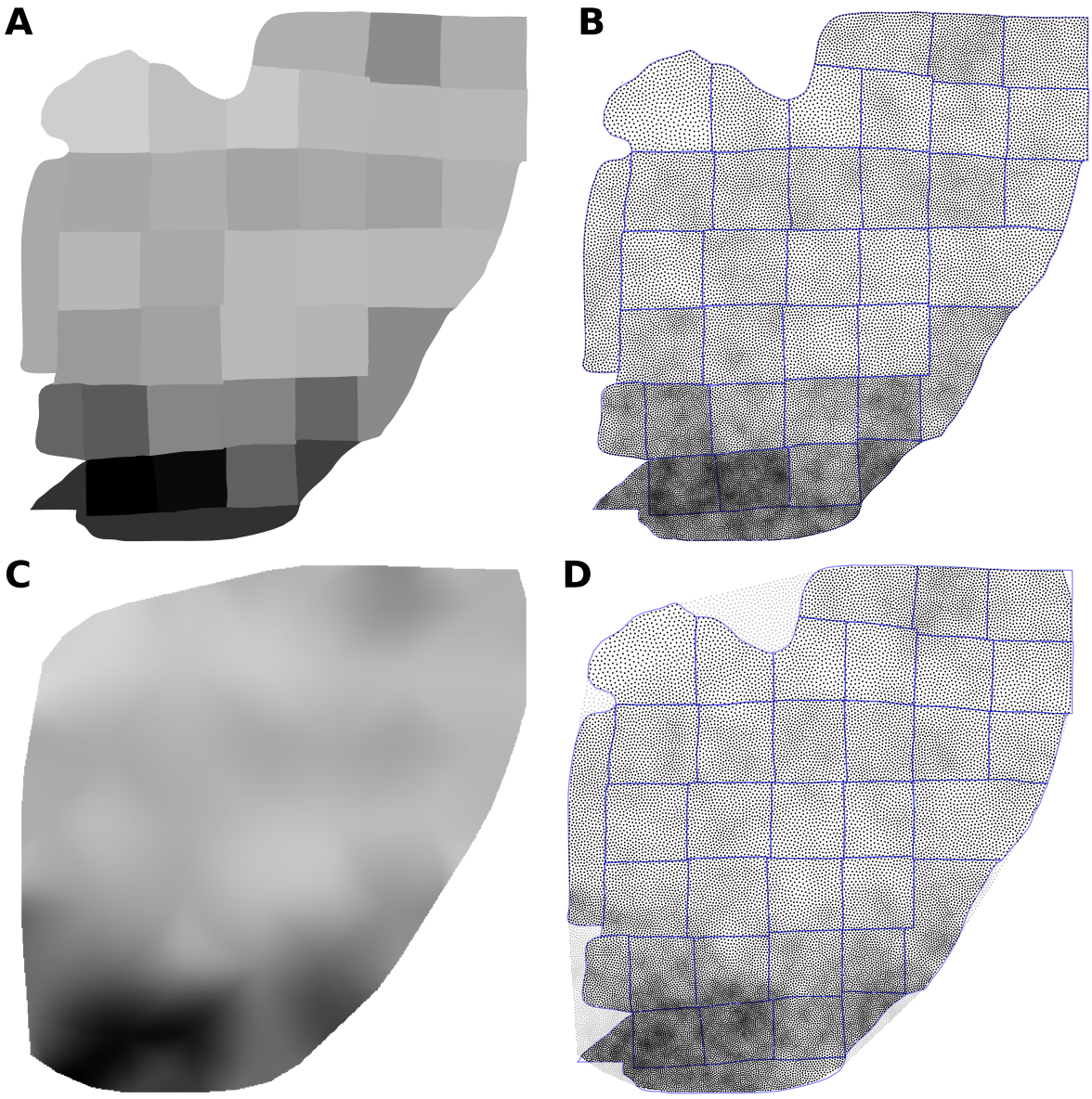

Figure 9: **Flattened cortex.** Data from (C. E. Collins, Airey, Young, Leitch, & Kaas, 2010), supplementary information. **A.** Each cortex piece was assigned a gray level corresponding to the normalized density (density 1.0 being assigned to the most densely populated area). The result was saved as a 1000×1000 bitmap file (PNG). **B.** Result of the stippling procedure for 25,000 sites and 25 iterations over the image generated in A. The mean difference for normalized density with actual data is 2.3% (±2.0%). Borders of the individual patches are drawn over the distribution (it is not an artefact). **C.** Continuous cubic interpolation of the normalized density over the convex hull of A, using the centroid of each patch for computing the interpolation. The result was saved as a 1000×1000 bitmap file (PNG). **D.** Result of the stippling procedure for 25,000 sites and 25 iterations over the image generated in C. The mean difference for normalized density with actual data is 2.9% (±2.8%). Outside sites (gray dots) are excluded and borders of the individual patches are drawn over the figure (it is not an artefact).

	Patch	Continuous	Computation time
N=1000	5.4% (±4.3%)	6.9% (±4.4%)	42s
N=5000	2.8% (±2.9%)	3.6% (±2.3%)	1m52
N=10000	2.8% (±1.8%)	4.0% (±3.0%)	2m55
N=25000	2.3% (±2.0%)	2.9% (±2.8%)	5m35
N=50000	0.8% (±0.6%)	2.6% (±2.0%)	8m44

Table 2: Mean difference between the actual (normalized) density and the mean neural density using a patch bitmap (figure 9-A) and a continuous cubic interpolated bitmap (figure 9-C) for a various number of sites. Computation times are only indicative and have been measured on a MacBook Pro with an Intel Core i7.

