VisNEST – Interactive Analysis of Neural Activity Data

Christian Nowke* Maximilian Schmidt† Sacha J. van Albada† Jochen M. Eppler†
Rembrandt Bakker† Markus Diesmann† Bernd Hentschel* Torsten Kuhlen*

*Virtual Reality Group, RWTH Aachen University, JARA – High Performance Computing

†Institute of Neuroscience and Medicine (INM-6) and Institute for Advanced Simulation (IAS-6), Jülich Research Center
JARA – Translational Brain Medicine

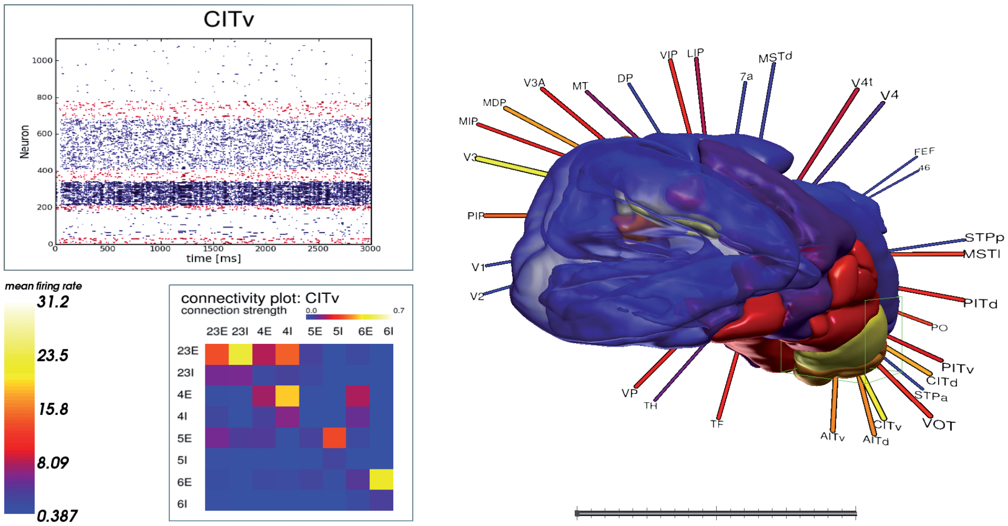

Figure 1: VisNEST's *control view* allows users to inspect neural activity linked to a geometric representation of a macaque brain. Textual annotations show anatomical area designators (right). By selecting a brain area, users may access the corresponding raster plot (top left) and connectivity of populations (bottom left).

ABSTRACT

The aim of computational neuroscience is to gain insight into the dynamics and functionality of the nervous system by means of modeling and simulation. Current research leverages the power of High Performance Computing facilities to enable multi-scale simulations capturing both low-level neural activity and large-scale interactions between brain regions. In this paper, we describe an interactive analysis tool that enables neuroscientists to explore data from such simulations. One of the driving challenges behind this work is the integration of macroscopic data at the level of brain regions with microscopic simulation results, such as the activity of individual neurons. While researchers validate their findings mainly by visualizing these data in a non-interactive fashion, state-of-the-art visualizations, tailored to the scientific question yet sufficiently general to accommodate different types of models, enable such analyses to be performed more efficiently. This work describes several visualization designs, conceived in close collaboration with domain experts, for the analysis of network models. We primarily focus on the exploration of neural activity data, inspecting connectivity of brain regions and populations, and visualizing activity flux across regions. We demonstrate the effectiveness of our approach in a case study conducted with domain experts.

Keywords: Interactive Visualization; Computational Neuroscience; Multi-View Visualization

Index Terms: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual Reality; I.6.5 [Simulation and Modeling]: Model Development; I.6.6 [Simulation and Modeling]: Simulation Output Analysis

1 INTRODUCTION

A main focus of contemporary neuroscience is to understand the relationships between the brain's structure and its dynamics on multiple scales, from microscopic circuits to networks at the brain scale. Large-scale simulations serve this purpose by representing neurons and synapses using reduced sets of dynamic equations and connecting them to form structured networks.

In spite of the heterogeneous data resulting from such simulation studies, little has been done to support neuroscientists with interactive tools for the analysis of these data. In fact, today's analysis workflows comprise a variety of different tools, ranging from standard shell commands to sophisticated analysis scripts, from simple, descriptive statistics to elaborate correlation analyses. The sheer amount of data combined with their complex inter-relationships makes an integrated analysis a demanding challenge.

In this paper, we describe the development of a visualization system for the analysis of simulated network models, including four distinct visualization designs that support the inspection of different model aspects. Our system design is based on an iterative, top-down development approach that emphasizes immediate expert feedback. As outlined above, the overarching challenge from the domain scientists' point of view is *data integration*: various pieces

---

*corresponding author: nowke@vr.rwth-aachen.de
†corresponding author: max.schmidt@fz-juelich.de

IEEE Symposium on Biological Data Visualization 2013
October 13 - 14, Atlanta, Georgia, USA
978-1-4799-1659-7/13/$31.00 ©2013 IEEE

of information – existing at different scales and levels of abstraction – have to be integrated into a holistic, interactive visualization.

In order facilitate this integrated analysis, we first identified the different data modalities (cf. Sec. 3.1). In close dialog with neuroscientists, we then gathered a number of key requirements, which we discuss in Sec. 3.2. Based on this input, we developed a high level system design for a modular visualization application that integrates multiple, linked views. Subsequently, we designed and implemented a number of different views along with suitable interactions. These are presented in Sec. 4. We discuss four different views, each of which addresses a specific data modality and user requirement. Throughout the development, we discussed new designs with users as early as feasible. Their feedback was used to streamline and optimize the respective visualization before starting the process over, addressing the next data modality. Using the resulting tool, domain scientists made several interesting discoveries, which were essentially facilitated by the fact that the system allowed them – for the first time – to interactively browse through and link different aspects of their data in a readily accessible form. Their findings are discussed in Sec. 5. While user feedback was generally positive, the discussions also revealed several open issues. We conclude this paper by discussing some of these as a motivation for future work in Sec. 6.

In summary, the contributions of this paper consist of a requirements analysis for the visualization of neuroscientific simulation data, the design of four custom-tailored visualizations for the presented use cases, and a demonstration of the overall effectiveness of the system based on domain expert feedback.

