Visualizing Loss Landscapes with Adaptive Sampling and Gradient Free Slicing

Arjun Prakash * Kevin Wang† Randall Balestriero

Brown University

Abstract

We extend the state-of-the-art of neural network loss landscape visualization in three ways. First we integrate adaptive sampling which chooses points from the most interesting parts of the landscape. Next, introduce gradient free optimization to allow the user to specify a custom heuristic to chose slices. Finally, we allow visualization beyond 2D slices by using simplicial complexes. We quantitatively demonstrate our method on a wide range of tasks including computer vision and physics informed networks. We also conduct a user study to show that our additions are qualitatively helpful to deep learning practitioners.

Keywords: Neural networks, deep learning, loss landscape, topological analysis

1 Introduction

Despite the high-dimensionality and non-convexity of neural network loss landscapes, neural networks are still somehow able to find global minima in practice. However, the trainability of a neural network depends on several design decisions including architecture, optimizer, and loss functions. Usually, practitioners only have a handful of quantitative metrics to help. Being able to visualize the training procedure offers a much richer way to characterize learning processes and build intuition into these design decisions.

First introduced by Li et al. [2], loss landscape (**LL**) visualization is done on the network parameters θ (which can consist of millions or billions of parameters) and loss function L by selecting two arbitrary vectors δ and η. The landscape is then generated by plotting the following function:

$$f(\alpha,\beta) = L(\theta + \alpha\delta + \beta\eta) \tag{1}$$

This "slice" is a 2D scalar field, which we can plot as a 2D contour plot or a surface in 3D, which can be inspected for properties like smoothness and local minima.

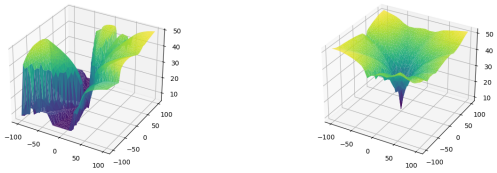
1.1 Contributions

We extend the work by [2] in three major ways.

1. In previous work, α, β are sampled at regular intervals. We integrate adaptive sampling which samples the most interesting parts of the parameter space. Our *Vanilla* implementation takes advantage of vectorization which leads to meaningful performance speedups.

2. In previous work, δ and η are chosen randomly. We instead allow the user to pick what kind of "slice" to visualize: They specify a function of the loss landscape (e.g. how curvy it is), and we try to find the slice that maximizes the function.

3. We use simplical complexes to visualize beyond 3D.



(a) A small network that generalizes poorly. Many local minima

(b) A larger regularized network that generalizes well. Smooth with a single minimum

Figure 1: Two neural network loss landscapes on the same task.

2 Implementation and Experiments

2.1 Adaptive Sampling

We use the adaptive algorithm from [3] to iteratively sample points of greatest interest. We evaluate our implementation on MNIST dataset on a 3 layer neural network. We compare three different implementations: the baseline adapted from [1], the adaptive implementation, and an efficient vectorized implementation of uniform sampling which we call *vanilla*.

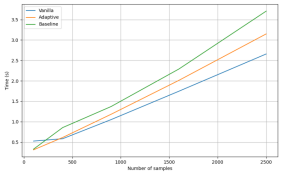


Figure 2: We show that Adaptive outperforms our baseline [1], but there is a small amount of overhead in determining the next best points to sample, which leads to a speed disadvantage compared to vanilla.

2.2 Custom Slices

We use the existing black-box optimization library Nevergrad [4] to maximize the given function f. We also compared the naive method of sampling N random slices, and returning the one with the highest value of f.

To evaluate curviness, in our experiments we defined f as an approximation of the sum of the second-order gradients of the LL. That is, for every point, we add the absolute values of the approximate second-derivative in both the x and y direction, and sum the values for all points.

In Figure 3, we compared the effectiveness of our methods and the baseline. As expected, both the Nevergrad method and the naive method (each with a budget of 100) outperformed the baseline (a

*e-mail: arjun_prakash@brown.edu

†e-mail: kevin_a_wang@brown.edu

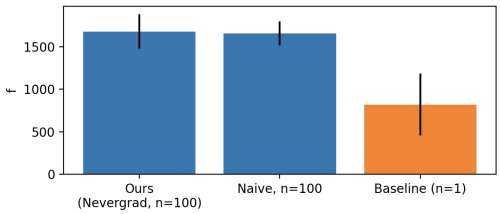


Figure 3: Custom Slices evaluation. Error bars show standard deviation.

single random slice). Unexpectedly, we were not able to get the Nevergrad method to perform much better than the naive method.

2.3 Mapper

(a) Many separated chains indicates many local minima

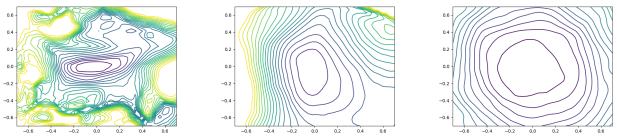
(b) A single long chain indicates smoothness and a global minimum.

Figure 4: Topological representations of 3D slices of the same networks as in Figure 1.

The Mapper algorithm [5] is a topological data analysis technique that simplifies high-dimensional data sets by creating a combinatorial representation that reflects the data's topological and geometric structure.

2.4 User Study: Evaluation of Custom 2D Slices

In the Task-Based Evaluation, users were presented with pairs of loss landscapes (LLs). Each LL was produced using a convolutional neural network (CNN). Each LL came from a CNN from one of three classes: Class **A** has *no* skip connections, **B** has *some* skip connections, and **C** has *full* skip connections. Figure 5 shows examples of the three classes. Some pairs were produced using the baseline method (random 2D slices) and some pairs were produced using our method (custom 2D slices). Our method used Nevergrad with a budget of 100 to maximize f.



(a) No skip connections (b) Some skip connections (c) Full skip connections

Figure 5: Representative examples of LLs from the three types of neural networks used in our task-based evaluation. LLs tend to be smoother in networks with more skip connections. More examples can be seen in Figure 7 in the appendix.

Users picked whether the LL on the left or the right came from the class with more skip connections (i.e. looked smoother). For each pair, users selected 1 of 5 options: Definitely left (1), probably left (2), unsure (3), probably right (4), definitely right (5).

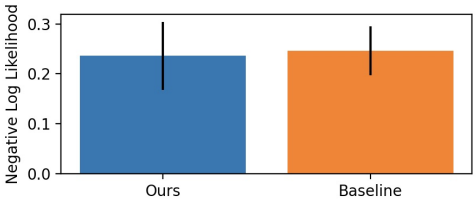


Figure 6: Comparison of methods with our scalar value of performance. Lower is better. Error bars show 95% confidence interval on the mean.

We hypothesized that on pairs generated using our method, users would choose the correct option more often and users would be more confident in their correct answers.

As a scalar measure of performance, we assigned probabilities to each option[1], and computed the average negative log-likelihood of choosing the correct answer. As seen in Figure 6, the results are too inconclusive to validate our hypothesis.

Our method was significantly better for correctly differentiating between class **A** and **B** LLs, but was worse for differentiating between **B** and **C** LLs. Detailed results can be seen in Table 2 in the Appendix.

2.5 User Study: PINNs, Mapper, and Adaptive Sampling

In the second user study, we surveyed 10 deep learning practitioners. Users were presented with LLs generated from a pair of physics-informed neural networks which were tasked with learning the temperature of a cup of coffee. Both networks had three layers, but one had 20 nodes per layer while the other had 2. The first was able to generalize while the second was not. The first question determined if the users could pick the neural network architecture that performed best on the test set given the LL on the training set. The second question was the same, but using the Mapper visualization. The final question determined if users preferred the adaptive sample over the uniform grid. In all cases users were given a brief preamble on how the LL was generated and interesting features (smoothness and minima).

Question	Percent correct	Confidence (1-5)
1	100	3.9
2	100	3.65

Table 1: User Study 2

For the final question, 5/10 users preferred uniform sampling. While, this means our results on the usefulness of adaptive sampling are unclear; it may be the case that adaptive sampling with a lower time budget may be as effective (if not more) as uniform sampling.

3 Conclusion

We extend loss landscape visualization by integrating adaptive sampling, gradient free slice selection and mapper complexes. We conduct quantitative evaluations to show that our implementation increases efficiency. Our user studies show that deep learning practitioners are able to draw insights about the performance of networks on the test set by visually inspecting the loss landscape on the training set.

[1] 10%, 30%, 50%, 70%, and 90%. Users were not given these numbers

REFERENCES

[1] A. Chatzimichailidis, J. Keuper, F.-J. Pfreundt, and N. R. Gauger. Gradvis: Visualization and second order analysis of optimization surfaces during the training of deep neural networks. In *2019 IEEE/ACM Workshop on Machine Learning in High Performance Computing Environments (MLHPC)*, pages 66–74. IEEE, 2019.

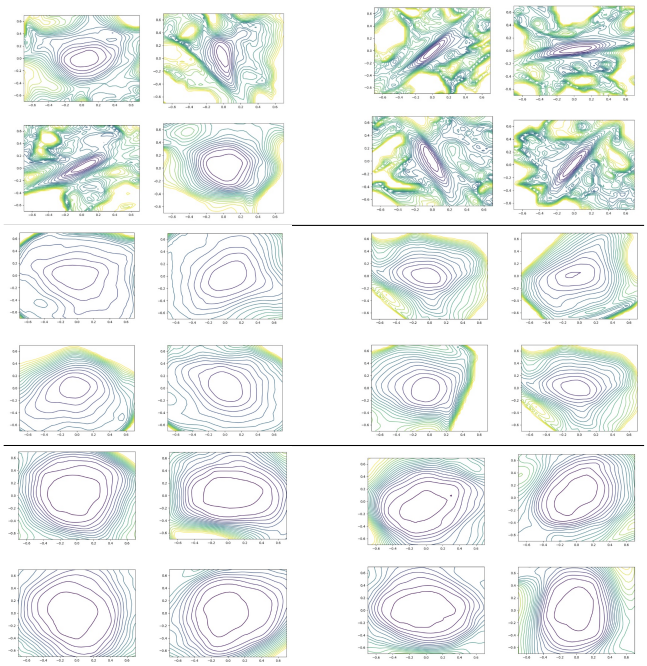
[2] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein. Visualizing the loss landscape of neural nets. *Advances in neural information processing systems*, 31, 2018.

[3] B. Nijholt, J. Weston, J. Hoofwijk, and A. Akhmerov. *Adaptive*: parallel active learning of mathematical functions, 2019.

[4] J. Rapin and O. Teytaud. Nevergrad - A gradient-free optimization platform. https://GitHub.com/FacebookResearch/Nevergrad, 2018.

[5] G. Singh, F. Mémoli, G. E. Carlsson, et al. Topological methods for the analysis of high dimensional data sets and 3d object recognition. *PBG@ Eurographics*, 2:091–100, 2007.

A DETAILED RESULTS



(a) Baseline (Random Slice) (b) Ours (Custom Slices)

Figure 7: Examples of loss landscapes presented to users during the Evaluation of Custom 2D Slices user study (Section 2.4). Landscapes in the left 2 columns were generated by the baseline method (random slice). Landscapes in the right 2 columns were generated by our method (maximum curviness via Nevergrad). The top two rows are from class A, the middle two are class B, and the bottom are class C. Notably, for class A: all landscapes using our method are easily identifiable, while some baseline (random slice) landscapes are less obvious.

Pair	Method	1	2	3	**4**	**5**
AB	Baseline	0	0	7%	37%	55%
AB	**Ours**	0	0	0	16%	83%
BC	**Baseline**	0	7%	11%	40%	40%
BC	Ours	0	11%	22%	27%	38%
AC	Baseline	0	0	0	14%	85%
AC	**Ours**	0	0	0	11%	88%

Table 2: Full results of task-based evaluation. Pair AB indicates results when one LL is from class **A** and one is from **B**. Column 1 indicates the percentage of users who were confident in the wrong answer: they chose "Definitely right" when the correct answer was left, or vice versa. Column 2 is "probably" the wrong answer, etc.

B ARJUN'S CONTRIBUTION

Arjun's core contributions were focused on adaptive sampling and going beyond 2d slices.

B.1 Intellectual Contribution

Arjun was responsible for the original project idea and for organizing the collaboration with Randall Balestriero. Other intellectual contributions were finding the run-time baselines. Arjun also researched topological analysis to figure out how to vizualize beyond 3D. Finally, he decided to use PINNs and designed the second user study.

B.2 Practical Contributions

Arjun implemented adaptive sampling and the vectorized vanilla baseline. Arjun also intergrated the Mapper algorithm and the functionality to generate gifs. Finally, Arjun ran the experiments on PINNs and sourced users for the study.

C KEVIN'S CONTRIBUTION

Kevin was responsible for the bulk of the custom slices research contribution.

C.1 Intellectual Contribution

This entailed scoping out exactly what the custom slices research contribution would be, and how to design evaluations for it. This involved talking with the collaborator during our meetings to understand how custom slices could help researchers, how it would help in his own experience, and how he envisioned it to be used. It also involved figuring out a good function to maximize, by talking with our collaborator and looking around at related works.

C.2 Practical Contribution

Practically, this involved designing, writing, and testing the code to perform the custom slice optimization (both the Nevergrad version and the naive, best-of-K version), and designing and testing various functions to maximize. This also involved designing, creating, and running the evaluations for custom slices.

Using Syntax-Based Context Visualizations To Understand Features of Language Models

Eric Xia*, Byron Butaney*, and Gonçalo Paulo**

*Department of Computer Science, Brown University
**EleutherAI

ABSTRACT

By identifying monosemantic features from model weights, Sparse Autoencoders (SAEs) allow for a more complete understanding of how neural language models function. This work introduces two novel methods for unifying SAE feature contexts, one based on syntax trees and one based on linear aggregation. Users found syntactic visualizations promising but confusing; initial survey results demonstrate that our linear aggregation method performed worse than the baseline. The results demonstrate the challenges of (1) employing syntactic methods for feature analysis and (2) facilitating textual comprehension through visualization.

Keywords: Human Computer Interaction, Interpretability, Dependency Parsing, Natural Language Processing, Large Language Models

1 INTRODUCTION

Large language models are becoming increasingly integrated in daily life, but their underlying mechanisms are not fully understood. Recently, sparse autoencoders (SAEs) have emerged as a promising way to extract features from models. [2] These features activate on input contexts in predictable ways, with some exhibiting consistent syntactic patterns. Improved understanding of features through their contexts can facilitate comparisons between features, identify training issues such as over-splitting, and simplify identification of highly syntactic features.