the anatomical layout of neurons as well as their projections while the work by (Schneider, Cuntz, & Soltesz, 2014) allows one to go even further down by taking into account the dendritic morphology of neurons. However, due to its simplicity and beyond a strict biological plausibility, we think the proposed method might be interesting for a number of models, intermediate between symbolic models and realistic models. Our intuition is that such topography may be an important aspect that needs to be taken into account and studied in order for a model to benefit from structural functionality. For example, the Figure 2 shows the influence of the spatial distribution on the signal propagation when considering a simple nearest neighbours connectivity scheme. Even though such connectivity is unlikely to exist inside the brain, it might be nonetheless worth to be studied because it may provide structural functionality, that is, a function that directly derives from the topography.

Notes: All figures were produced using the Python scientific stack, namely, SciPy (Jones, Oliphant, & Peterson, 2001), Matplotlib (Hunter, 2007) and NumPy (van der Walt, Colbert, & Varoquaux, 2011). All sources are available on GitHub at github.com/rougier/density-driven (Rougier, 2018).

References

Ahnelt, P. K. & Kolb, H. (2000). The mammalian photoreceptor mosaic-adaptive design. *Progress in Retinal and Eye Research*, *19*(6), 711–777. doi:10.1016/s1350-9462(00)00012-4

Alfeld, P. (1984). A trivariate clough—tocher scheme for tetrahedral data. *Computer Aided Geometric Design*, *1*(2), 169–181. doi:10.1016/0167-8396(84)90029-3

Amari, S.-I. (1977). Dynamics of pattern formation in lateral-inhibition type neural fields. *Biological Cybernetics*, *27*(2), 77–87. doi:10.1007/bf00337259

Blazquez-Llorca, L., Woodruff, A., Inan, M., Anderson, S. A., Yuste, R., DeFelipe, J., & Merchan-Perez, A. (2014). Spatial distribution of neurons innervated by chandelier cells. *Brain Structure and Function*, *220*(5), 2817–2834. doi:10.1007/s00429-014-0828-3