2 RELATED WORK

There are a number of research articles on the depiction of neuroscientific connectivity data. Nordlie et al. present *connectivity pattern tables*, a convenient and unifying way to formalize the representation of large-scale neuronal networks [15]. In contrast to our work, the authors focus on large connectivity matrices. Ciechomski et al. present methods for visualizing a neocortical column of a mouse brain in real-time. Their work focuses on providing neuroscientists with a way to explore circuits of multi-compartment neurons, to follow their electrical history, and record it [3]. Von Kapri et al. present a virtual reality application providing a 3D visualization of cortical layers, which reflects neuronal activity including cell membrane potential and spiking events of individual neurons (spikes are short electrical pulses by which neurons communicate) [22]. In contrast to these articles, we do not visualize circuits at the level of single neurons, since this is not yet feasible for simulations with millions of neurons and billions of synapses.

Kasiński et al. introduce a multiple-view approach for the analysis of dynamical processes in large, spiking neuronal networks [10]. They focus mainly on the 3D network representation and system performance. Aspect and scale changes are supported in conjunction with 3D navigation techniques. Furthermore, several graphical representations are tied together to form a consistent, highly integrated interactive system. This approach addresses the need for an interactive tool supporting neuroscientists in their analysis. However, our work deviates from Kasiński by focusing on network representation at the macroscopic scale, including connectivity and exchange of activity between brain regions.

Alper et al. explore visual designs for weighted graph comparison techniques in the context of brain connectivity data. They introduce a novel visual encoding for comparing different aspects of two graphs and evaluate it by means of a user study [1]. While Alper et al. target 2D representations, we use stereoscopic, 3D depictions of graphs.

Lasserre et al. focus on the exact visual reproduction of morphology of neurons in electrophysiological simulations [13] and Hernando et al. explore real-time rendering of massive amounts of neurons [8]. In contrast, we do not aim at realistic representations of individual neurons. Instead, we are interested in relating structure to dynamics at multiple scales, ranging from single neurons to brain regions, displaying averaged quantities to aid understanding.

A system that targets multi-scale visualization is neuroVISAS [18], a generic platform integrating different data modalities for the analysis of biologically realistic neuronal networks. A major focus is on pre-simulation data integration, i.e. on the integration of different ontologies with the goal to identify biologically consistent network connectivity and prepare it for simulation. The authors specifically note the ability to define simulations using the NEST engine [6]. In contrast, VisNEST targets only the analysis of simulation output data.

3 REQUIREMENTS ANALYSIS

In this section, we first summarize the simulation output that forms the input to our visualization system before discussing the requirements we gathered from domain experts.

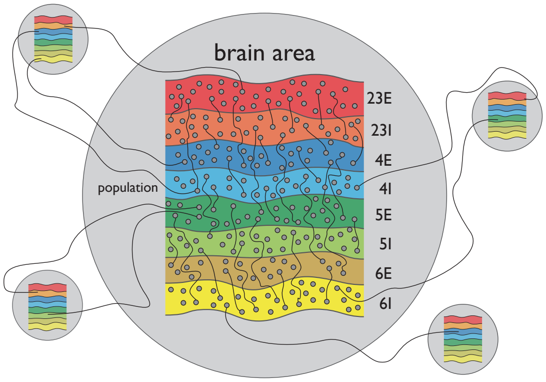

Figure 2: Schematic view of the input data. The entire data set consists of 32 brain areas. For the simulation, each area is represented by a layered model of eight neuron populations. Connections exist at all levels, i.e. within the same population, across populations of the same area, and between areas.

3.1 Input Data

The simulated neural model underlying this work mimics the visual cortex of a macaque monkey, subdivided into 32 areas following [4]. Each area contains eight neural populations, corresponding to excitatory and inhibitory neurons in each of four cortical layers. A schematic overview is given in Fig. 2. Population-specific connectivity was carefully compiled based on anatomical records [14, 21] and newly derived regularities. Intra-area connectivity is based on a model of the early sensory cortex [17] and has been adjusted to the macaque. The model is currently fixed to 32 areas. A range of observables from population-level activity to inter-area interactions is computed. Each set of simulation data consists of the following:

Spiking events — The simulation yields the spiking patterns of all neurons in each area and population over time.

Mean firing rate — Time-varying firing rates were computed at both the population level and the area level, by averaging spiking activity across neurons.

Polygonal geometry — The geometry of the visual areas of the macaque brain was taken from the Scalable Brain Atlas [2]. Each brain area was reconstructed from volumetric data. The resulting geometry was decimated for better rendering performance.

Functional descriptions — For each brain area, a text string is stored, containing a short description of the area's role in visual function. The option of showing these descriptions allows the user to place the areas in a functional context.

Area and population connectivity — Connectivity information is stored at both the area and population level. At each of these levels, connectivity is represented in two ways: as mean numbers of synapses impinging onto individual neurons (in-degrees), and as mean numbers of synapses established by individual neurons (out-degrees).

Hierarchy of the visual cortex — In a seminal work, Felleman and Van Essen proposed a visual cortical hierarchy which forms one choice for arranging the areas [4]. This choice is used for defining a layout of the visualization elements.

Flux — The simulation yields synaptic activity per unit time between areas, which we here refer to as *activity flux*. Conceptually, flux data is stored in the same way as area connectivity, but it is time-varying. Our data contains 32×32 fluxes for each time step, including self-flux, i.e. activity exchanged within an area.

3.2 Functional Requirements

After classifying the data modalities to be visualized, we started to collect functional requirements for the desired visualization system. In order to involve users as early and as regularly as possible, we established a monthly meeting schedule with our collaborators from neuroscience. A regular topic in these meetings was the identification, discussion, clarification, and prioritization of software requirements. Additionally, these meetings were used to assess progress and suggest improvements to already implemented functionality, an aspect that will be discussed in more detail in the next section. From initial discussions, we identified the following requirements, which are directly based on observations of the users' normal data analysis workflows.