Consequently, this work investigates unified context visualizations for individual SAE features. One critical issue with current feature dashboards is their use of a list of text contexts to characterize a features. Although this may aid in identifying repetition over contexts, characterizing features with textual contexts fail to highlight the linguistic abstractions which features represent.

2 RELATED WORK

Current research in the field of mechanistic interpretability shows that SAEs can successfully train on larger and more capable models, such as Gemma Scope [6] and Claude 3.5 Sonnet [11], providing promising opportunities to advance interpretability. However, achieving feature monosemanticity only serves as a first step in interpretability [2]. Crucial to utilizing features in interpretability is a way to understand the role they serve within language models. For an SAE, features are defined as weighted combinations of neurons from specific layers of the model. Still, many methods have been proposed for extracting features, such as transcoders, [3], crosscoders [10], and Meta SAEs [1]. Feature characterization is critical for any interpretability method which relies on regularly activating patterns in text. Research in the field, as conducted in this work, has long-term implications for interpretability.

There are many specific applications which would benefit from improved feature identification. One common goal within interpretability is to identify universal features across models. These notions of universality require features identified for one model to be compared to others. Through neuron-level comparisons of output contexts, universal activations have been identified across GPT-2 models on punctuation, dates, and medical terms [4]. We build on prior, text-based methods through the creation of merged visualizations, allowing researchers to compare emergent semantic features across models.

Other papers in interpretability that utilize SAE features do so by identifying groups of features that work together. For example, feature comparisons have led to the identification of features where occlusion and over-splitting occur [7]. By unifying text contexts, our visualization aims to provide an increased understanding of structure among features, aiding in identification of occlusion and over-splitting.

Ultimately, understanding both the scope and context of feature activations will be necessary to characterize the performance of interpretability techniques. Current text views fail to identify or aggregate shared contexts and scope. Our work explores methods for characterizing features through visual aggregation.

3 METHODOLOGY

Our context visualizations are implemented with SpaCy, Huggingface, and Plotly. We additionally use Uniform Manifold Approximation and Projection (UMAP) and the Transformer Lens libraries to visualize decoder features through dimensionality reduction. We primarily used data provided by our collaborator, consisting of intermediate-layer feature activations of a JumpReLU SAE, a current state-of-the-art architecture [9]. These activations were on tokens from Google's Gemma-9b model on a eWeb-100m dataset [8].

Two focus areas were identified. The first focus was to aggregate part of speech tags for each feature to improve characterization of the feature space. The second focus was to unify contexts through syntactic methods.

3.1 Feature Navigation

Our tool presents users with a generated UMAP upon initiating the tool. Viewing the feature space in this way allows users to navigate the set of features based on the UMAP clusters, which may be syntactically significant. This provides another layer of information to users that could be used to discriminate between features of potential interest. Users can also highlight a region of the UMAP with their cursor to zoom into the area, allowing for more precise choices between features on the plot. After selecting a feature from the UMAP, users are able to view the contexts upon which the feature activates in multiple views. Having access to this more fine-grained information about the features allows users to determine its relevance and thus further navigate the dataset.

3.2 Feature Views

There are two primary views developed for the study. For each feature, the top 50 activations in their surrounding context sentences are displayed. Our syntactic trees are created using SpaCy's dependency parser [1] [2].

Joint. We present a joint view displaying the dependency trees for each sentence in parallel (see Figure 1). The visualization defaults to omitting inactive tokens, although they may be re-enabled through a checkbox at the top.

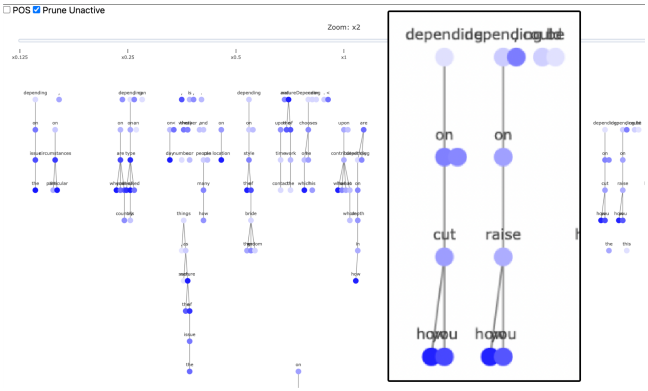


Figure 1: The joint view for this Gemma-9b Layer 11 feature visually demonstrates feature activations over *depending* and *on*, followed by a noun phrase. The following phrase (e.g. "how you") typically receives higher activations than the shared tokens, which is not obvious from text contexts.

Merged. We also introduce a second view to display merged sentences (see Figure 2). If a token sequence matched an existing branch, it is subsumed into the branch and visually emphasized. This view does not incorporate syntactic information. We pivoted to this view after receiving feedback that other syntactic structures were hard to understand.

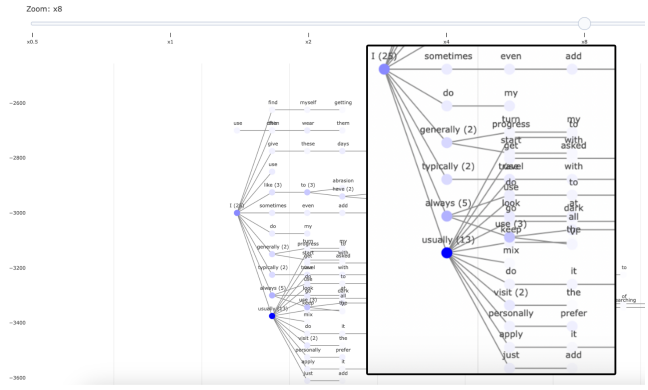


Figure 2: The initial token activations *I* and *usually* are shared across many contexts. The merged view for Gemma-9b Layer 11 feature displays a single tree representation of the shared tokens, and notes recurring sequences in the text ("I usually use").

4 Research Findings

We were able to identify several features through our visualization that could be easily identified based on shared activations. These features are not easily visualized using previously existing tools and primarily consisted of phrases with shared following contexts. **Depending on [P]** and **I usually [VP]** are two examples shown above; emphatic **do** phrases, **be clear**, and **too [adjP]** are other examples identified.

[1] See [5] for details on the dependency parser used.

[2] The Gemma-9b features are available at the following dataset

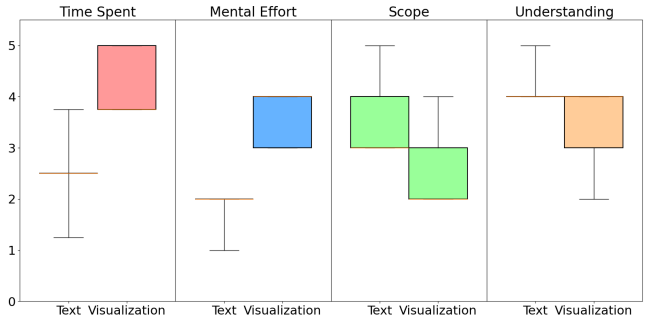


Figure 3: For the users in our online study (n=5), the merged view achieved comparable understanding of the feature and scope, but expended significantly more time and effort.

5 User Studies

We conducted two user studies involving an expert researcher and researchers from online interpretability communities.

We interviewed a researcher who specialized in SAE interpretability. They mentioned that existing metrics for dealing with high-dimensional space often did not characterize features well. They identified the merged view as the most promising and indicated that features in which contexts had longer syntactic regularities would be crucial in a proof-of-concept for a syntactic visualization.

The second study performed a head-to-head comparison of the text baseline and merged visualization. Users were shown a feature through both methods, and asked to describe what the pattern was across contexts. Users found the merged view less informative and intuitive than text contexts. Though these negative results could be partially attributed to overlapping text in the merged view, they additionally point towards the challenge of improving textual comprehension through visualization techniques.

6 Discussion

In this work, we have developed several methods for utilizing syntactic information inherent to a feature's activating contexts. We have found that UMAP plots of the feature space cluster based on part of speech distribution, and identified certain features which are well-characterized by the visualizations developed. The results of the user study then raise the question: **how might feature with highly regular syntactic activations be systematically determined?** Syntax-based feature identification would enable researchers to view a relevant subset of features, allowing them to compare and group them more easily.

Our user study tested one prototype and did not evaluate any syntactic views between feature contexts. Unfortunately, the primary negative feedback we received was about text overlapping and display, which precluded observations on the significance of the visualization overall. As seen in Figure 1, syntactic information is able to disambiguate regular from irregular feature activations. Further research might focus on how best to present syntactic regularities for visualization and comparison.

Acknowledgements

The authors wish to thank Professor David Laidlaw and their collaborator, Gonçalo Paulo, for their guidance and support throughout this project. The authors also thank their classmates for their feedback throughout the various stages of the research process.

REFERENCES

[1] P. L. J. I. B. L. S. N. N. Bart Bussmann, Michael Pearce. Showing sae latents are not atomic using meta-saes, 2024. Accessed: 2024-12-08.

[2] T. Bricken, A. Templeton, J. Batson, B. Chen, A. Jermyn, T. Conerly, N. Turner, C. Anil, C. Denison, A. Askell, R. Lasenby, Y. Wu, S. Kravec, N. Schiefer, T. Maxwell, N. Joseph, Z. Hatfield-Dodds, A. Tamkin, K. Nguyen, B. McLean, J. E. Burke, T. Hume, S. Carter, T. Henighan, and C. Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features/index.html.

[3] J. Dunefsky, P. Chlenski, and N. Nanda. Transcoders find interpretable llm feature circuits. *arXiv preprint arXiv:2406.11944*, 2024.

[4] W. Gurnee, T. Horsley, Z. C. Guo, T. R. Kheirkhah, Q. Sun, W. Hathaway, N. Nanda, and D. Bertsimas. Universal neurons in gpt2 language models. *arXiv preprint arXiv:2401.12181*, 2024.

[5] M. Honnibal and M. Johnson. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1373–1378, 2015.

[6] T. Lieberum, S. Rajamanoharan, A. Conmy, L. Smith, N. Sonnerat, V. Varma, J. Kramár, A. Dragan, R. Shah, and N. Nanda. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2. *arXiv preprint arXiv:2408.05147*, 2024.

[7] A. Makelov, G. Lange, and N. Nanda. Towards principled evaluations of sparse autoencoders for interpretability and control. *arXiv preprint arXiv:2405.08366*, 2024.

[8] G. Penedo, H. Kydlíček, A. Lozhkov, M. Mitchell, C. Raffel, L. Von Werra, T. Wolf, et al. The fineweb datasets: Decanting the web for the finest text data at scale. *arXiv preprint arXiv:2406.17557*, 2024.

[9] S. Rajamanoharan, T. Lieberum, N. Sonnerat, A. Conmy, V. Varma, J. Kramár, and N. Nanda. Jumping ahead: Improving reconstruction fidelity with jumprelu sparse autoencoders. *arXiv preprint arXiv:2407.14435*, 2024.

[10] T. C. Team. Crosscoders: A transformer circuits analysis, 2024. Accessed: 2024-12-08.

[11] A. Templeton, T. Conerly, J. Marcus, J. Lindsey, T. Bricken, B. Chen, A. Pearce, C. Citro, E. Ameisen, A. Jones, H. Cunningham, N. L. Turner, C. McDougall, M. MacDiarmid, C. D. Freeman, T. R. Sumers, E. Rees, J. Batson, A. Jermyn, S. Carter, C. Olah, and T. Henighan. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*, 2024.

Eric Xia - Individual Contributions

Intellectual Contributions

As the primary investigator for the project, I drove ideation and prototyping of the visualization. I initiated contact with the collaborator, and wrote the initial proposal. I identified research questions and the initial motivations for the project. I summarized survey feedback and revised the user study according to the comments of classmates. I provided feedback and guidance to Byron on working with part of speech statistics. I was the primary contributor to many sections of the final report. Finally, I identified future research directions and open questions.

I also contributed to each of the weekly presentations and survey development. These included the initial in-class user study, the revised interpretability researcher study, the week 2 presentation examples, the week 3 presentation examples, the week 4 slides, the week 5 joint examples and study reflection, the week 6 results, and the final presentation.

Lastly, I was responsible for identifying and setting up meetings with interested parties, including extended conversations with Neuronpedia developers, and the user interview with Curt Tigges, Head of Science at Decode Research.

Technical Contributions.

I wrote the majority of the syntactic processing and token alignment code, which is present in the 'graphs' module referenced in the final code repository. These included the following technical contributions: Converting the raw token and location data into feature-specific dictionaries. Using the SpaCy dependency parser to convert tokens into tagged words, and adding token activation as an attribute. Using the SpaCy sentence tagger to extract relevant context. Writing tree merge functions, including identifying common nodes by lemma, counting total activations for each tree, and a subtree matching algorithm. Caching feature parses, contexts and activations to database to enable fast and frictionless retrieval. Finally, I implemented various text processing utilities, which converted between batch, sentence, and character indices.

I also created three main individual feature views with Plotly, Javascript, and the HTML/Jinja2 templating language, which are present in the 'templates' and 'static' folders in the final code repository. These front-end graph views started from LLM-generated base visualizations and were heavily modified. The views were served with Flask on a DigitalOcean Droplet virtual machine, and served as the focus for the user study. My technical contributions to the visualization included the following: introducing color gradients for activation values. Modifying a recursive node and edge display algorithm to prevent overlapping nodes. Adding the option to hide inactive nodes. Creating a custom text tag which updated dynamically with the node Part-of-Speech or text. Adding zoom functionality to the joint and merged views. Finally, I implemented the UMAP scatter navigation used in the class study.

Byron Butaney - Individual Contributions

Intellectual Contributions

My intellectual and practical contributions consisted of various milestones that changed throughout the course of the project. My intellectual contributions included meeting and ideating with Eric and our collaborator on a weekly basis, coming up with ways to create a UMAP that would provide more/different information compared to the Neuronpedia UMAP, and working with Eric to devise open research questions generated from our projects. I also helped devise, structure, and deploy the student and expert user studies. Finally, Eric and I met with Curt, the Neuronpedia developer, to demonstrate our tool, gain more insight into the problem space, and revise our user survey to be more suited to online expert users.