Bridson, R. (2007). Fast poisson disk sampling in arbitrary dimensions. In *ACM SIGGRAPH 2007 sketches on - SIGGRAPH '07*. ACM Press. doi:10.1145/1278780.1278807

Chen, Z., Yuan, Z., Choi, Y.-K., Liu, L., & Wang, W. (2012). Variational blue noise sampling. *IEEE Transactions on Visualization and Computer Graphics*, *18*(10), 1784–1796. doi:10.1109/tvcg.2012.94

Collins, C. E. [C. E.], Airey, D. C., Young, N. A., Leitch, D. B., & Kaas, J. H. (2010). Neuron densities vary across and within cortical areas in primates. *Proceedings of the National Academy of Sciences*, *107*(36), 15927–15932. doi:10.1073/pnas.1010356107

Collins, C. E. [Christine E.], Turner, E. C., Sawyer, E. K., Reed, J. L., Young, N. A., Flaherty, D. K., & Kaas, J. H. (2016). Cortical cell and neuron density estimates in one chimpanzee hemisphere. *Proceedings of the National Academy of Sciences*, *113*(3), 740–745. doi:10.1073/pnas.1524208113

Cook, R. L. (1986). Stochastic sampling in computer graphics. *ACM Transactions on Graphics*, *5*(1), 51–72. doi:10.1145/7529.8927

Coombes, S., beim Graben, P., Potthast, R., & Wright, J. (Eds.). (2014). *Neural fields*. Springer Berlin Heidelberg. doi:10.1007/978-3-642-54593-1

Curcio, C. A., Sloan, K. R., Kalina, R. E., & Hendrickson, A. E. (1990). Human photoreceptor topography. *The Journal of Comparative Neurology*, *292*(4), 497–523. doi:10.1002/cne.902920402

Ebeida, M. S., Awad, M. A., Ge, X., Mahmoud, A. H., Mitchell, S. A., Knupp, P. M., & Wei, L.-Y. (2014). Improving spatial coverage while preserving the blue noise of point sets. *Computer-Aided Design*, *46*, 25–36. doi:10.1016/j.cad.2013.08.015

Ercsey-Ravasz, M., Markov, N., Lamy, C., Van Essen, D., Knoblauch, K., Toroczkai, Z., & Kennedy, H. (2013). A predictive network model of cerebral cortical connectivity based on a distance rule. *Neuron*, *80*(1), 184–197. doi:10.1016/j.neuron.2013.07.036

Gillies, S. et al. (2007). Shapely: Manipulation and analysis of geometric objects. toblerity.org. Retrieved from https://github.com/Toblerity/Shapely

Hebb, D. O. (1949). *The organization of behavior: A neuropsychological theory*. New York: Wiley.

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing In Science & Engineering*, *9*(3), 90–95. doi:10.1109/MCSE.2007.55

Ivenshitz, M. & Segal, M. (2010). Neuronal density determines network connectivity and spontaneous activity in cultured hippocampus. *Journal of Neurophysiology*, *104*(2), 1052–1060. doi:10.1152/jn.00914.2009

Jiang, X., Shen, S., Cadwell, C. R., Berens, P., Sinz, F., Ecker, A. S., Patel, S., & Tolias, A. S. (2015). Principles of connectivity among morphologically defined cell types in adult neocortex. *Science*, *350*(6264), aac9462–aac9462. doi:10.1126/science.aac9462

Jones, E., Oliphant, T., & Peterson, P. (2001). SciPy: Open source scientific tools for Python. Retrieved from http://www.scipy.org

Lagae, A. & Dutré, P. (2008). A comparison of methods for generating poisson disk distributions. *Computer Graphics Forum*, *27*(1), 114–129. doi:10.1111/j.1467-8659.2007.01100.x

Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory*, *28*(2), 129–137. doi:10.1109/tit.1982.1056489