R1: **Relate a geometric representation to activity data** — In order to explore spatial relationships among simulated areas, the system has to provide a geometric representation. Moreover, activity data should be tied to this geometric representation to provide a quick overview of the simulation run, and to enable comparisons with imaging data (e.g., from functional magnetic resonance imaging) that are commonly displayed on the cortical surface. Additionally, the geometric view should provide a means for selecting areas of interest, and for querying this selection for more detailed information.

R2: **Control simulation playback** — Navigation in time, including starting, stopping, and changing the playback speed of the simulation is essential in order to analyze the data effectively. Therefore, navigation techniques have to be provided that intuitively offer such functionality.

R3: **Visualize local and long-range connectivity** — Enabling the exploration of connectivity both within and between areas in an interactive fashion helps to form a detailed understanding of the model structure and its relation to the dynamics.

R4: **Visualize local activity** — The activity of an area should be resolvable into its constituent populations, i.e. mean firing rate per population should be readily accessible in the visualization.

R5: **Visualize area hierarchy** — The areas of the visual cortex form an approximate hierarchy defined by laminar connection patterns [4] and reflecting successive processing steps. Activation patterns can potentially be identified by visualizing areas according to such a hierarchy.

R6: **Visualize interactions between areas** — Activity exchange between areas is an important focus of large-scale brain modeling studies. In combination with inter-area connectivity, characterizing such interactions sheds light on the large-scale communication patterns of the simulated network.

R7: **Use techniques from the neuroscientists' workflow** — In order to ease the transition to new visualization designs, we strive to integrate well-known neuroscientific analysis methods, e.g. raster plots or connectivity matrices.

R8: **Provide means of intuitive interaction** — It should be possible to interactively link different aspects of the visualization in order to allow users to assess relationships between the underlying data modalities. Moreover, it should be possible to customize certain aspects of the visualization, for instance by adjusting lookup tables to enhance contrast in the range of interest.

4 SYSTEM DESIGN

In this section, we describe the visualization approaches that were designed in order to address the aforementioned requirements, and discuss how users interact with the system.

During initial discussions, we suggested building the target system on a Virtual Reality (VR) enabled software stack. The use of VR would combine stereoscopic presentation with direct 3D interaction, which potentially offers a more intuitive way to interact with 3D data (cf. **R8**). This choice was based on the following two arguments. First, we knew from previous collaborations with experts from other fields, e.g. computational fluid dynamics, that immersive visualization techniques were perceived as being beneficial for the understanding of complex, 3D data [7]. Second, recent work in the field, e.g. by Laha et al. supports this anecdotal evidence with a formal evaluation for the case of VR-based volume data exploration [11]. Although their work focuses on volume data, we hypothesize that VR technology analogously provides similar benefits for the visualization of 3D geometries which are part of our first requirement (cf. **R1**). However, we are not aware of a formal evaluation of the benefits of VR for this latter case. Nonetheless, the domain experts in this project agreed on a VR-enabled visualization tool provided it would be accessible via a standard desktop workstation as well.

In order to address the central challenge of data integration, we decided to design the visualization system to use multiple views to present information in its relevant context. Currently, four different views are available. The *control view* uses polygonal meshes to relate activity data to the brain areas' three-dimensional geometry. The *population view* provides an abstract overview of all populations in each brain area and shows their activity. Moreover, this view displays the connectivity of populations within a brain area. The *connectivity view* visualizes the connectivity of brain areas, whereas the *flux view* depicts time-varying activity flux between brain areas. Finally, we will discuss how users interact with these views.

4.1 Control View

The *control view* forms the central element of our visualization. Its main task is to link area activity data to three-dimensional brain area geometry as outlined in **R1**. Additionally, it provides the user with an intuitive way to select individual brain areas in order to inspect these in more detail (cf. **R8**).

A general mapping of activity data to area geometry is achieved by color coding. Per default we use the lookup table illustrated in Fig. 1, but this can interactively be changed by the user. Yet, the partial, mutual occlusion of brain areas and their complex shape make it difficult to recognize individual areas regardless of perspective. Therefore, we additionally map an area's activity value to its opacity so that areas with lower activity do not overly occlude

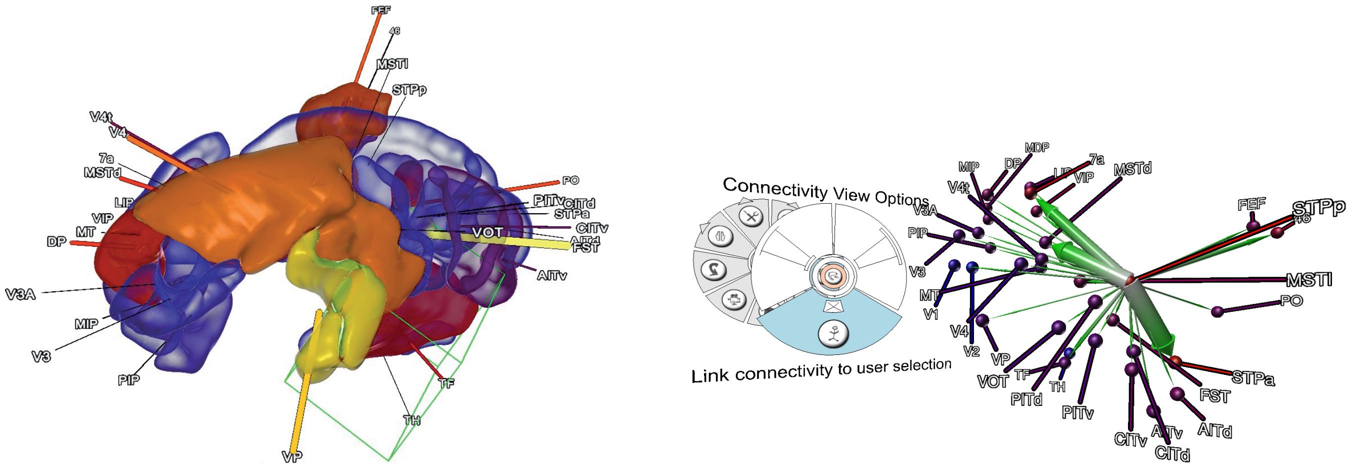

Figure 3: Left: *Control view* showing a semi-transparent rendering of vision-related areas of the macaque brain. Coloring of brain areas is tied to their activity. Right: *Connection view* showing anatomical connections from area STPp, with direction indicated by the arrows. Thickness of edges encodes connection strength. Connection visibility can be controlled by use of extended pie menus.