Technical Contributions

One of my overarching contributions was to identify any highly syntactic activating contexts for features. This included finding features that had significant connections between contexts and computing the part-of-speech distribution for every activating context of every feature. Another overarching goal was to apply dimensionality reduction techniques to the features in order to highlight any clustering by both their part of speech distribution as well as their most dominant part of speech. To visualize features by most dominant part of speech, I first computed the most-dominant part of speech for each feature based on the most frequently occurring POS context in that feature's list of contexts. I then labeled each feature and applied a UMAP with these labels included. To generate a visualization by part of speech distribution, I generated a list where each feature was given a row of 11 values (one for each possible POS). I then filled in these rows with the percentages that each POS appeared in the feature's activating contexts. UMAP-ing this data allowed us to see more clear clusters. This UMAP served as the basis for the UMAP used in our online tool. Since the data processing took such a long time, I ran the preprocessing locally on my computer over a couple of nights and saved the results as .npy files to speed up our visualization tool. I worked with Eric to design and conduct the three user studies, to create the final presentation, and to write the final report.

Exploration of UMAP Stability and Variability for Genomic SNP Data

Jasmine Liu* PI
Musa Tahir† Co-PI
Alex Diaz-Papkovich‡ Collaborator
Sohini Ramachandran§ Collaborator
David Laidlaw¶ Collaborator

Brown University

ABSTRACT

In this study, we explore the effects of genotype variability and single-nucleotide polymorphisms (SNP) subsampling on the performance and stability of uniform manifold approximation and projection (UMAP). UMAP is a dimensionality reduction technique increasingly used in population genetics due to its computational efficiency and ability to preserve local and, to an extent, global structure. SNP selection and UMAP visualization artifacts may obscure hidden genetic relationships and overemphasize differences between clusters, which motivates the need to promote a clearer understanding of these visualizations in the context of population genomics. Using data from the 1000 Genomes Project consisting of 3,450 individuals and more than 45,000 SNPs, our research revealed the following key insights: 1. Cluster morphology stability: cluster shapes are relatively stable across different subsamples; 2. Point Variability: the position of points within clusters, on the other hand, is highly variable; 3. Admixture challenges: cluster assignments for admixed populations may not necessarily be consistent. Such insights provide researchers with a deeper understanding of UMAP's behavior, ultimately supporting better-informed decisions in genomics research visualization. Furthermore, we identify several open research questions regarding the challenges of handling admixed populations and the need for deeper analysis of UMAP's sensitivity to SNP subsampling.

Keywords: UMAP, SNP subsampling, Genotype variability, Population Genomics.

1 INTRODUCTION

Genotype matrices capture SNPs across individuals, encoding variations of a single nucleotide at specific positions in the DNA sequence. Dimensionality reduction techniques like principal component analysis (PCA), t-distributed stochastic neighbor embedding (t-SNE), and UMAP are widely used to visualize genotype data [1] [2]. UMAP is increasingly used in genetic data analysis as it can process large datasets with complex relationships efficiently.

Diaz-Papkovich et al. demonstrated the ability of UMAP to uncover cryptic population structures and fine-scale relationships between genetic variation, geography, and phenotypes in data sets such as the 1000 Genomes Project and the UK Biobank [2]. However, while effective, UMAP's findings can sometimes result in the overinterpretation of inter-cluster and intra-cluster boundaries, reinforcing biologically deterministic conclusions and, in turn, harmful narratives [4]. In reality, the boundaries between geographical groups are often fluid and influenced by many factors [3].

*jasmine_c_liu@brown.edu
†musa_tahir@brown.edu
‡alex_diaz-papkovich@brown.edu
§sohini_ramachandran@brown.edu
¶david_laidlaw@brown.edu

Large-scale genomic studies often rely on subsampled SNPs for computational efficiency, but this sampling can introduce noise and impact clustering results, sometimes leading to misleading interpretations [6] [7]. By analyzing the effect of SNP subsampling, cluster variability, and admixture populations, this research gives researchers more insight into UMAP's behavior and equips them with the tools to produce more scientifically robust visualizations.

2 RELATED WORK

While UMAP has been implemented successfully on genotype data to reveal subtle population structures, there is limited guidance on how UMAP handles variability introduced by factors such as SNP subsampling and admixture populations [2]. This gap in UMAP understanding can make a reliable interpretation of visualizations challenging, particularly when analyzing complex genetic relationships or mixed populations.

Previous research has highlighted the impact of SNP subsampling in genetic data and population structure analysis. Pook et al. investigated imputation strategies, computational techniques to infer missing genetic information in a dataset, to mitigate SNP ascertainment bias [7]. Malomane et al. explored how SNP panel selection biases genetic diversity studies [6]. Both studies highlight the need for a more robust analysis when subsampling SNPs. This study builds on their results by exploring SNP subsampling in the context of UMAP visualizations and genotype variability. Unlike previous studies, we specifically addressed cluster morphology stability, intracluster point stability, and admixture challenges.

3 METHODOLOGY

This study utilized data from the 1000 Genomes Project, which includes more than 45,000 SNPs across 3,450 individuals [1]. Our implementation workflow included data preprocessing, dimensionality reduction using UMAP, and visualization. The preprocessing entailed using several filters to reduce bias and ensure quality. Filtering steps included removing variants with high correlation using PLINK, excluding the highly variable HLA region, only retaining variants with at least 5% frequency, and removing variants with excessive missing data.

To explore UMAP behavior and robustness, subsampling was employed by randomly selecting percentages of SNPs (e.g., 10%, 50%, 75%) to test the stability and variability of the cluster. For each of our visualizations, Procrustes alignment and the UMAP axis scaling were implemented to align the UMAP embeddings. Procrustes alignment is a mathematical technique that is widely applied in genomics to align two sets of points for meaningful comparison. In this context, Procrustes alignment helped align embeddings for an accurate comparison by ensuring consistent positioning across samples using scaling, translation, rotation, and normalization. For further analysis of cluster behavior, we visualized the projected and Procrustes aligned data using side-by-side comparisons, animated plots, stacked plots, and density contour overlays.

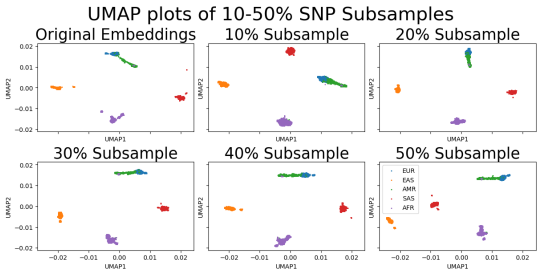
4 RESULTS

Based on our exploration of UMAP's performance and behavior, we observed three crucial results that reveal some of its strengths

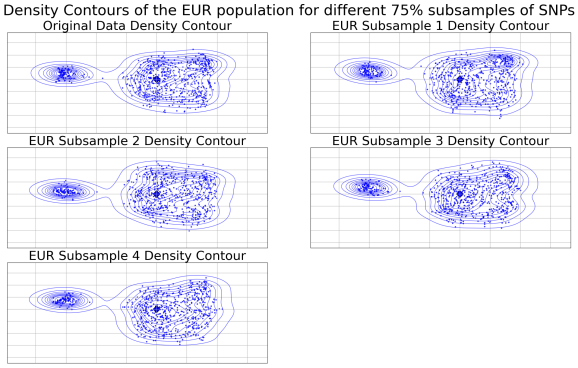
and weaknesses in the context of genetic data visualization.

Cluster Morphology is Stable

As seen from Figure 1, the cluster shapes remained relatively stable across the subsampling levels of 10%, 20%, 30%, 40%, and 50% relative to the original embedding. The density contour overlays highlight the concentration of points within clusters, indicating consistent morphological behavior as the general structure of the cluster is preserved for each subsample.



(a) Side-by-side UMAP plots using the original data (all SNPs) and using incremental subsamples of SNPs. Cluster morphology appears to be stable across 10%-50% subsamples.



(b) Side-by-side plots of the European population across the original data and different 75% subsamples of SNPs with density contour overlays. The density contour shapes appear to be fairly consistent across subsamples.

Figure 1: Cluster morphology visualizations.

Intracluster Point Variability is Unstable

Figure 2 reveals significant variability in point positions within the clusters. While the general regions of high density persisted, individual point placement within clusters was highly variable due to UMAP's inherent stochasticity. This suggests that UMAP reliably groups individuals into the same cluster shapes and regions, with their exact positions within the cluster shifting between iterations. The distribution of points corresponding to individuals can be quantified with the density contour overlays seen in Figure 2.

Cluster Assignment may be Inconsistent

AMR cluster assignments were inconsistent between two different 75% SNP subsamples. In one subsample, the individual clustered with the AMR group (American population). In the other, they shifted to the AFR group (African population). We suspect this individual may be admixed, having genetic ancestry from both AMR and AFR groups. Furthermore, randomizing UMAP's initialization based on changing the random state also influenced the cluster assignment for this admixed individual. These findings demonstrate

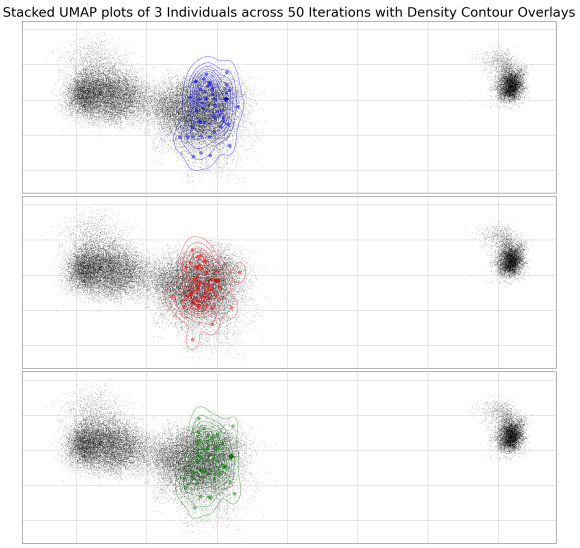


Figure 2: Point variability of three individuals across 50 subsamples using 90% of SNPs. The points for each individual appear to be quite scattered across the overall cluster.

how the specific SNPs included in each sample, UMAP's sensitivity to input data, and its inherent stochasticity collectively influence cluster assignments. However, this cluster assignment variation only occurs for a very small percentage of individuals. In our case, this was only true for one individual out of the 528 people in the American population.

5 Discussion and Conclusions

This study helps genomic scientists better understand the robustness and limitations of UMAP in visualizing genomic data. While cluster morphology remained stable across subsamples, the variability of point positions and cluster assignment for admixed groups highlights UMAP's sensitivity to input data and inherent variability. In fact, UMAP's impact on clustering for certain populations is consistent with previous research highlighting the role of initialization for preserving structure in UMAP [5].

The findings of this study provide practical guidance and key insights for scientists implementing UMAP for genetic data analysis. Our findings highlight the need for caution when interpreting certain individual level placements. Moreover, such results underscore the importance of careful SNP selection and imply that filtering for key SNPs could help researchers focus their analysis on more meaningful genetic variations. Our study raises several important open research questions. The distance between clusters in UMAP embeddings is not necessarily meaningful and can be prone to overinterpretation. More work needs to be done to convey this information more effectively, such as displaying clusters in separate panels or illustrating breaks in the underlying space. Furthermore, admixed populations present a unique challenge for UMAP. Further research could investigate the specific SNPs that drive cluster divergence for mixed populations. More generally, one could analyze how removing specific SNPs, such as those correlated with hereditary, hormonal, or epigenetic data, affects UMAP visualizations. This analysis would help us better understand the genetic underpinnings of clusters and what drives divergence. Lastly, future research should focus on how UMAP's performance compares with other dimensionality reduction techniques like t-SNE and PCA in the context of genetic data and subsampling. Ultimately, our findings and conclusions contribute to the broader goal of enabling more accurate and responsible use of UMAP in genomic studies.

ACKNOWLEDGEMENTS

The authors wish to express their gratitude to collaborators Dr. Diaz-Papkovich and Dr. Ramachandram for their invaluable feedback and insights throughout the study. We would also like to thank Professor Dr. Laidlaw not only for his guidance on this study but also the opportunity to take CSCI2370 and participate in exciting interdisciplinary research.

REFERENCES

[1] T. . G. P. Consortium. A global reference for human genetic variation. *Nature*, 526:68–74, 2015.

[2] A. Diaz-Papkovich, L. Anderson-Trocme, and S. Gravel. A review of umap in population genetics. *Journal of Human Genetics*, 66(1):85–91, Jan 2021.

[3] E. Elhaik. Principal component analyses (pca)-based findings in population genetic studies are highly biased and must be reevaluated. *Scientific Reports*, 12:14683, 2022.

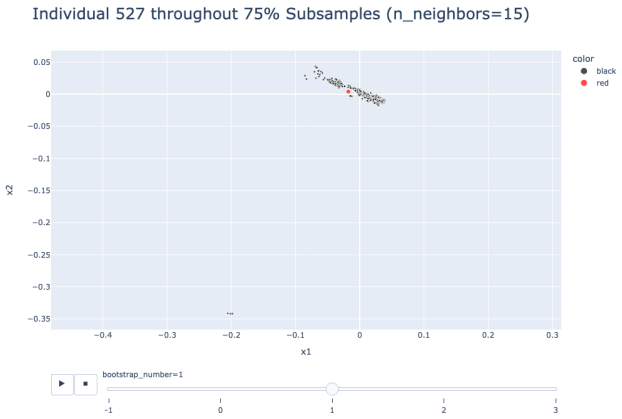
[4] S. M. Gopal et al. 'all of us' genetics chart stirs unease over controversial visualization. *Nature*, 617(7960):12–13, 2024.

[5] D. Kobak and G. C. Linderman. Initialization is critical for preserving global data structure in both t-sne and umap. *Nature Biotechnology*, 39:156–157, 2021.

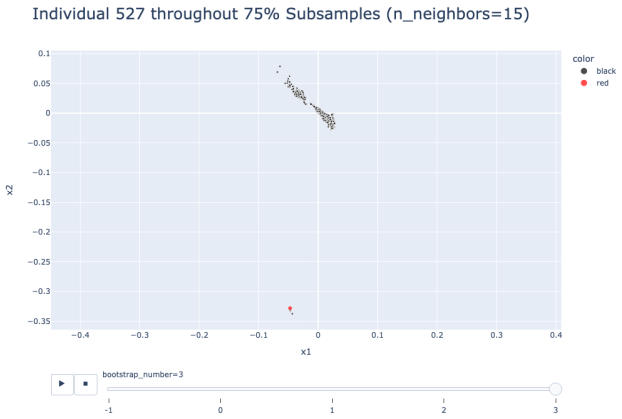
[6] D. K. Malomane et al. Efficiency of different strategies to mitigate ascertainment bias when using snp panels in diversity studies. *BMC Genomics*, 19(1):22, 2018.

[7] M. Pook et al. How imputation can mitigate snp ascertainment bias. *BMC Genomics*, 22(1):4, 2021.