Markov, N. T., Ercsey-Ravasz, M., Lamy, C., Gomes, A. R. R., Magrou, L., Misery, P., Giroud, P., Barone, P., Dehay, C., Toroczkai, Z., Knoblauch, K., Essen, D. C. V., & Kennedy, H. (2013). The role of long-range connections on the specificity of the macaque interareal cortical network. *Proceedings of the National Academy of Sciences*, *110*(13), 5187–5192. doi:10.1073/pnas.1218972110

McCormick, B., DeVaul, R., Shankle, W., & Fallon, J. (2000). Modeling neuron spatial distribution and morphology in the developing human cerebral cortex. *Neurocomputing*, *32-33*, 897–904. doi:10.1016/s0925-2312(00)00258-7

Mehta, S. U., Wang, B., & Ramamoorthi, R. (2012). Axis-aligned filtering for interactive sampled soft shadows. *ACM Transactions on Graphics*, *31*(6), 1. doi:10.1145/2366145.2366182

Nallapu, B. T., Surampudi, B. R., & Rougier, N. P. (2017). The art of scaling up : A computational account on action selection in basal ganglia. In *2017 international joint conference on neural networks (IJCNN)*. IEEE. doi:10.1109/ijcnn.2017.7965835

Pasternak, J. F. & Woolsey, T. A. (1975). The number, size and spatial distribution of neurons in lamina IV of the mouse SmI neocortex. *The Journal of Comparative Neurology*, *160*(3), 291–306. doi:10.1002/cne.901600303

Potjans, T. C. & Diesmann, M. (2012). The cell-type specific cortical microcircuit: Relating structure and activity in a full-scale spiking network model. *Cerebral Cortex*, *24*(3), 785–806. doi:10.1093/cercor/bhs358

Pyka, M., Klatt, S., & Cheng, S. (2014). Parametric anatomical modeling: A method for modeling the anatomical layout of neurons and their projections. *Frontiers in Neuroanatomy*, *8*. doi:10.3389/fnana.2014.00091

Rougier, N. P. (2017). [Re] Weighted Voronoi Stippling. *ReScience*, *3*(1). doi:10.5281/zenodo.802285

Rougier, N. P. (2018). A density-driven method for the placement of biological cells over two-dimensional manifolds: Code. doi:10.5281/zenodo.1172461

Schneider, C. J., Cuntz, H., & Soltesz, I. (2014). Linking macroscopic with microscopic neuroanatomy using synthetic neuronal populations. *PLoS Computational Biology*, *10*(10), e1003921. doi:10.1371/journal.pcbi.1003921

Secord, A. (2002). Weighted voronoi stippling. In *Proceedings of the second international symposium on non-photorealistic animation and rendering - NPAR '02*. ACM Press. doi:10.1145/508530.508537

Ulichney, R. (1987). *Digital halftoning*. Cambridge, MA, USA: MIT Press.

van der Walt, S., Colbert, S. C., & Varoquaux, G. (2011). The NumPy array: A structure for efficient numerical computation. *Computing in Science & Engineering*, *13*(2), 22–30. doi:10.1109/mcse.2011.37

Wilson, H. R. & Cowan, J. D. (1972). Excitatory and inhibitory interactions in localized populations of model neurons. *Biophysical Journal*, *12*(1), 1–24. doi:10.1016/s0006-3495(72)86068-5

Wilson, H. R. & Cowan, J. D. (1973). A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue. *Kybernetik*, *13*(2), 55–80. doi:10.1007/bf00288786

Young, N. A., Collins, C. E. [Christine E.], & Kaas, J. H. (2013). Cell and neuron densities in the primary motor cortex of primates. *Frontiers in Neural Circuits*, *7*. doi:10.3389/fncir.2013.00030

Zhang, S., Guo, J., Zhang, H., Jia, X., Yan, D.-M., Yong, J., & Wonka, P. (2016). Capacity constrained blue-noise sampling on surfaces. *Computers & Graphics*, *55*, 44–54. doi:10.1016/j.cag.2015.11.002