more active areas. Additionally, we chose to highlight the complex geometrical structure of the brain by using ideas of *angle-based transparency* as motivated by Hummel et al. [9]. This illustrative approach modulates the α-value based on local view direction and surface normal. This results in increased transparency of surface areas orthogonal to the direction of view, and decreased transparency where the surface of the area curves away from the view direction. However, it provides no possibility to control opaqueness based on average spiking activity. In contrast to the original approach, we introduce an α-modulation method which combines aspects of angle-based transparency and a mapping of activity data. Let α_{base} denote the base transparency, and $\overline{a}$ be the average spiking activity, which is normalized to the unit interval over all simulation time steps. Then we calculate the overall transparency α_{final} by

$$a = \begin{cases} \alpha_{\text{base}} + \overline{a} & \text{if } (\alpha_{\text{base}} + \overline{a}) \leq 1, \\ 1 & \text{otherwise} \end{cases} \tag{1}$$

$$\alpha_1 = a e^{b e^{ca}} \tag{2}$$

$$\alpha_2 = 1 - \|\hat{n} \cdot \hat{v}\| \tag{3}$$

$$\alpha_{\text{final}} = \begin{cases} \alpha_1 + \alpha_2 & \text{if } \alpha_1 + \alpha_2 \leq 1, \\ 1 & \text{otherwise} \end{cases} \tag{4}$$

In Eq. 1, we add the two transparency values and clamp the sum to 1. Eq. 2 is given by the *Gompertz function*, which incorporates the activity values. Parameters b and c define an x-axis displacement and a growth rate, respectively, but are not used to convey data values; we empirically set them to $b = -7$ and $c = -5$.

Additionally, the base transparency α_{base} in Eq. 1 can be adjusted interactively, thus enabling the user to inspect an area even when, according to activity data, the area would not be visible. Finally, both α-values are combined in Eq. 4. For visual depth cues and lighting, we use Phong illumination. In contrast to simple activity plots, this depiction places the activity data in a geometric context, which is important for the analysis of cross-area interactions.

In order to aid navigation, each brain area is annotated by its anatomical designator. Because the view onto the scenery can be changed by users, we interactively adapt annotation positions [16]. To show activity even in case of occluded areas, we additionally encode this information via the connecting line between annotation and area. Its width is modulated with the local activity and the color is chosen according to the activity color mapping.

One key goal for the visualization design is linking macroscopic data, e.g., the average firing rate per area, to fine-grained microscopic features, e.g., the spiking behavior of individual neurons. We address this issue by interactively displaying raster plots for selected brain areas in this view, a standard technique from the neuroscience tool set and easy to interpret for domain experts. In addition, this design decision is motivated by **R7**. By directly pointing at brain areas, the user selects an area and the corresponding spike trains are displayed in a raster plot (cf. Fig. 1). Another technique, which also addresses this requirement is the connectivity matrix as depicted in Fig. 1. It visualizes the density of connections between populations and is also brought up by selecting a brain area.

4.2 Population View

The *population view* provides access to connectivity and activity data within each brain area. Hence, it addresses multiple requirements. First, it facilitates the inspection and comparison of population-specific connection patterns influencing the dynamics of the network (cf. Fig. 4), according to **R3**. Second, it visualizes the activity of individual populations by displaying their mean firing rates using an animated bar. These bars are grouped into a single panel per brain area and relate back to **R4**. Finally, the panels for all areas are arranged in a hierarchical fashion specified by domain experts, e.g. Felleman and Van Essen [4]. This aims to facilitate the detection of activation patterns across areas following transient input and enables the discovery of regularities across the hierarchy as required by **R5** and **R6**.

The entire hierarchy layout effectively resembles a *small-multiples* design as shown in Fig. 5. Each panel has been carefully designed to integrate a number of data items in a readily accessible way. It consists of the following components. First, for every population within an area, the mean activity is displayed by a labeled bar. The bar coloring uniquely identifies populations across multiple panels in order to facilitate the detection of regularities across areas. Bars are stacked analogously to the actual anatomical configuration of the different layers which are illustrated in Fig. 2.

Second, each panel features a circuit diagram that depicts intra-area connection patterns between populations within a single brain area. This circuit diagram is arranged according to the vertically aligned bar chart layout of the activity data. This circuit is drawn to the left of the activity bar chart. Its arrows indicate the direction and strength of connections, the latter being scaled logarithmically. To identify directions more easily, we divide the connection graph

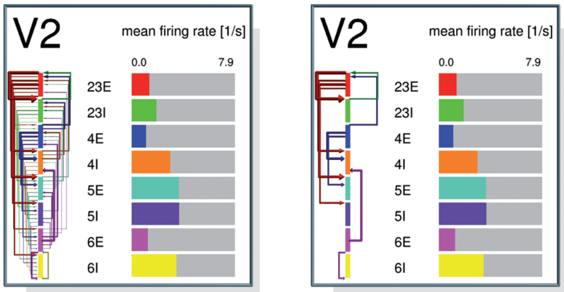

Figure 4: The *population view* shows detailed information about the eight populations in a single brain area. Left: Population connectivity without restricting connection strength. Right: The same connectivity with weak connections masked.

into downward connections on the left and upward connections on the right. Arrow colors correspond to the color of the source population in the bar chart. For example, in Fig. 4 the red arrows depict projections originating in layer 23E.

A first prototype of this design revealed that the connection display suffers from heavy clutter for any reasonable data. Therefore, we introduced an interactive brushing, i.e. users may limit the connections that are actually shown by a dynamic region query on the connection strength.