6 APPENDIX

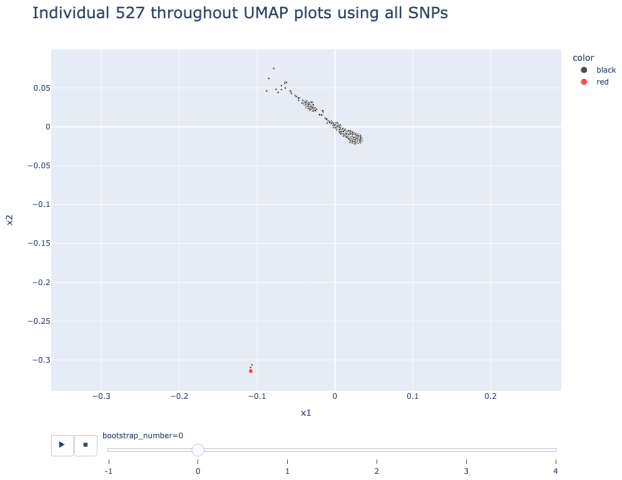


(a) Individual 527 is placed in the AMR cluster for one 75% SNP subsample.

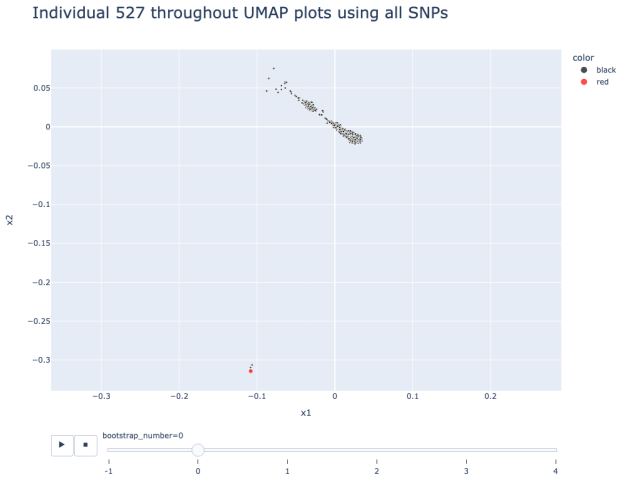


(b) Individual 527 is placed in the AFR cluster for a different 75% SNP subsample.

Figure 3: Cluster assignment by SNP subsampling.



(a) Individual 527 is placed in the AMR cluster for one UMAP random_state value.



(b) Individual 527 is placed in the AFR cluster for a different UMAP random_state value.

Figure 4: Cluster assignment by UMAP random_state.

Figures 3 and 4 both support our finding that cluster assignment may be inconsistent in some cases wfor certain individuals. Figure 3 highlights the effect of SNP subsampling while figure 4 demonstrates the impact of UMAP randomization.

7 Jasmine Liu Individual Contributions

7.1 Intellectual Contributions

As the Primary Investigator, at the start of the project, I co-developed the research framework by reviewing findings from prior studies on population genetics and dimensionality reduction. Although our research later pivoted, this helped establish a solid foundation for our analysis of UMAP and genomic data. During our pivot, I encouraged the transition from bootstrapping to subsampling in order to better simulate realistic depictions of genetic data. In terms of our results, I led the investigation into cluster assignment inconsistencies for mixed populations, identifying that both UMAP's stochasticity and SNP subsampling significantly contributed to variability in assignments. Throughout the project and in collaboration with Musa, I provided critical feedback on analysis results, helping to refine research questions and enhance the overall scope of the project.

7.2 Practical Contributions

Many of my practical contributions involved writing Python scripts for various tasks related to preprocessing and UMAP dimensionality reduction using subsampling techniques. For subsample implementation, I developed scripts to randomly subsample SNPs and compare UMAP embeddings across different sample sizes. I also implemented UMAP algorithms on our filtered dataset to evaluate point stability and cluster assignment consistency. Furthermore, I wrote scripts to create stacked UMAP plots, which visualized the embeddings of three individuals across 50 iterations and highlighted positional variability. Additionally, I generated UMAP plots for the AMR population across two different 75% SNP subsamples to illustrate the impact of subsampling on cluster assignments. I also implemented Procrustes alignment to align and compare embeddings across iterations, ensuring consistent comparisons. Toward the end, I led the integration of our analysis and outputs into the final paper and presentation materials, ensuring all figures were legible, appropriately titled, and easily interpretable.

8 Musa Tahir Individual Contributions

8.1 Intellectual Contributions

Generally, throughout the project, along with Jasmine, I regularly reviewed our results and offered feedback to refine our interpretation and clarify the scope of our project. One specific example is my contribution to analyzing and interpreting cluster morphological stability across different SNP subsampling levels. For many of our other UMAP visualizations, I proposed and developed the idea of using density contour overlays to represent variation and visualize point concentrations more mathematically, enhancing the clarity of results. Additionally, I contributed to discussions on using cluster centroids to track the average position of points across iterations or subsamples. This method could help quantify the stability of clusters, even when individual point positions vary due to UMAP's stochasticity. I also helped frame the project to emphasize the practical implications of UMAP visualizations in genetic studies, guiding researchers toward more responsible interpretations.

8.2 Practical Contributions

In terms of practical contributions, I helped adjust and normalize the UMAP axes in Python to ensure consistent visualizations across embeddings for clear comparative analysis. I also implemented the density contour plots to help illustrate cluster point visualizations and quantify variability effectively. In addition, I implemented the cluster morphology visualizations, which entailed generating and refining the plots to assess the stability of UMAP embeddings under different subsampling conditions. For these visualizations, I wrote Python scripts to randomly subsample SNPs and generate UMAP embeddings for different percentages of SNPs. Lastly, I was responsible for creating many of the slides for our progress reports and outlining the key points to ensure clear communication of our weekly progress.

Improving Understanding of Disease Spread Through Interactive 2D & 3D Visualization

Kei Yoshida* Richard Huang† Sambo Dachollom Simon Su

Brown University, Morgan State University, NIST

ABSTRACT

We present a novel synchronized and interactive 2D and 3D visualization framework for disease transmission, *Disease Dynamics Explorer (DDX)*. Traditional infectious disease data visualizations often rely on a simple 2D data visualization, but this often requires expert interpretation, making it less accessible to the general public, policymakers, and healthcare professionals. Our proposed visualization addresses these limitations by offering an intuitive and accessible way to explore disease dynamics. Using simulated data of Lassa fever spread in Nigeria as a case study, we demonstrated the effects of control measures on the disease spread with a synchronized 2D bar plot and 3D animations, showing the population in different states on a selected day. Evaluation results from both experts and non-experts suggested that our approach improves interpretability for disease-spread data visualizations.

Keywords: scientific visualization, disease transmission, interactive 2D and 3D visualization, interactivity.

1 INTRODUCTION

Lassa fever is an infectious disease that is endemic in parts of West Africa, and it continues to spread throughout other parts of the world [4, 8]. The work of [3] introduced the first mathematical model for Lassa fever with parameterized data from Nigeria, which simulates daily population counts for various human and rodent agent categories (e.g. infectious, susceptible, deceased), including categories related to control measures (quarantine and isolation).

In this study, we developed a visualization of datasets generated by this model, which helps researchers visually evaluate the effectiveness of certain control measures without diving deep into the details of the model or raw data. In collaboration with domain experts (including the first author from [3]), we designed and implemented *Disease Dynamics Explorer (DDX)*, a novel synchronized and interactive 2D & 3D visualization framework of Lassa fever data from their model.

Specifically, DDX is a 3D animated visualization of disease data over a geographical region that is synchronized with a 2D time-series view of corresponding variables, demonstrating the effects of non-pharmacological control measures on disease spread. Through user studies with domain experts and non-experts, we demonstrate that DDX **(1)** provides a general technique for visualizing disease data beyond existing simple tools (e.g., line charts), **(2)** facilitates deeper exploration of disease data by scientists even beyond Lassa fever, and **(3)** enables informed decision-making in disease management due to its improved interpretability. Because of this, our work is not only significant in scientific visualization but also for broader scientific research in disease spread.

1.1 Related Work

Modeling and visualizing Lassa fever spread is still a very new area of research, but there is a wide literature on visualizing disease data. Generally, simple 2D visualizations (e.g., Figure 1) have been commonly used for such purposes [2, 3]. These are often limited in the amount of information they can convey, as disease data is often represented in very high dimensions, whereas 3D visualizations have been shown to be more engaging and interpretable [10]. Additionally, 3D visualizations are more useful in illustrating disease spread over geographical regions [7, 12]. However, the simplicity of 2D visualizations also has benefits in facilitating fast parsing and understanding of data.

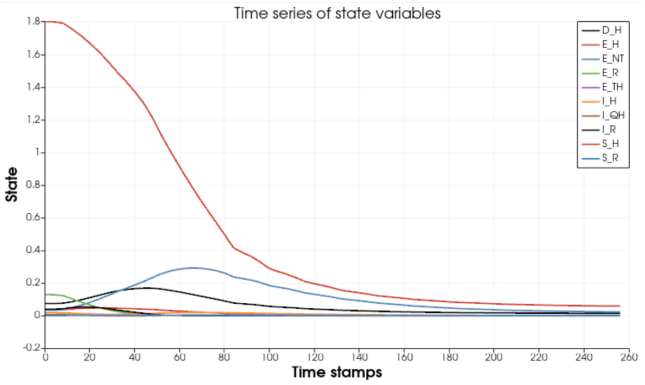


Figure 1: Traditional 2D line visualization of Lassa fever data as a line graph.

We elect to combine both 2D and 3D in our approach with DDX to bring the best of both worlds. There has been previous work using a similar approach to address the complexity of high-dimensional visualizations by developing hybrid 2D and 3D visualizations (e.g., visualization of network traffic data [9, 6, 11]), but to our knowledge, this approach has not been applied to disease spread data. DDX integrates this approach with interactive animations to help users track disease transmission over time.

2 LASSA DISEASE SPREAD VISUALIZATION

2.1 Data

We visualized three datasets provided by our collaborator, Sambo Dachollom. Each dataset, in CSV format, contained simulated data describing the progression of Lassa fever over time. The datasets consisted of daily population counts for various state variables for humans and rodents; each row corresponds to a day and each column corresponds to the number of individuals/rodents in each state on that day. In this project, we did not visualize the rodent variables to emphasize the disease spread in the human population. We preprocessed this data to generate random points in 2D space for each 3D agent in each state.

2.2 Implementation

We developed an interactive visualization of synchronized 2D and 3D data representations (see Figure 2) using ParaView [1] and Trame [5]. The visualization consists of two visualization panels with an interactive slider, offering a comprehensive view of the simulated Lassa fever dynamics.

*e-mail: kei_yoshida@brown.edu

†e-mail: richard_huang2@brown.edu

Figure 2: Disease Dynamics Explorer (DDX) visualizing a Lassa fever dataset for a user-selected time step. On the left is the 3D component, the top right is the 2D component, and the bottom right is the day-selection slider. In the 3D visualization, each avatar represents 10 humans.

3D Visualization: The left panel in Figure 2 shows 3D human-shaped figures (each avatar representing 10 people) with distinct colors corresponding to the state variables. The figures are arranged over a geographical map of Nigeria, providing some spatial context that the data represents. ParaView [1] was used to create vtk files for the 3D visualization, which were then integrated into our application using Trame [5]. Users can zoom in and out to explore regions of interest. Note that for our data, the position of each avatar in the map does not represent spatial information about the particular avatar, but other datasets with spatial information should incorporate this in future studies.

2D Visualization and Slider: The upper section of the right panel in Figure 2 contains a 2D bar plot, with each bar representing the state variables. The lower section has a slider allowing users to select a specific day to view. The 2D bar plot and the 3D visualization are dynamically updated based on the selected day. This enables users to interactively explore temporal changes in the data.

3 Evaluation

The evaluation of DDX consisted of two phases:

Expert Feedback: We consulted with our collaborators (domain experts), who expressed high satisfaction with the visualization. The use of ParaView [1] and Trame [5] provided accessibility, as the tool can be modified and repurposed easily by researchers. Experts appreciated the dynamic interactivity of the tool, particularly the ability to explore data over time and interact with the spatial aspects in the 3D visualization. While the current data used in this study lacks spatial information for each agent, the experts emphasized the potential of the visualization tool to be extended for future datasets. Their aims include (1) visualizing real data over larger geographical regions, where each individual agent is associated with a specific location, and (2) expanding the model to simulate the disease dynamics across multiple regions. They described the tool as a significant improvement from previous line chart visualizations, with the potential to increase the efficiency of data analysis for disease spread researchers beyond the current Lassa fever data.

Non-Expert User Study: We conducted a study with 10 participants unfamiliar with the dataset or domain. We compared their experience with DDX (Figure 2) to a traditional 2D visualization (Figure 1). From the quantitative analysis of the study result, we found three main results: **(1)** No significant difference in the number of listed insights was found between the traditional 2D ($M = 3.80$, $SD = 1.03$) and DDX ($M = 3.60$, $SD = 0.97$) ($t = 0.45$, $p = 0.66$). **(2)** Participants spent much more time (in seconds) exploring DDX ($M = 37.50$, $SD = 11.37$) than the traditional 2D visualization ($M = 14.70$, $SD = 6.98$) ($t = -5.41$, $p < 0.001$). **(3)** On a scale from 1 to 7, participants rated DDX much higher for DDX ($t = -9.19$, $p < 0.001$), with a mean rating of $M = 5.80$ (SD = 0.63) compared to $M = 3.20$ (SD = 0.63) for the traditional 2D visualization.

With traditional 2D visualizations (Figure 1), participants provided detailed, descriptive insights about individual trajectories (e.g., "Very high number of exposed humans, decreased after 100 days", "Exposed but not quarantined humans increased after dead humans increased"). However, they often struggled to contextualize the data (e.g., "I can tell you what each line means, but I'm not sure what the figure represents"). In contrast, insights from DDX focused more on overarching trends (e.g., "Initially everyone was exposed/infectious/dead, but the numbers decreased as susceptible people increased" or "Colors changed from red/black to yellow").

4 Conclusion

We implemented DDX, a novel synchronized and interactive 2D & 3D visualization of Lassa fever data with temporal tracking on Paraview/Trame [1, 5], using simulated data generated by the previously developed model [3].

Our user study highlighted the key strengths and limitations of DDX. While non-experts found it more intuitive and engaging, their insights were less detailed compared to traditional 2D visualizations, suggesting that DDX provides a broader understanding of disease progression but may require more features to aid in granular analysis. Furthermore, our qualitative feedback underscores the importance of tailoring visualizations to the intended audience. The current DDX may offer a compelling solution for users seeking high-level trends. However, for detailed trajectory analysis, traditional methods or a hybrid approach may be preferable. Future iterations could incorporate features like annotation tools or summary overlays to enhance both the breadth and depth of analysis.