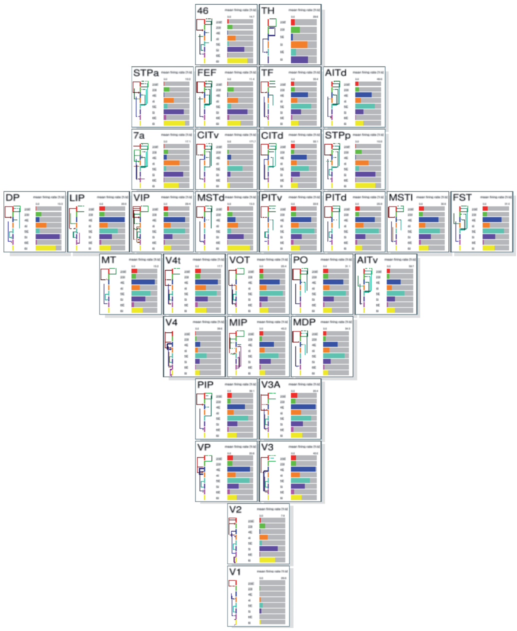

Figure 5: *Population view* of activity and connectivity where areas are arranged according to the visual cortical hierarchy proposed in [4].

4.3 Connectivity View

As outlined in Sec. 3.2, connectivity is a fundamental aspect of neuronal network models. To explore connectivity data of the underlying model at the area scale, as requested by **R3**, we provide the *connectivity view*. This view interactively shows how each brain area is connected to others and how it may spread activity along its pathways. The user can freely navigate around this view with 3D navigation techniques, e.g. a trackball metaphor or a SpaceMouse in a 2D setting or 6DOF head-tracking, where available. The user may select a specific area and all outgoing connections are drawn from this area. Connectivity data is represented by a weighted, directed graph. A brain area defines a node; a directed, weighted edge is introduced for each existing connection between any two areas.

Currently, the resulting graph is visualized in a relatively straightforward manner. By request of the domain experts, each node is represented by a sphere placed at the center of the respective area's bounding box. In this way, the design preserves the approximate geometrical relations and thus eases the transition between different views, e.g. from the *control view* to the *connectivity view*. To provide a means of relating activity to connectivity, the spheres are colored according to the area's current activity value with the same color mapping as in the *control view*. Edges are drawn as straight lines with arrows depicting direction, i.e. no edge bundling is performed. The color of the edges is linearly interpolated between white at the source and green at the target, while line thickness indicates connection strength. Similar to the *control view*, annotations depict the area's anatomical designator to aid users in navigating the data set.

To reduce information load and cluttering, the display of connectivity can be influenced by the user. Connections can interactively be masked based on *min*/*max* threshold values for the connection strength, which effectively implements a region query on the edge set. Fig. 3 (right) gives an example for connections originating in area STPp and shows the extended pie menu to control several aspects of the connectivity visualization. Furthermore, the user can turn off the drawing of connections by disabling individual areas.

Since the connection display used in the *population view* would suffer from clutter when applied to larger graphs, we decided not to use a circuit representation in a similar form for the visualization of inter-area connectivity. First, the node degree for every area in the *connectivity view* is higher than the one in the population case. We therefore assume that this approach would not lead to a very compelling representation of inter-area connectivity. Second, since the number of areas for this model is fixed and not too high, we decided to embed the representation in 3D space. Third, this embedding makes it possible to align the views for connectivity and flux thus preserving orientation in the data set.

4.4 Flux View

While the *connectivity view* presented in the previous section depicts the static connectivity between brain areas, the *flux view* is designed to visualize dynamic activity exchange between them. It thus addresses **R6**.

Interactions between areas can be studied in terms of numbers of spikes transmitted per unit time, a quantity we call *activity flux*. To translate activity flux into a visual representation, we distinguish the following. First, self-flux or activity exchange within an area, which is obtained from the stored graph of interacting brain areas by extracting all edges with equal origin and destination. Second, inter-area flux, i.e. flux between two distinct areas, which corresponds to all edges not considered to be self-flux. Both flux modalities are normalized to the unit interval by dividing either by *max* self-flux or *max* flux over the entire simulation. Flux is visualized in an animated graph display that follows the design of the *connectivity view*, as shown in Fig. 8 (center and right). Again, node positions are fixed in order to maintain context when switching between different views.

In contrast to the static *connectivity view*, sphere sizes and the edges are animated according to the time-dependent flux data in the following way. Self-flux is mapped to the radius of the spheres which represent brain areas. A default minimum radius is defined for spheres with zero flux ensuring general visibility. Non-zero self-flux is then mapped to a radius increase. To avoid discrete jumps of sphere size between time steps, we use linear interpolation for a smooth animation.

Inter-area flux between two areas A_i and A_j is indicated by an arrow in each direction. Each arrow's length is limited to $0.5 \cdot d(A_i, A_j)$ to avoid overlapping if flux is exchanged in both directions. The flux value at the current point in time is mapped to arrow thickness. Between time steps, we interpolate these values analogously to the self-flux mapping. Direction of flow is additionally indicated using a color coding similar to [19]. Similarly to the previously described views, the user may mask fluxes by defining thresholds and deselecting areas.

4.5 Interaction

As stated above, our system is based on VR technology. This affects not only the display itself, but also significantly changes the way users interact with the system.

Navigation in 3D space is implemented in several ways, depending on the actual system being used. For desktop setups, we implemented a mouse-based trackball metaphor and rotation and translation via a SpaceMouse device. In a head-tracked environment, users may naturally move around to browse the scene, which is perceived as a major benefit of using VR technology. Moreover, rotation is then mapped to a suitable input device, e.g. the coolie hat of a tracked 6DOF joystick.

Three-dimensional selection is performed via a picking ray, which is tested against the brain areas' axis-aligned bounding boxes; an area is highlighted whenever the ray intersects its bounding box and has the closest distance to the ray's origin. In this way, users can point at a specific brain area and mark it for further investigation. The selection operation is linked to additional displays, as outlined above, e.g. selecting a brain area will bring up the corresponding raster plot display.

For general system control, we decided to use extended pie menus [5]. Several visualization properties, e.g. color lookup tables and range queries, can be controlled by corresponding elements in these menus. In partial fulfillment of **R8**, using a single, consistent GUI technique for all supported systems ensures that interaction scales well from desktop- to fully immersive systems.

Time navigation in all views is implemented via video cassette recorder (VCR) controls. In addition, there is a time slider on which the user can click to directly jump to a desired point in time (cf. Fig. 1, bottom right). Time is linked across views, i.e. dragging the slider in one view will synchronize all other views to the given point in (data) time. This interface helps the user to quickly navigate in time, and complies with the requirement on controlling the simulation playback (cf. **R2**).

5 CASE STUDY

All views discussed above have been integrated in a prototype visualization application. We evaluated this system's effectiveness by asking domain experts for their feedback and suggestions; the findings discussed in this section directly reflect their input. Yet, before we demonstrate some of the findings generated with VisNEST so far, we first outline our collaborators' previous workflow.

To date, the typical analysis workflow in computational neuroscience uses static figures which display individual data modalities ranging from simple measures such as firing rates to sophisticated observables, e.g. information transfer. These figures are typically generated using a multitude of tools, e.g. scripts written in Matlab or Python. Using separate, static figures makes the analysis of large networks with many different subunits laborious, in particular when studying dynamic interactions between subnetworks. Notably, the separate depictions are not linked to each other in any meaningful way. Hence it is hard to relate information evident in one illustration to aspects shown in another. Moreover, turnaround times are further increased by the fact that the generation of each figure – e.g. for changed parameter settings – requires manual execution of a number of recurring steps. It is therefore helpful to easily display different data modalities simultaneously in an interactive visualization in order to reveal meaningful relationships.

The design of VisNEST was directly motivated by these shortcomings, and the following paragraphs reflect how domain scientists used VisNEST to analyze the model described in Sec. 3.1.

Analysis of parameter impact on large-scale activity — Not all simulation parameter settings yield realistic network behavior. This is partly due to the fact that available biological data are not sufficiently restrictive to completely specify the model, and to some extent due to simplifications inherent in the model. Both to obtain realistic dynamics and to gain an understanding of the underlying mechanisms, it is important to study the influence of parameter settings on the network dynamics. The basic influence of parameters can be assessed in the *control view*, which gives a rapid overview of the large-scale network dynamics including its stability and the distribution of firing rates across areas. Moreover, all views can be compared across runs simply by launching the visualization multiple times, or by calculating a difference signal offline. This immediate visual access makes the judgement of simulation output faster and more intuitive. An example is presented in Fig. 6, where external inputs and relative inhibitory synaptic strengths were changed between simulation runs, and found to only yield realistic activity for one of the settings.

Brain activity has a specific distribution not only across areas, but also across layers and populations [17, 20]. For instance, inhibitory populations usually have higher firing rates than excitatory ones. Fig. 7 shows how the *population view* helps to assess the impact of parameter settings on population-specific activity. Other types of unrealistic activity, such as pathologically high firing rates or strongly oscillatory behavior, can also be quickly identified using this view. For both examples, the previous workflow would have forced scientists to manually generate these images; even the rather straightforward mapping of activity data to area geometries would not have been a standard operation. Morover, the resulting stills would not have provided interactive access to the temporal evolution of the underlying data.

Assessment of detailed firing patterns — When modeling cortical networks, one usually aims to achieve a regime where single neurons emit spikes irregularly and groups of neurons fire asynchronously. To gain an overview of the detailed firing patterns of the areas, one can select areas in the *control view* and bring up the corresponding raster plot (cf. Fig. 1). An advantage with respect to the previous workflow is that the animated geometric representation enables the quick identification of areas based on their overall activity or co-activation with other areas. Additional detail can be accessed immediately, e.g. by bringing up the raster plot for an arbitrary brain area, again something that formerly would have required significant manual intervention. raster plots form a standard part of the neuroscientific toolkit and enable experts to directly assess an area's activity. The example in Fig. 1 reveals generally asynchronous irregular activity, with horizontal stripes reflecting spike bursts, and some vertical bands indicating a degree of synchrony.

Comparison of structural and functional relationships — A general aim of the visual cortex study is to understand the relationship between the cortical structure and its dynamics. VisNEST enables the comparison of structural connectivity and functional relationships through the *connectivity* and *flux views*. Comparing Fig. 8 (left) and Fig. 8 (center and right), one can see how the activity follows the structural connectivity since only the flux via the strongest connection from V1 to V2 is not masked by the applied threshold. This interactive comparison of structural and dynamical relationships between areas was previously not possible.

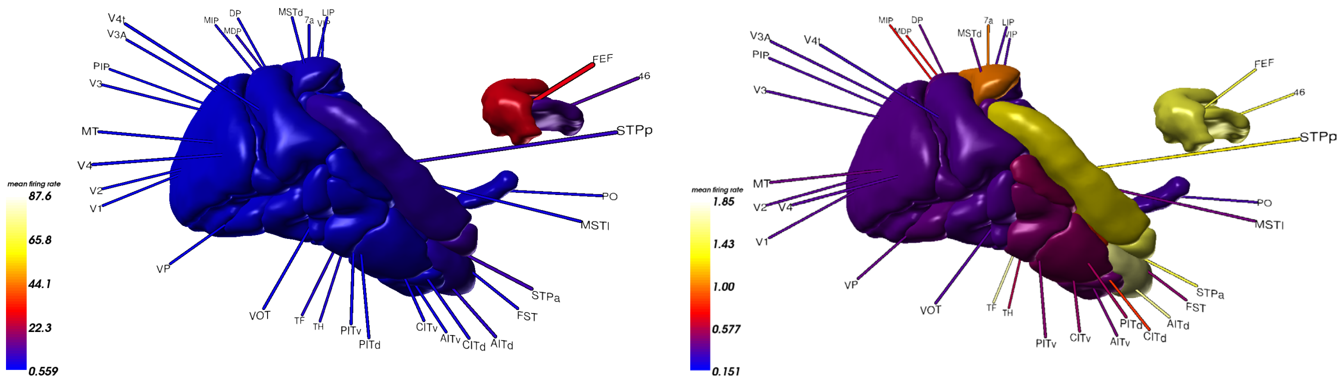

Figure 6: Comparison of two simulations using the *control view*. In the simulation on the left, a few areas are highly active, while the rest is nearly silent. On the right, a more realistic distribution of activity across areas is seen.