There is also a clear direction for future work involving the dataset. Data features that would also improve the visualization include incorporating spatial data for agents in the model to make use of the geographical map, and tracking transitions of each agent between states (e.g., from healthy to sick). Specifically, these would make for a more intuitive use of the 3D space for visualization. Future work should experiment with the generalizability and the potential of DDX further by using other disease-spread data, especially those with spatial data.

ACKNOWLEDGEMENTS

The authors wish to thank Professor David Laidlaw and the students in CSCI 2370 for their feedback and support throughout the semester, and the participants in the user study.

REFERENCES

[1] U. Ayachit. *The ParaView Guide: A Parallel Visualization Application*. Kitware, Inc., Clifton Park, NY, USA, 2015.

[2] L. N. Carroll, A. P. Au, L. T. Detwiler, T.-c. Fu, I. S. Painter, and N. F. Abernethy. Visualization and analytics tools for infectious disease epidemiology: A systematic review. *Journal of Biomedical Informatics*, 51:287–298, 2014.

[3] S. Dachollom and C. E. Madubueze. Mathematical model of the transmission dynamics of lassa fever infection with controls. *Mathematical Modelling and Applications*, 5:65–86, 2020.

[4] E. Fichet-Calvet and D. J. Rogers. Risk maps of lassa fever in west africa. *PLoS Neglected Tropical Diseases*, 3(3):e388, 2009.

[5] S. Jourdain (Kitware), P. Avery, actions user, C. Harris, J. BOURDAIS, K. Vrabec, Will, A. Stucky, A. Huebl, J.-C. Fillion-Robin, K. Marchais, L. Macron, P. Tunison, RichardScottOZ, Nijso, charrisIT, and W. Dunklin. Kitware/trame: v3.7.0, Oct. 2024.

[6] E. Le Malécot, M. Kohara, Y. Hori, and K. Sakurai. Interactively combining 2d and 3d visualization for network traffic monitoring. In *Proceedings of the 3rd International Workshop on Visualization for Computer Security*, VizSEC '06, page 123–127, New York, NY, USA, 2006. Association for Computing Machinery.

[7] C. K. Leung, Y. Chen, C. S. Hoi, S. Shang, Y. Wen, and A. Cuzzocrea. Big data visualization and visual analytics of covid-19 data. In *2020 24th international conference information visualisation (iv)*, pages 415–420. IEEE, 2020.

[8] J. K. Richmond and D. J. Baglole. Lassa fever: epidemiology, clinical features, and social consequences. *Bmj*, 327(7426):1271–1275, 2003.

[9] S. Su, V. Perry, L. Bravo, S. Kase, H. Roy, K. Cox, and V. R. Dasari. Virtual and augmented reality applications to support data analysis and assessment of science and engineering. *Computing in Science Engineering*, 22(3):27–39, 2020.

[10] M. Teplá, P. Teplỳ, and P. Šmejkal. Influence of 3d models and animations on students in natural subjects. *International Journal of STEM Education*, 9(1):65, 2022.

[11] M. Vuckovic, J. Schmidt, T. Ortner, and D. Cornel. Combining 2d and 3d visualization with visual analytics in the environmental domain. *Information*, 13(1), 2022.

[12] J. Zhang, Y. Wang, W. Wanta, Q. Zheng, and X. Wang. Reactions to geographic data visualization of infectious disease outbreaks: an experiment on the effectiveness of data presentation format and past occurrence information. *Public Health*, 202:106–112, 2022.

A Individual Contributions

A.1 Kei

A.1.1 Intellectual Contributions

As the Primary Investigator, I developed the initial visualization framework in collaboration with Simon. Although we eventually had to make significant changes to the initial idea due to a misunderstanding of the existing dataset, it still served as a critical foundation for this project. Throughout the project, and in collaboration with Simon and Sambo, I provided initiative and guidance in shaping the visualization tool to align with the project's goals. Working closely with Richard, I led the design and implementation of the synchronized 2D and 3D visualization framework, ensuring its relevance and usability by incorporating feedback from collaborators. I facilitated regular discussions to refine key features, such as dynamic interactivity and time-dependent spatial representations.

A.1.2 Technical Contributions

Richard and I equally contributed to designing the entire visualization framework while maintaining clear communication with our collaborators. I initially implemented a traditional 2D line graph using our dataset as a proof of concept. Subsequently, I developed the visualization platform to display synchronized 2D and 3D visualizations, enabling seamless integration of the csv data and the vtk files prepared by Richard. This implementation in Python using Trame allowed users to view 2D and 3D visualizations interactively based on input. Additionally, I oversaw the integration of ParaView and Trame into the workflow, ensuring that the tool was accessible, customizable, and adaptable for future datasets. Richard and I jointly contributed to conducting the user study, writing this abstract, and preparing the final presentation. My technical contributions also involved addressing challenges in accurately representing disease spread dynamics and fostering innovative and intuitive approaches to the visualization of complex datasets.

A.2 Richard

A.2.1 Intellectual Contributions

Kei and I both contributed to designing many research questions for our collaborators and planning meetings in order to (more efficiently) extract the information we needed to complete the project. Near the beginning we both also helped narrow the scope of the project goals based on the new information we had received. Since my practical role was more involved with the static 3D visualization on Paraview, I was also able to help come up with how to design certain details for this component, such as how to place the human figures and overlay them with the geographical map. I helped come up with metrics that would be helpful in recording for our user study, and also noted that we may expect users to spend more time on our visualization as it is more complex than the baseline.

A.2.2 Technical Contributions

Kei and I contributed to brainstorming on the design of the 3D visualization component. I then implemented the static 3D visualization on Paraview. To do so, I also wrote some scripts for data preprocessing. These scripts generalize to other generated datasets from the Lassa disease model, allowing Sambo to continue visualizing different datasets after the conclusion of the project. Once the static 3D visualization was finalized, I designed an automated pipeline to create the same vtk files I was initially creating directly on Paraview. Using this, I wrote a Paraview Python script to automatically generate visualizations for every day in the simulation, which is then passed into Kei's Trame workflow. I contributed to the user study by helping record input on the visualization from some non-expert users.

A 3D Visualization Platform for GEDI Lidar Waveforms to Improve Ecological Understanding

Matthew Yoon*

Brown University

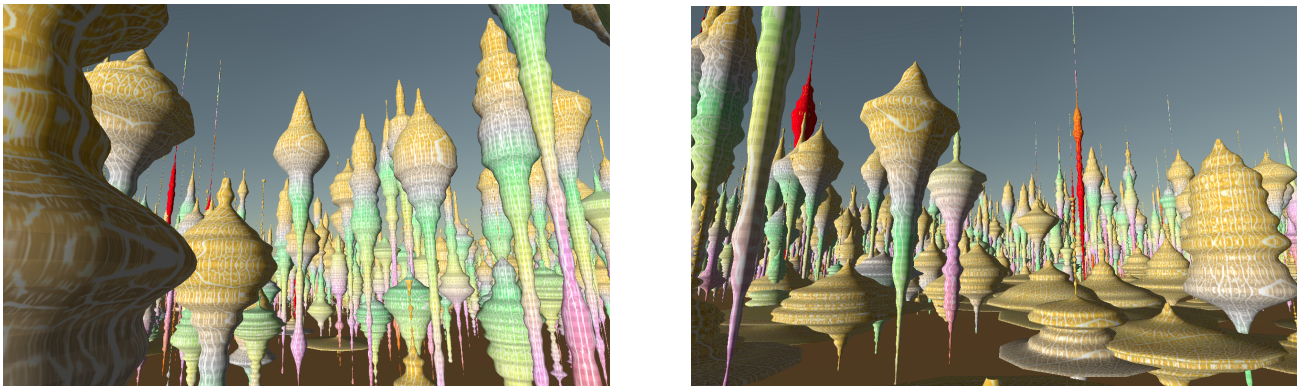


Figure 1: GEDI footprint of Amazon rainforest visualized in 3D platform.

Abstract

NASA's Global Ecosystem Dynamics Investigation (GEDI) mission captures detailed vertical profiles of Earth's surface using Lidar [1]. These footprints capture complex vegetation structures and terrain information [1]. Current state-of-the-art visualizations of GEDI data include maps of scalar values and isolated waveform plots [2], which lack spatial context and dimensionality to fully convey ecological complexity to researchers [3]. In this research, I developed a three-dimensional visualization platform in Unity to represent raw GEDI Lidar waveforms as generalized cylinder meshes. This tool integrates multiple ecological attributes and scalar values into a cohesive and immersive 3D environment, allowing users to "feel" the data. Initial user studies indicate that both researchers and students can more rapidly interpret canopy structure and distinguish between biomes using the 3D visualization compared to current two-dimensional methods. This work lays a foundation for future comprehensive ecological analyses, with potential applications in wildlife conservation, fire spread modeling, and also integration of temporal data for trend recognition.

Keywords: Forest canopy, vegetation structure, generalized cylinder mesh.

1 Introduction

Ecology researchers utilize these intricate vertical canopy profiles to understand forest structure, biodiversity, and ecosystem health [7]. Traditionally, researchers visualized these data through two-dimensional visualizations such as Cartesian maps of canopy height and numeric metrics (e.g., RH-98) alongside isolated waveforms [2, 7]. Although these methods have been useful, they possess critical limitations. 2D maps are limiting because they separate scalar values from their three-dimensional context, while examining individual waveforms lose spatial context, obscuring patterns within the broader landscape [3, 4]. These constraints could hinder nuanced understanding of ecological characteristics like understanding canopy habitats, vegetation layering, and habitat suitability for wildlife.

To address these limitations, I created an immersive 3D platform that spatially represents GEDI Lidar waveforms along with associated ecological metrics. By transitioning from 2D maps and disjointed waveform plots to an integrated 3D environment, this approach may enable users to more intuitively identify ecological patterns and interpret complex canopy structures [4]. In this paper, I describe the methods used to create this 3D visualization, report on user studies that indicated improved efficiency in ecological assessments, and discuss the implications for future ecological research.

2 Related Work

Previous efforts in Lidar visualization have largely focused on representing canopy height metrics or derived point clouds in 2D maps or static profiles [6]. While these methods enable advanced ecological analysis, they often lack the intuitive spatial sense that a true 3D environment can provide, similar to the navigable experience of Google Earth. Existing work on full-waveform Lidar 3D data usually emphasizes algorithmic or processing frameworks rather than developing tools that enable true immersion [5]. This presented platform builds on these foundations by converting raw GEDI waveforms into generalized cylinder meshes, allowing researchers and non-experts to navigate and dynamically filter complex ecosystems interactively.

3 Methodology

Each waveform was converted into a generalized 3D mesh by revolving the vertical amplitude around a central axis, creating a cylindrical form that reflects the waveform's profile. Each generalized cylinder maintained a consistent integral for uniformity. This approach preserves subtle structural information, which is often lost when waveforms are reduced to discrete point clouds or

*e-mail: matthew_yoon@brown.edu

scalar maps. Figure 2 depicts a raw waveform next to its 3D representation.

In addition to raw waveform geometry, I integrated scalar attributes derived from the data. For example, relative height metrics (e.g., RH50, RH98) were encoded as surface textures on the cylinders as color gradients, allowing users to understand mid-canopy structures and potential wildlife habitats. The platform supports other scalar values such as entropy or soil moisture to be mapped as additional noise textures on the surface. This Unity platform also allows for dynamic filtering, texture adjustments, and user navigation. Users can zoom, rotate, traverse, and scale the visualization.

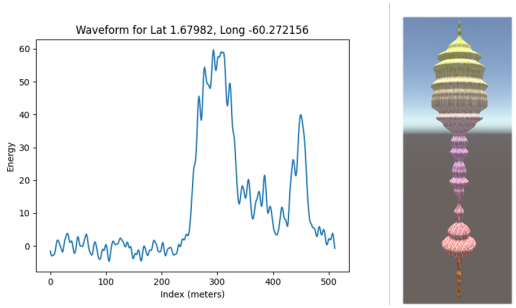


Figure 2: 2D and 3D representation of the same waveform.

4 User Study Design and Results

I evaluated the effectiveness of the 3D visualization platform by comparing it directly to traditional 2D methods in some tasks. Participants performed four main tasks involving datasets from four distinct biomes: desert, swamp, forest, and rainforest.

4.1 Attribute Ranking Task

Participants first used standard 2D waveform comparisons to rank multiple ecological attributes (vegetation height, vegetation density, vertical complexity, ground openness, and terrain variation) of each biome. Although there is not necessarily an accuracy, the 3D environment allowed users to quickly form more nuanced characterizations of the landscape, revealing subtle ecological differences more intuitively. These results are shown in Figure 3. The 2D characterizations featured either mostly uniform rankings (swamp and forest) or mostly extreme rankings (rainforest and desert). However, users described each biome with greater nuance when using 3D visualizations. The average completion time was 5:17.4 in 2D, and 3:16.1 in 3D, demonstrating more rapid characterization.

4.2 Biome Identification Task

Given subsets of nine 2D waveforms corresponding to a specific biome, participants matched each subset to one biome. For environments with similar characteristics such as forest versus swamp, 2D methods proved challenging. When using 3D visualizations, participants maintained similar accuracy to the 2D baseline, albeit with a minor decrease in one task. However, the 3D approach achieved these results in less than half the time (1:27.3 in 2D versus 0:40.1 in 3D), demonstrating a significant improvement in efficiency. This gain is likely due to the enhanced spatial context and more nuanced structural differences that were not as readily apparent in 2D.

4.3 Study Limitations

It is important to note that the 3D visualizations were presented as static images, which limited the inherent spatial depth and interactivity of the 3D tool. This constraint prevented participants from leveraging the platform's interactive capabilities, likely affecting both accuracy and efficiency of the user study.

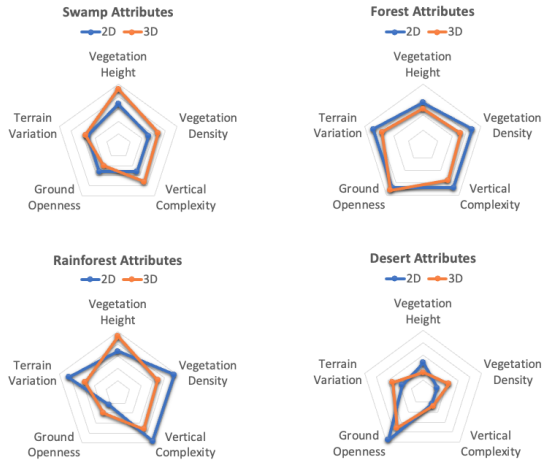


Figure 3: Results of attribute ranking task.