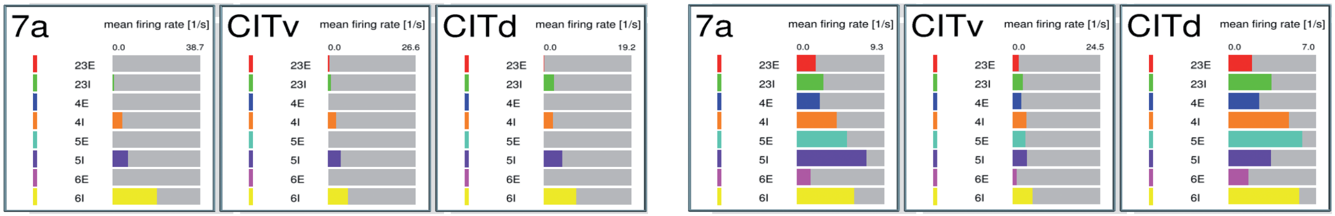

Figure 7: Comparison of two simulations using the *population view*. On the left, the excitatory populations are silent despite reasonable activity on the area level. On the right, both excitatory and inhibitory populations display activity, conforming more closely to the biology.

The functional interactions also depend on population-specific connection patterns, which are shown in the *population view* (cf. Fig. 4). It enables a direct comparison between such detailed connectivity and the population-level activity. Fig. 4 for instance shows fairly strong connectivity from layer 4 to layer 2/3, as also seen in the connectivity matrix in the control view (cf. Fig. 1). In the animated bar chart, this is reflected in the fact that layer 4 activation tends to be followed by an increase in layer 2/3 activity, an observation that was previously quite complicated to make. Moreover, the control of visualization speed enables neuroscientists to observe interactions on different time scales.

Pathways — The perception of visual stimuli relies on time-dependent activation of areas [12]. Studying the spread of activity upon transient external stimulation is therefore important for an understanding of visual cortical function. Area V1 is the main input station of the visual cortex, being activated by the thalamus upon visual stimulation. Fig. 8 (center and right) shows how the transient activation of area V1 yields a temporary increase in its outgoing flux, particularly to V2. Observing this activation in an animated visualization was previously impossible and is an important step for studying these processes, particularly in combination with the control of visualization speed.

The propagation of activity can also be followed in the *population view* (cf. Fig. 5). The resolution at the level of single populations allows the activities to be related to population-specific connection patterns which are included in the model. For instance, feed-forward connections – i.e. projections from early stages of visual processing, e.g. V1, to higher areas – preferentially originate in the upper layers and terminate in layer 4 [4]. Thus, the match between such known anatomy and patterns like upper layer activation in a lower-order area followed by layer 4 activation at the next hierarchical level can be investigated in this view.

6 CONCLUSION AND FUTURE WORK

In this paper, we have presented the VisNEST visualization application, which helps neuroscientists to interactively analyze neural activity data. A prototype of the application has been implemented and used by domain experts to simultaneously visualize 32 vision-related areas of the macaque brain.

The system has been designed in close collaboration with neuroscientists. It follows a multiple-view approach in order to integrate a variety of data modalities, e.g. geometrical information, raw simulation input, and derived, aggregate information. The system enables users to interactively browse brain areas, identify areas of interest and interactively analyze these in more detail. These capabilities directly address shortcomings in previous workflows as illustrated by our case study.

Initial user feedback on the utility of VisNEST is positive. Yet, the collaborative development process outlined in Sec. 1 strongly encourages continuous improvement based on user feedback. A key aspect that has recently been targeted by comments is the depiction of graph information. The current design is very much based on the initial requirement to leave node positions fixed. However, in order to avoid cluttering in both graph-based views, we would like to investigate the use of more sophisticated layout algorithms and edge bundling techniques. This might also positively affect the readability of the circuit displays in the *population view*.

Finally, we would like to note that the use of a VR-based application model – while discussed controversially in the beginning – is perceived as very advantageous in the meantime. Yet, aside from anecdotal evidence, we do not have any formal evaluation for this statement. Therefore, conducting formal user studies for individual views as well as their interactions will help us to better understand the benefits of VisNEST in particular and may eventually lead to insights regarding VR-based data visualization in general. Having the current version of VisNEST as an effective tool and being able to use it as a base-line is a first step in this direction.

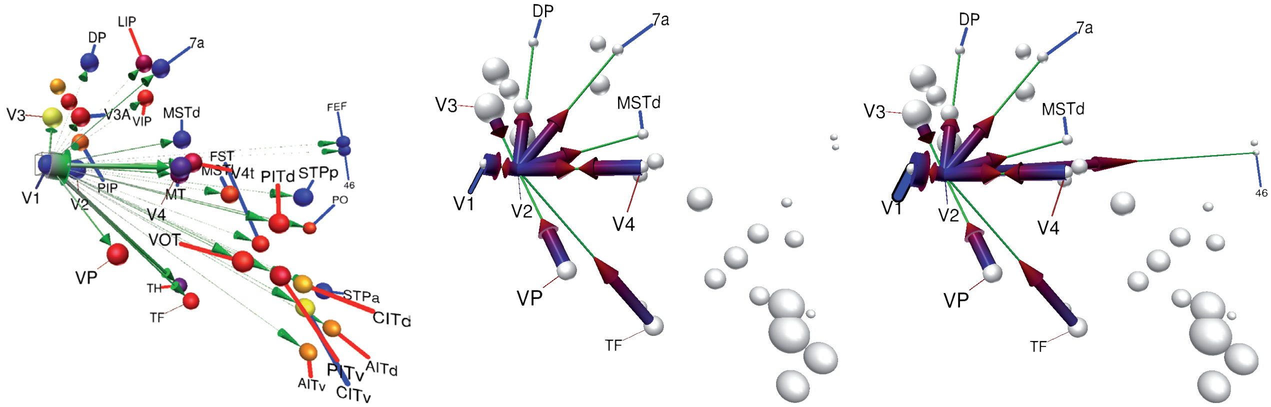

Figure 8: Left: Comparison of structural and functional connectivity for a simulation with transient input to V1. A strong structural connection exists from area V1 to V2. Center and Right: The *flux view* reveals a transient increase of the transmission from V1 to V2, in line with the anatomy. Fluxes are thresholded to enable focusing on the main pathways.

ACKNOWLEDGMENTS

We are grateful to Michael Denker and Sonja Grün for inspiring discussions.

This work has been funded by the Excellence Initiative of the German federal and state governments and the Jülich Aachen Research Alliance – High-Performance Computing. In addition it was partly supported by the Helmholtz Alliance on Systems Biology, the Next-Generation Supercomputer Project of MEXT, EU Grant 269921 (BrainScaleS), and the Helmholtz portfolio theme "Supercomputing and Modeling for the Human Brain".

REFERENCES

[1] B. Alper, B. Bach, N. Henry Riche, T. Isenberg, and J.-D. Fekete. Weighted Graph Comparison Techniques for Brain Connectivity Analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 483–492, 2013.

[2] R. Bakker, S. Larson, S. Strobelt, A. Hess, D. Wójcik, P. Majka, and R. Kötter. Scalable Brain Atlas: From Stereotaxic Coordinate to Delineated Brain Region. In *Frontiers in Neuroscience. Conference Abstract: Neuroinformatics 2010*, 2010.

[3] P. de Heras Ciechomski and R. Mange. Realtime Neocortical Column Visualization. In *BIOSIGNALS (2)*, pages 283–288, 2008.

[4] D. J. Felleman and D. C. Van Essen. Distributed Hierarchical Processing in the Primate Cerebral Cortex. *Cerebral Cortex*, 1(1):1–47, 1991.

[5] S. Gebhardt, S. Pick, F. Leithold, B. Hentschel, and T. Kuhlen. Extended Pie Menus for Immersive Virtual Environments. *IEEE Transaction on Visualization and Computer Graphics*, 19(4):644–51, 2013.

[6] M.-O. Gewaltig and M. Diesmann. NEST (NEural Simulation Tool). *Scholarpedia*, 2(4):1430, 2007.

[7] B. Hentschel, I. Tedjo, M. Probst, M. Wolter, M. Behr, C. Bischof, and T. Kuhlen. Interactive Blood Damage Analysis for Ventricular Assist Devices. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1515–1522, 2008.

[8] J. Hernando, F. Schürmann, and L. Pastor. Towards Real-Time Visualization of Detailed Neural Tissue Models: View Frustum Culling for Parallel Rendering. In *Proceedings of the IEEE Symposium on Biological Data Visualization (BioVis)*, pages 25–32, 2012.

[9] M. Hummel, C. Garth, B. Hamann, H. Hagen, and K. I. Joy. IRIS: Illustrative Rendering for Integral Surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1319–1328, 2010.

[10] A. Kasiński, J. Pawlowski, and F. Ponulak. 'SNN3DViewer' - 3D Visualization Tool for Spiking Neural Network Analysis. In *Proceedings of the International Conference on Computer Vision and Graphics: Revised Papers*, ICCVG 2008, pages 469–476, 2009.

[11] B. Laha, K. Sensharma, J. D. Schiffbauer, and D. A. Bowman. Effects of Immersion on Visual Analysis of Volume Data. *IEEE Transactions on Visualization and Computer Graphics*, 18(4):597–606, 2012.

[12] V. A. Lamme, H. Super, H. Spekreijse, et al. Feedforward, Horizontal, and Feedback Processing in the Visual Cortex. *Current opinion in neurobiology*, 8(4):529–535, 1998.

[13] S. Lasserre, J. Hernando, S. Hill, F. Schürmann, P. de Miguel Anasagasti, G. Jaoude, and H. Markram. A Neuron Membrane Mesh Representation for Visualization of Electrophysiological Simulations. *IEEE Transactions on Visualization and Computer Graphics*, 18(2):214–227, 2012.

[14] N. T. Markov, M. M. Ercsey-Ravasz, A. R. Ribeiro Gomes, C. Lamy, L. Magrou, J. Vezoli, P. Misery, A. Falchier, R. Quilodran, M. A. Gariel, J. Sallet, R. Gamanut, C. Huissoud, S. Clavagnier, P. Giroud, D. Sappey-Marinier, P. Barone, C. Dehay, Z. Toroczkai, K. Knoblauch, D. C. Van Essen, and H. Kennedy. A Weighted and Directed Interareal Connectivity Matrix for Macaque Cerebral Cortex. *Cerebral Cortex*, 2012.

[15] E. Nordlie and H. E. Plesser. Visualizing Neuronal Network Connectivity with Connectivity Pattern Tables. *Frontiers in Neuroinformatics*, 3(39), 2010.

[16] S. Pick, B. Hentschel, M. Wolter, I. Tedjo-Palczynski, and T. Kuhlen. Automated Positioning of Annotations in Immersive Virtual Environments. In *Proceedings of the Joint Virtual Reality Conference of EuroVR - EGVE - VEC*, pages 1–8, 2010.

[17] T. C. Potjans and M. Diesmann. The Cell-Type Specific Cortical Microcircuit: Relating Structure and Activity in a Full-Scale Spiking Network Model. *Cerebral Cortex*, 2012. Online first.

[18] O. Schmitt and P. Eipert. neuroVIISAS: Approaching Multiscale Simulation of the Rat Connectome. *Neuroinformatics*, 10:243–267, 2012.

[19] D. Selassie, B. Heller, and J. Heer. Divided Edge Bundling for Directional Network Data. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2354–2363, 2011.

[20] S. Shinomoto, H. Kim, T. Shimokawa, N. Matsuno, S. Funahashi, K. Shima, I. Fujita, H. Tamura, T. Doi, K. Kawano, N. Inaba, K. Fukushima, S. Kurkin, K. Kurata, M. Taira, K.-I. Tsutsui, H. Komatsu, T. Ogawa, K. Koida, J. Tanji, and K. Toyama. Relating Neuronal Firing Patterns to Functional Differentiation of Cerebral Cortex. *PLoS Computational Biology*, 5(7):1, 2009.

[21] K. Stephan, L. Kamper, A. Bozkurt, G. Burns, M. Young, and R. Kötter. Advanced Database Methodology for the Collation of Connectivity Data on the Macaque Brain (CoCoMac). *Philosophical Transactions of the Royal Society London, Series B*, 356:1159–1186, 2001.

[22] A. von Kapri, T. C. Potjans, T. Kuhlen, and M. Diesmann. A Virtual Reality Exploration Tool for Multi-Scale Data from Brain-Scale Simulations. *Frontiers in Neuroinformatics*, (51), 2011.