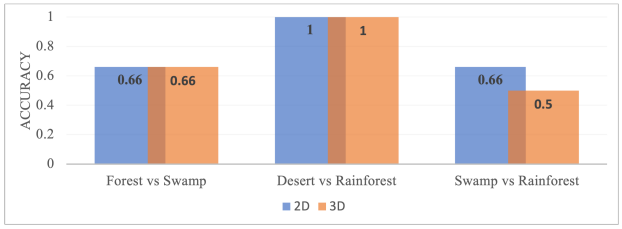


Figure 4: Results of biome identification task.

4.4 Implications and Future Work

These results suggest that incorporating spatial context alongside multidimensional attributes into one 3D platform can accelerate ecological analysis and decision making, which is especially critical in areas such as wildfire modeling and wildlife conservation. Moving forward, improving the user study to include interactive exploration would likely enhance data interpretation, potentially yielding improvements to both accuracy and efficiency.

5 Applications and Future Research Directions

The introduced 3D platform could be applied to a broad range of ecological research challenges. For wildlife conservation, identifying mid-canopy habitats through RH metrics becomes more straightforward in a 3D environment. Time-sensitive applications, such as modeling fire spread, benefit from quicker analysis through spatial representations of vegetation layering and density. Future research could also consider web-based platforms for non-experts in hopes of democratizing ecological data interpretation.

6 Conclusion

This research presents a 3D visualization platform that transforms raw GEDI Lidar waveforms into interactive cylindrical representations, providing a new way to view and interpret ecosystem structure. While preliminary user studies suggest that 3D visualizations may facilitate more nuanced and efficient interpretation compared to standard 2D methods, these findings are not yet definitive. However, they point toward the potential value of immersive, context-rich visualization techniques in improving ecological understanding. As the platform is applied to a wider range of datasets and tasks, it may become an important tool in supporting more informed decision-making in ecology, climate modeling, and resource management.

Acknowledgements

The author thanks Dr. David Laidlaw and Ziang Liu.

References

[1] R. Dubayah, J. Armston, S. P. Healey, J. M. Bruening, P. L. Patterson, J. R. Kellner, L. Duncanson, S. Saarela, G. Ståhl, Z. Yang, et al. Gedi launches a new era of biomass inference from space. *Environmental Research Letters*, 17(9):095001, 2022.

[2] L. Duncanson, J. R. Kellner, J. Armston, R. Dubayah, D. M. Minor, S. Hancock, S. P. Healey, P. L. Patterson, S. Saarela, S. Marselis, et al. Aboveground biomass density models for nasa's global ecosystem dynamics investigation (gedi) lidar mission. *Remote Sensing of Environment*, 270:112845, 2022.

[3] T. D'Urban Jackson, G. J. Williams, G. Walker-Springett, and A. J. Davies. Three-dimensional digital mapping of ecosystems: a new era in spatial ecology. *Proceedings of the Royal Society B*, 287(1920):20192383, 2020.

[4] T. Niedomysl, E. Elldér, A. Larsson, M. Thelin, and B. Jansund. Learning benefits of using 2d versus 3d maps: Evidence from a randomized controlled experiment. *Journal of Geography*, 112(3):87–96, 2013.

[5] Å. Persson, U. Söderman, J. Töpel, and S. Ahlberg. Visualization and analysis of full-waveform airborne laser scanner data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(3/W19):103–108, 2005.

[6] T. Smith, A. Rheinwalt, and B. Bookhagen. Determining the optimal grid resolution for topographic analysis on an airborne lidar dataset. *Earth Surface Dynamics*, 7(2):475–489, 2019.

[7] M. Torresani, D. Rocchini, A. Alberti, V. Moudrỳ, M. Heym, E. Thouverai, P. Kacic, and E. Tomelleri. Lidar gedi derived tree canopy height heterogeneity reveals patterns of biodiversity in forest ecosystems. *Ecological Informatics*, 76:102082, 2023.

Appendix

These are the images of the 3D platform shown in the user study.

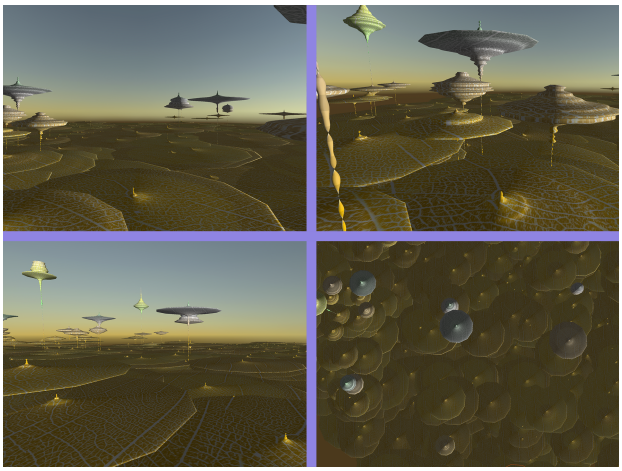


Figure 5: Sahara desert.

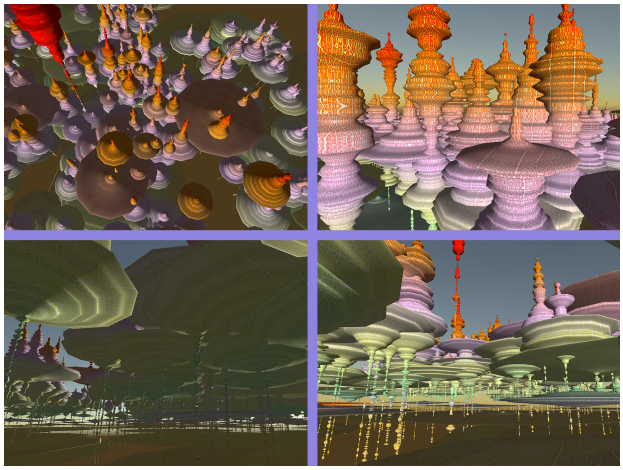


Figure 6: Amazon rainforest.

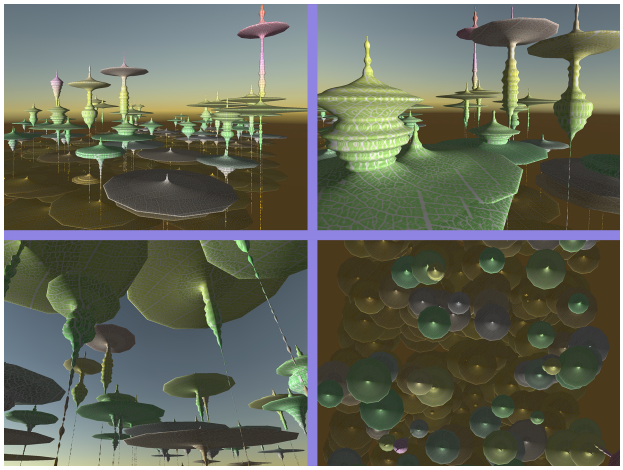


Figure 7: Temperate forest.

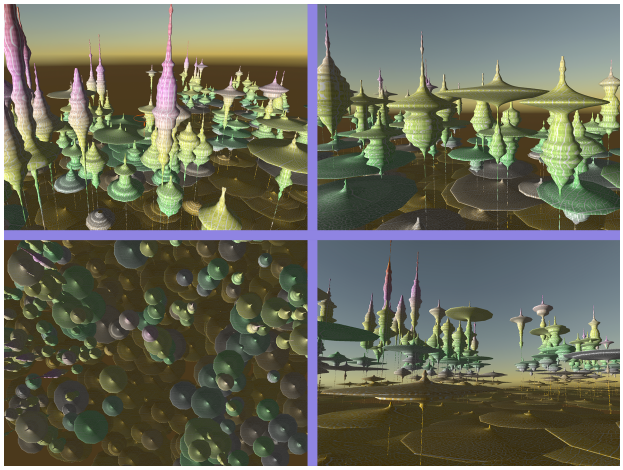


Figure 8: Swamp.

Subcircuit visualizations enhance neural network interpretability

Sam Musker* Aalok Sathe†

Brown University: CS2370 Interdisciplinary scientific visualization, Fall 2024

ABSTRACT

We develop a visualization tool for neural subnetworks, future developments of which could aid experts in scientific discovery. Our tool visualizes the internal subcircuits responsible for specific tasks at the parameter level, showing multiple subnetworks simultaneously along with their overlaps and evolution during training. We evaluate our tool through comparison with existing visualization methods and through a user study with domain experts. Additionally, we contribute a modification of the continuous sparsification algorithm for more stable subnetwork identification and introduce two novel metrics for assessing identified subnetworks. Our work may enhance the understanding of neural networks and future developments could support advances in neuroscience and cognitive science.

Keywords: neural network interpretabiliy, subcircuit identification, subcircuit visualization

1 INTRODUCTION

Advances in neural network interpretability present opportunities for visualizing newly available information. Techniques for identifying subnetworks responsible for specific subtasks have emerged, such as training masks over parameters to isolate functionalities. While promising, there is a need for better tools to visualize these subcircuits. We develop parameter-level visualization methods that display multiple subnetworks simultaneously, showing their overlaps and evolution during training. Our approach represents the network as a graph with nodes and edges for neurons and connections, using color mixing to depict overlapping subnetworks.

Developing advanced visualization methods for subnetwork identification is significant for artificial intelligence and its interdisciplinary applications. As neural networks become more complex and are deployed in critical sectors, understanding their internal workings becomes increasingly important [5]. Firstly, enhancing interpretability addresses the "black box" challenge in deep learning. Visualizing subcircuits responsible for specific tasks allows researchers to understand information processing and representation. This transparency is crucial for diagnosing and mitigating biases, understanding decision-making processes, and ensuring models behave as intended.

Secondly, future developments of this work could accelerate scientific discovery by facilitating cross-disciplinary collaboration. Mapping and visualizing subnetworks aligns with neuroscientific methods of studying brain functionality [2]. By drawing parallels between artificial neural networks and biological systems, this research has the potential to contribute to neuroscience and cognitive science [1]. For example, cognitive scientists could use these visualization methods to identify that two diverse tasks are solved by a network using a common or overlapping subnetwork, supporting hypotheses about task relatedness.

*e-mail: samuel_musker@brown.edu

†e-mail: aalok@brown.edu

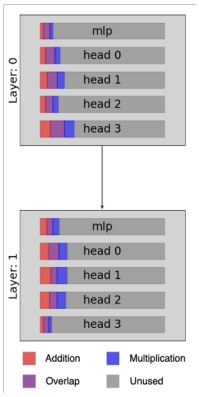


Figure 1: The basic visualization due to Lepori [4]. Note that this static visualization does not show parameter level information or changes during training.

2 RELATED WORK

Prior work with ANNs has focused on feature-level and representation-level explanations of function [3, 6, 7, 11]. However, recent advances in subnetwork identification have provided new avenues for neural network interpretability. Lepori [4] introduced NeuroSurgeon, a toolkit for identifying the subnetworks involved in a neural network solving a particular part of a composite task. This technique enables the extraction of subnetworks responsible for specific functionalities, but does not provide a visualization of subnetworks at the parameter level or a dynamic view of subnetwork changes through training, limiting possible insights [10].

3 METHOD

We implement a visualization tool that improves upon existing approaches in four key ways: (1) visualizing subnetwork structure at the parameter level, (2) showing multiple subnetworks simultaneously, (3) displaying overlaps between subnetworks through color mixing, and (4) showing subnetwork evolution during training. Finally, we evaluate our tool through an insights-based approach using expert users.

To demonstrate the tool's capabilities, we train a neural network to classify synthetically generated shapes that are either circle, oval, square, or rectangle, as shown in 2. The network architecture is a fully-connected deep neural network with two hidden layers, each of width 16, with a ReLu activation function in the hidden layers and a softmax activation function on the output layer. The network achieves a validation accuracy of 53% after 20 epochs of training compared to a baseline guessing rate of 25%. We implement the continuous sparsification algorithm [9] to identify four subnetworks, one for each shape.

4 RESULTS

Using our visualization tool, we derive several key insights about neural network mechanisms. We observe that while most two-way subnetwork overlaps maintain around 40% overlap, the circle-oval and square-rectangle subnetwork pairs show 70-80% overlap by the

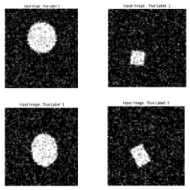


Figure 2: Example generated shapes classified by the network.

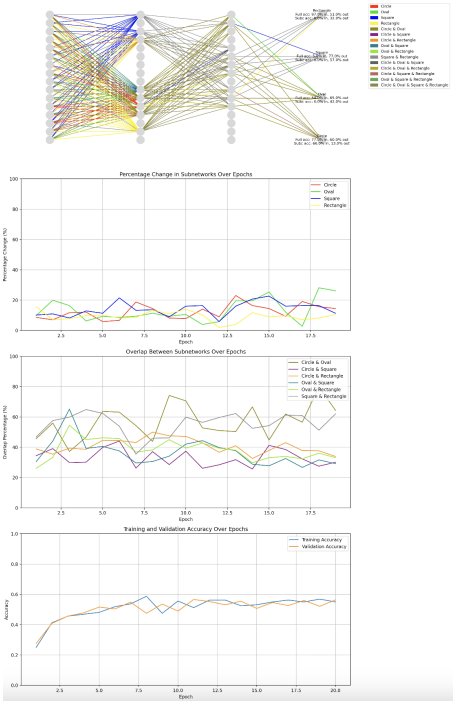


Figure 3: A final-epoch snapshot of our new subnetwork visualization. Users can identify that the network does not learn an elegant elongation-roundness solution by the absence of relevant overlaps, and can diagnose subnetwork selection deficiencies of high subnetwork turnover and "hanging" neurons. To view the demo in action, the reader may access a video recording here.

end of training. This suggests the network treats circles/ovals and squares/rectangles as paired categories rather than learning an elegant solution based on roundness and elongation features.

Our visualization also revealed deficiencies in the subnetwork identification algorithm, showing high turnover (10-20%) in identified weights between epochs and the presence of "hanging" neurons with incomplete connections. To address these issues, we developed a modified algorithm that optimizes over three trailing snapshots, reducing weight turnover to 0-10%.

There is a deficit of good methods for evaluating subcircuits delivered by selection algorithms. [4] relies primarily on a method in which the subcircuit is removed and the effect on in class versus out of class performance is evaluated. However, this method could be improved or augmented with other techniques.

We additionally contribute two novel metrics for evaluating subcircuits: a perturbation metric comparing the effects of perturbing subnetwork versus random weights, and an activation prediction metric using subnetwork masks to predict full network activations. While these metrics show promise, they remain under development, with the perturbation metric showing high variability and the activation prediction metric showing unexpectedly similar results between different subnetwwork selection algorithms.

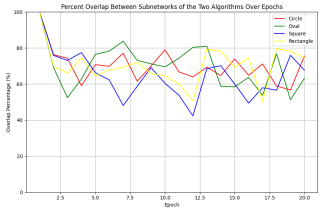


Figure 4: Overlaps between the subnetworks using either single or the trailing 3 snapshots. Overlaps of 70% suggest that the modified algorithm may select meaningfully different subnetworks.

We conducted a user-study using an insight-based methodology [8] to understand whether our method provided additional insights not available before. Testing confirmed that users were able to more confidently draw insights into the neural network than when using a baseline replication of an existing visualization, for example:

> *"I am more confident [after viewing the new dynamic parameter-level visualization, in contrast to my original visualization] that a template is being applied at layer 0. This layer contains the highest overall density of connections after pruning, and appears to have particular neurons dedicated to individual shapes (i.e. neuron 4 and the final neuron dedicated to squares)."*
>
> *– Michael Lepori (author of NeuroSurgeon [4])*

5 Open Research Directions

Two main research directions remain: improving subnetwork identification algorithms and developing better metrics for evaluating selected subcircuits. First, addressing "hanging" neurons would enhance subnetwork identification. This could be done post-hoc by pruning disconnected neurons or finding connections that maintain good performance. Alternatively, devising algorithms that search only over connected circuits could solve this issue. Second, improving evaluation metrics offers several paths. The perturbation metric's noise from excessive perturbation magnitude can be remedied. Re-perturbing differently during evaluation may average out alterations that cause performance drops. The activation prediction metric might not differentiate between subnetworks due to trivial predictions, such as all weights activating or none. Constraining the prediction output space may resolve this.

6 Conclusion

Our work introduces a novel visualization tool for neural subnetworks, future developments of which could enable deeper understanding of network mechanisms and support scientific discovery. Applied to a shape recognition task, it reveals insights about network learning and diagnoses algorithmic deficiencies. We also present a modified subnetwork identification algorithm and new evaluation metrics. Open questions include enhancing subnetwork identification algorithms to address "hanging" neurons and refining the evaluation metrics.

Acknowledgements

The authors wish to thank Ellie Pavlick, Roman Feiman, and Michael Lepori for their expert direction and collaborative involvement in this work. The authors wish to thank David Laidlaw for his support in the development of the project.

REFERENCES

[1] R. Cao and D. Yamins. Explanatory models in neuroscience: Part 1 – taking mechanistic abstraction seriously. *arXiv*, Apr. 2021.

[2] S. Chung and L. F. Abbott. Neural population geometry: An approach for understanding biological and artificial neural networks. *Current opinion in neurobiology*, 70:137–144, 2021.

[3] L. Gao, X. Liu, C. Liu, Y. Zhang, G. Fiumara, and P. D. Meo. Key nodes identification in complex networks based on subnetwork feature extraction. *Journal of King Saud University - Computer and Information Sciences*, 35(7):101631, 2023.

[4] M. A. Lepori, E. Pavlick, and T. Serre. Neurosurgeon: A toolkit for subnetwork analysis, 2023.

[5] S. A. Matveev, I. V. Oseledets, E. S. Ponomarev, and A. V. Chertkov. Overview of visualization methods for artificial neural networks. *Computational Mathematics and Mathematical Physics*, 61(5):887–899, May 2021.

[6] A. Nasser, D. Hamad, and C. Nasr. Kernel pca as a visualization tools for clusters identifications. In S. Kollias, A. Stafylopatis, W. Duch, and E. Oja, editors, *Artificial Neural Networks – ICANN 2006*, pages 321–329, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[7] C. Olah, L. Schubert, and A. Mordvintsev. Feature visualization. *Distill*, 2017.

[8] P. Saraiya, C. North, and K. Duca. An insight-based methodology for evaluating bioinformatics visualizations. *IEEE transactions on visualization and computer graphics*, 11:443–56, 07 2005.

[9] P. Savarese, H. Silva, and M. Maire. Winning the lottery with continuous sparsification, 2021.

[10] W. Schneider, W. Eckstein, and C. T. Steger. Real-time visualization of interactive parameter changes in image processing systems. In G. G. Grinstein and R. F. Erbacher, editors, *Visual Data Exploration and Analysis IV*, volume 3017, pages 286 – 295. International Society for Optics and Photonics, SPIE, 1997.

[11] J. Vig. A multiscale visualization of attention in the transformer model. *CoRR*, abs/1906.05714, 2019.

CONTRIBUTIONS

Sam Musker

Initial project conceptualization. Proposal writing. Consultation with expert users. Software engineering. Qualitative analysis of results. Report writing.

Aalok Sathe

Consultation with expert users. Software engineering assistance. Qualitative analysis of results. Report writing and progress documentation.

Applying High-Resolution Q-Space MRI to Map the Mouse Atria

Thais Del Rosario Hernandez* Richard Gilbert, MD†

Brown University

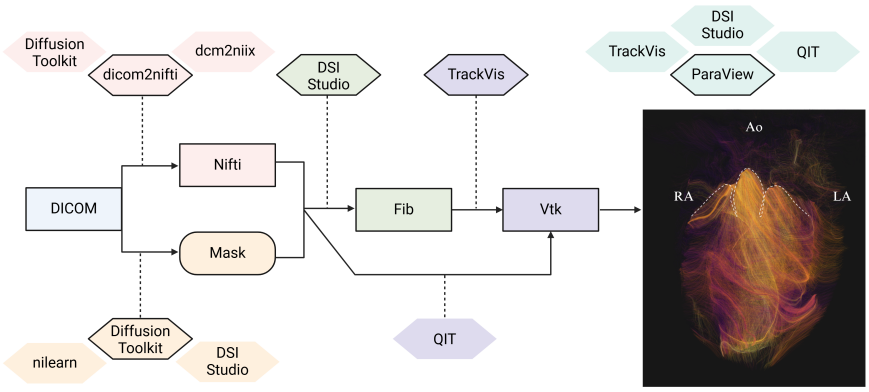


Figure 1: 3D tractography reconstruction of mouse hearts, visualized using ParaView, with a focus on atrial architecture. The nodes outlined in black represent the workflow and software used to process DICOM files. Alternative software and/or steps are shown without an outline. Ao: Aorta; LA: left atrium; RA: right atrium

Abstract

Cardiac structure is widely studied due to the multitude of pathologies that can impact heart function and the severity of their symptoms. However, the atria remain an understudied region of the heart due to their thin walls and architectural complexity. To address this gap, we processed and visualized mouse heart imaging data obtained with high geometric resolution Q-space MRI (QSI) to assess the myoarchitectural organization of the atria. We documented the macro- and micro-structure of the atria in 10 mouse hearts, highlighting inter-sample patterns and morphological landmarks. Our work provides a necessary step for the construction of an architectural atlas of the atria in mammals.

Keywords: Cardiovascular disease, tractography, cardiac structure.

1 Introduction

Cardiovascular diseases (CVDs) are the leading cause of death globally[1]. Arrhythmia is a common complication of CVD and refers to irregular heart rhythms that are the result of malfunctioning electrical pathways in the heart. The heart is anatomically divided into four chambers: the two upper chambers, known as the atria, and the two lower chambers, known as the ventricles. Electrical signals must travel through these chambers in a highly coordinated manner for the heart to function properly. Atrial fibrillation (AF) is the most prevalent type of arrhythmia, affecting more than 33 million people worldwide with a range of complications such as palpitations, fatigue, stroke, and even heart failure. There is a growing need to develop new diagnostic and therapeutic strategies for this widespread and often debilitating condition.

*e-mail: thais_del_rosario_hernandez@brown.edu

†e-mail: rjgilbert12@gmail.com

Fiber orientation — how the heart's muscle fibers are aligned — plays a critical role in how electrical signals propagate through the heart. In the ventricles, the orientation of these fibers has been well studied [2]. However, the atria, which are critical regions in the aberrant signaling characteristic of AF, have not been studied as extensively. The atrial walls are much thinner than the ventricular walls, and the complexity of fiber organization in the atria further complicates the use of imaging techniques that do not have enough resolution to capture their microstructure.

To address this gap, we investigated atrial architecture using a rodent model system. Mice and rats are widely used in cardiovascular research due to their physiological similarities to humans [3, 4, 5]; Additionally, their ease of handling and relatively short lifespans allow for rapid study of disease progression under both baseline and genetically altered conditions. The primary data for this project was obtained from excised mouse hearts, which were imaged using high-resolution Q-space MRI (QSI). This technique allows for detailed mapping of the fiber orientations in small structures such as the atria. This project aims to advance our understanding of atrial architecture and its role in atrial fibrillation, with the potential to apply our findings to other mammalian samples and eventually the human heart.

2 Related Work

Tractography is a powerful imaging technique that allows us to visualize 3D reconstructions of tissue fibers. It is primarily used to investigate brain structure - specifically, the connectivity and integrity of neural pathways across different regions of the brain in different disease states. While tractography has been used to study cardiac structure, several studies focusing on atrial architecture highlight the limitations of low-resolution imaging modalities [6, 7, 8]. Furthermore, reconstruction software and tractography visualization tools are tailored for brain data; they often include references and default parameters optimized for white matter data analysis, which is markedly different from the ever-moving cardiac muscle. QSI is highly sensitive to microstructural changes, allowing visualization

of fiber tracts at a resolution fine enough to capture changes occurring at the scale of individual myocytes in rodent models. This level of detail is essential for studying the intricate architecture of the atria and provides a means to investigate how electrical signals propagate through the heart at a cellular level.

3 Approach

3.1 Data processing

Our selected processing steps are as follows: DICOM files were converted to Nifti format using the Python package dicom2nifti [9]. The bvals, bvec and b-table files were extracted from the 2dseq files using DSI Studio [10]. Mask files were generated using the DICOM files as input in Diffusion Toolkit [11]. Nifti files and masks were used for track reconstruction in DSI Studio. Finally, the tracks were exported into .trk format and converted into .vtk format using TrackVis for visualization in ParaView [11, 12]. In addition to these software, we tested other tools - namely Python package nilearn for mask generation, dcm2niix for DICOM to Nifti file conversion, and Quantitative Imaging Toolkit (QIT) for track reconstruction [13, 14, 15]. The alternative software and their corresponding steps are shown in Figure 1. The parameters for each tool used in the data processing workflow were adjusted according to documentation recommendations and qualitative evaluation of the reconstructed tractography.

3.2 Macrostructural validation

In order to validate the workflow, software, and parameters chosen for data processing, we evaluated the ventricular structure of the mouse heart (Figure 4). The ventricles exhibit a distinct helical arrangement of fibers previously documented in both rodent and human studies [2].

4 Results

The four heart chambers are connected by four valves that regulate the flow of blood from upper chambers down to the lower chambers, and then back up to the aorta and the pulmonary artery. These bridging structures are key to identifying the beginning of the atrial chambers. We generated coronal and sagittal slices of the heart to delineate the gross morphology of the atrial chambers and segment the fibers comprising the outer walls, respectively. The tracks for the outer walls of the atria could then be independently segmented, revealing a thin yet coherent layer of fibers that do not follow the directionality of the ventricular fibers (Figure 2). Key differences between the right and left atrium are noted in the following subsections.

Right Atrium The right atrium (RA) is located closer to the top of the heart and its outer wall protrudes mainly vertically away from the thicker ventricular wall. The shape of its outer wall is rounded and the fibers are not externally connected to the ventricular walls. Although the atrioventricular transition can be clearly visualized with a sagittal view, the whole chamber is best visualized using a 3D view due to its ”folded” structure.

Left Atrium The left atrium (LA) is larger than the RA, with a more elongated shape directed toward the apex of the heart. In our experience, the LA is more difficult to differentiate from the left ventricle due to its proximity to the aorta and its corresponding aortic valve, which can be mistaken for the marker between the left ventricle and LA: the mitral valve.

Previous work using tractography to visualize cardiac structure in pigs, mice, rats, and rabbits have only focused on ventricular structure [16, 17, 18, 19]. A recent paper investigated and visualized the LA in goat hearts using DTI and found no common fiber pattern across samples, reporting instead a broad range of structures [7] (Figure 3). The authors emphasized the limitations of the resolution used for their imaging. Consequently, although goat hearts are markedly larger than mouse hearts, the resolution was not sufficient to capture enough consistency in the goat model to characterize the LA. In our study we observed consistent patterning of the fibers in both the left and right atrium, and we were able to validate our findings with histology references.

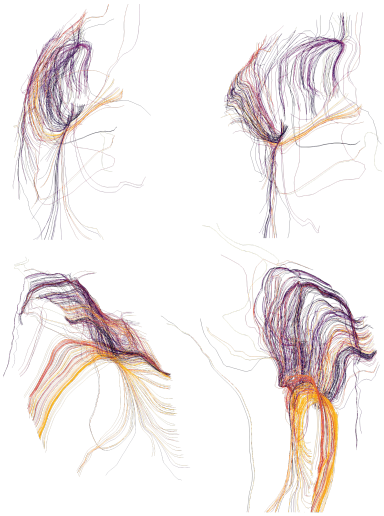


Figure 2: Coronal (left) and sagittal (right) views of the segmented outer walls of the RA (top) and the LA (bottom).

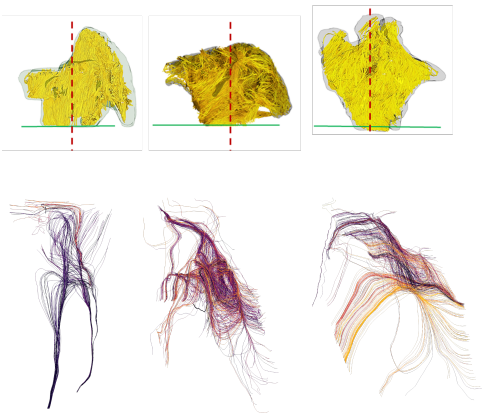


Figure 3: Comparison of the structural coherence obtained using DTI and QSI. (Top) Tractography reconstruction of the LA from 3 goats using DTI, adapted from Kamali et al. (2023). (Bottom) Tractography reconstruction of the LA from 3 mice using QSI.

5 Conclusion

We present a series of annotated tractography visualizations generated with parameters tailored for cardiac reconstruction obtained using high-resolution QSI data. We highlight the similarities in atrial structure and organization in the context of other heart regions, as well as some key differences between the RA and the LA. Our current work provides an essential step to generate a reference for comparison across rodent cardiac models and, eventually, human data.

Acknowledgements

Special thanks to Dr. David Laidlaw for his continuous support and prompt feedback. The author would also like to thank collabora-

tor Dr. Richard Gilbert for taking the time to meet regularly and providing helpful suggestions throughout the project.

REFERENCES

[1] World Health Organization. *Cardiovascular diseases*. Accessed: 2024-12-10. URL: https://www.who.int/health-topics/cardiovascular-diseases.

[2] E. N. Taylor et al. "Alterations in Multi-Scale Cardiac Architecture in Association With Phosphorylation of Myosin Binding Protein-C". In: *J Am Heart Assoc* (2016).

[3] D. Schüttler, A. Bapat, and S. Kääb. "Animal Models of Atrial Fibrillation". In: *Circ Res* (2020).

[4] Breckenridge R. "Heart failure and mouse models". In: *Disease models mechanisms* 3.138–143 (2010). DOI: 10.1242/dmm.005017.

[5] Hasenfuss G. "Animal models of human cardiovascular disease, heart failure and hypertrophy". In: *Cardiovascular research* 39.60-76 (1998). DOI: 10.1016/s0008-6363(98)00110-2.

[6] T. T. Wang et al. "Resolving myoarchitectural disarray in the mouse ventricular wall with diffusion spectrum magnetic resonance imaging". In: *Annals of biomedical engineering* 38.9 (2010). DOI: 10.1007/s10439-010-0031-5.

[7] R. Kamali et al. "Contribution of atrial myofiber architecture to atrial fibrillation". In: *PloS one* 18.1 (2023). DOI: 10.1371/journal.pone.0279974.

[8] O. Berenfeld et al. "Animal models of human cardiovascular disease, heart failure and hypertrophy". In: *Cardiovascular research* 39.60-76 (1998). DOI: 10.1016/s0008-6363(98)00110-2.

[9] A. Brys. *dicom2nifti*. icometrix, 2016. URL: https://github.com/icometrix/dicom2nifti.

[10] F. Yeh. *DSI Studio*. URL: http://dsi-studio.labsolver.org.

[11] R. Wang and V.J. Wedeen. *TrackVis*. URL: https://trackvis.org/.

[12] J. Ahrens, B. Geveci, and C. Law. "Visualization Handbook". In: ed. by C. D. Hansen and C. R. Johnson. Burlington, MA, USA: Elsevier Inc., 2005. Chap. ParaView: An End-User Tool for Large Data Visualization, pp. 717–731. URL: https://www.sciencedirect.com/book/9780123875822/visualization-handbook.

[13] A. Abraham et al. "Machine learning for neuroimaging with scikit-learn". In: *Frontiers in neuroinformatics* (2014).

[14] Li X. et al. "The first step for neuroimaging data analysis: DICOM to NIfTI conversion". In: *J Neurosci Methods* (2016).

[15] R.P. Cabeen, D. H. Laidlaw, and A. W. Toga. "Quantitative imaging toolkit: software for interactive 3D visualization, data exploration, and computational analysis of neuroimaging datasets". In: *ISMRM-ESMRMB Abstracts* (2018), pp. 12–14.

[16] J. J. Torres, S. P. Dies, and T. S. Hidalgo. "Visualization and study of the heart fibers by means of the Diffusion Tensor technique". In: *ISSSD* (2020).

[17] S. Angeli et al. "A high-resolution cardiovascular magnetic resonance diffusion tensor map from ex-vivo C57BL/6 murine hearts". In: *Journal of cardiovascular magnetic resonance : official journal of the Society for Cardiovascular Magnetic Resonance* 16.1 (2014). DOI: 10.1186/s12968-014-0077-x.

[18] E. A. Mendiola et al. "Right Ventricular Architectural Remodeling and Functional Adaptation in Pulmonary Hypertension". In: *Circulation. Heart Failure* 16.2 (2023). DOI: 10.1161/CIRCHEARTFAILURE.122.009768.

[19] I. Teh et al. "Mapping cardiac microstructure of rabbit heart in different mechanical states by high resolution diffusion tensor imaging: A proof-of-principle study". In: *Progress in biophysics and molecular biology* 121.2 (2016). DOI: 10.1016/j.pbiomolbio.2016.06.001.

6 APPENDIX

6.1 Macrostructural validation using ventricular patterns

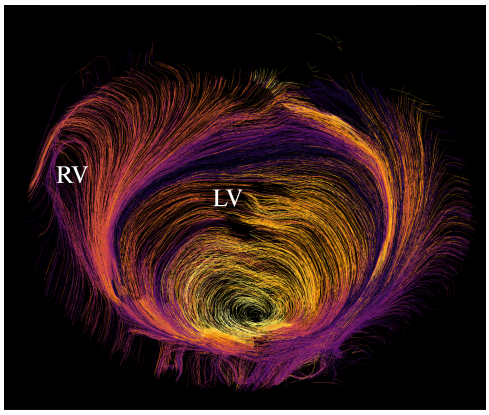


Figure 4: ParaView visualization of the mouse heart showing the characteristic helical structure of the ventricles. LV: left ventricle; RV: right ventricle.

RegiViz - A Tool for Generating and Visualizing Cancer Regimens

Yang Xiang * Brendan Leahey†

Brown University

Abstract

Cancer regimens guide patients through chemotherapy, but existing visualizations lack automation, personalization, and clarity. RegiViz, a web-based tool, automates cancer regimen visualization using a fine-tuned large language model (LLM) trained on the HemOnc database. It features intuitive visual encodings for drug schedules, types, and treatment timelines, with holiday-aware adjustments and patient-doctor feedback integration. User studies show RegiViz improves efficiency and clarity over existing regimen visualizations such as ChemoExperts [1]. RegiViz is available at https://yang2888.github.io/Regimen-demo/.

Keywords: Cancer regimens, oncology, large language models, human-computer interaction, evaluation.

1 Introduction

Cancer regimens consist of various treatment medications, respective doses, routes of administration, and treatment dates. Regimens typically consist of *cycles* of fixed-length drug sequences. Visualizing regimens is one of several ways that oncologists can guide patients through a stressful treatment process. A recent study indicated a gap in patient needs and patient information on chemotherapy side effects, duration of treatment, and general ability to communicate with doctors on miscellaneous aspects of treatment, such as supportive care or other contacts [2]. Our visualization seeks to better address these aspects of patient care through visual encoding such as color, shape, and text, which we evaluate in a user study.

In addition, our collaborators identified that oncologists operate within tight timelines in the clinic, making conventional manual methods of visualizing regimens difficult to use. These collaborators have maintained an extensive database of chemotherapy regimens through HemOnc.org [5]. Leveraging an LLM finetuned on Hemonc's database, our tool enables rapid, automated synthesis of cancer regimens from raw text towards in-clinic deployment.

2 Related Work

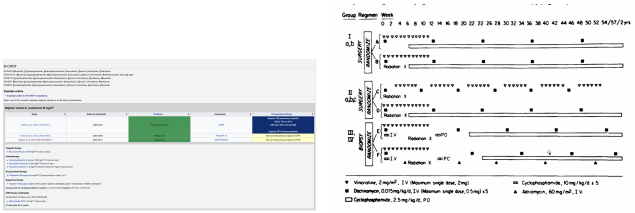
The HemOnc knowledgebase is an invaluable oncology reference for clinicians, offering detailed insights into therapy type, timing, dosage, and cycle length/timing (Figure 1a) [5]. Its backend data tables serve as a great example of structured regimen data.

Dr. Warner of HemOnc has previously automatically synthesized regimen networks representing guidelines founded in randomized clinical trial (RCT) data [4]. They used color, size, and opacity of nodes to represent RCT comparisons of drug regimens, inspiring our automatic synthesis and visual encoding methods.

Conventional regimen visualizations, such as figure 1b provided by our collaborators, are handcrafted or hand-drawn and often preprinted, which requires time or preparation. Further, these visualizations may lack encodings such as color or shape variations, and less flexibly specify fixed cycle dates rather than calendar dates.

*e-mail: yang_xiang@brown.edu

†e-mail: bleahey@cs.brown.edu



(a) HemOnc representation in structured text form, lacking visual encodings [5]

(b) Hand-crafted visualization lacking color encodings and calendar dates

Figure 1: Comparison of baseline regimen representations

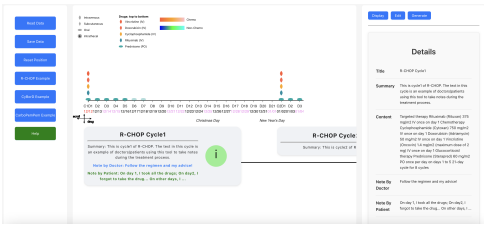


Figure 2: RegiViz contains color, shape, and opacity encoding on a real time axis

ChemoExperts, a freely available online platform, is the state-of-the-art regimen visualizer. It enables manual entry of drug, regimen, radiation, doctor visits, and other life events into an importable calendar-based representation [1]. It also encodes drug routes and treatment locations, providing inspiration for our design.

3 Methodology

3.1 Visualization Methods

We employ D3.js and React to generate regimen visualizations on a webpage. The visualization form, shown in figure 2, represents key clinical elements identified in our introduction.

The horizontal axis represents the treatment timeline, including cycle and calendar dates, cycle start dates, and cycle length. Today's date is colored red. Since patients cannot go to the clinic on holidays and weekends, which are colored pink, we account for this uncertainty through drug "afterimages". Afterimages are transparent copies of the drug on the nearest date the clinic is open.

Other color encodings are used to distinguish drug type. Chemotherapy, generally having more side effects and stigma, is visualized in the red/orange spectrum. Meanwhile, non-chemotherapy drugs are in the blue-green range. In addition, drug shapes specify the route of administration of the drug, such as intravenous (IV) or oral (PO). Additional drug and regimen information such as drug dose and treatment phase may be found in the details tab on the right side.

Other novel visual elements include patient/doctor feedback, which may be inputted using the "Edit" button. This patient-doctor communication functionality is not present in any other existing tools to our knowledge.

3.2 Language Model and Finetuning

Since there is no widely accepted cancer regimen syntax, physicians' descriptions of regimens can widely vary. We use a LLM to

take this variable input and create visualizations based on inputted properties: dose, dose unit, route, time sequence, cycle of regimen, cycle length, cycle unit.

While convenient, LLMs may hallucinate as shown in 4. Thus, we fine-tuned to improve generation accuracy. We created a dataset with 330 regimens: 40 regimen-input pairs, and 290 paper-regimen pairs. User inputs represent potential doctor inputs, collected from our collaborators and existing HemOnc data. Papers are linked from regimen sources in our data, and their content is extracted as text. We used the *GPT-4o-2024-08-06* model for finetuning.

3.3 Evaluation Methods

3.3.1 User Study

We performed a study of one clinical expert and two post-doc researchers. Participants generated a regimen in 10 minutes or less with ChemoExperts and RegiViz, recording time from regimen name entry to completion. Then, they evaluated the information accuracy and visualization effectiveness through a survey.

Literature on Empirical Studies on Information Visualization specifies a framework for evaluating communication through visualization [3]. We applied this framework when constructing questions to assess the efficiency and visualization effectiveness of our tool.

Clinical effectiveness and accuracy was also evaluated based on criteria from existing user studies and collaborator input [2]. For example, when assessing efficiency, we chose 5 and 10 minutes as our threshold—collaborators specified this as the typical time with a patient in clinic.

3.3.2 Metrics

We evaluate the performance of our LLM with accuracy, precision and recall. Low precision indicates incorrectly identified drugs, and low recall indicates missing drugs. Categorical properties like dose and route are represented by accuracy.

4 Results

4.1 User Study Results

All participants were able to create regimens with both tools. Our tool proved more efficient, with 2/3 users generating regimens in < 5 mins and 1/3 in 5-10 mins on our tool, while the baseline tool took > 10 mins for 2/3 users and 5-10 mins for 1/3 users.

Among the three users, visual appeal compared to the baseline was rated 3.67/5 on average. Color usage was rated 2 by the first user, but an average 4.5/5 by the remaining users with access to the color legend. Specific comparisons of visual elements and effectiveness can be found in figure 3.

4.2 Experimental Results

Since this is the first tool for LLM generation of regimen representations, we compare our fine-tuned model to the base GPT-4o.

Our results show that both models perform well, producing correct generations on user input, which is relatively short and contains little redundant info. Meanwhile, drug identification performance declines on paper input. For example, the LLM may identify 'placebo' as a drug used in the regimen.

Our fine-tuned model performs better across nearly all metrics. It enhances the accuracy of sequence timing and cycle length determination. However, it has a worse recall for the paper dataset, possibly over-fitting to the training set and ignoring certain info.

5 Discussion + Conclusions

5.1 Conclusions

We presented the first automated cancer regimen visualization tool. We demonstrated more clinically accurate, efficient, and patient-friendly interface for regimen visualization. Based on our user study results, this tool will help fill an important gap in cancer patient care delivery by helping patients understand complex drug regimens. Finally, we proposed fine-tuning as a method of improving accuracy of automatic generation, backed by promising experimental results.

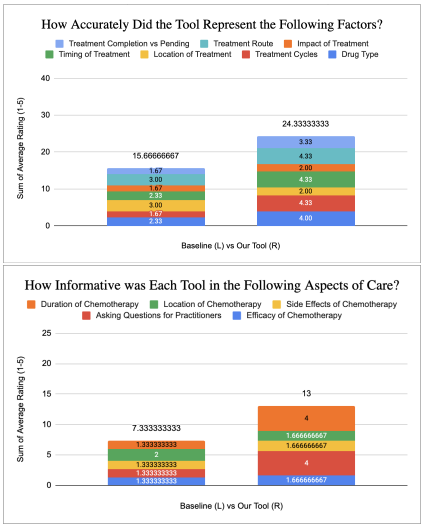


Figure 3: Comparison of Baseline and RegiViz shows that our tool was generally more accurate and informative than the baseline. RegiViz notably outperformed the baseline in accuracy of treatment cycles and timing, as well as informativeness of chemotherapy duration and asking practitioners questions.

Metric	Dataset	Baseline (GPT-4o)	Fine-tuned GPT-4o
Precision	Input-dataset	100%	100%
	Paper-dataset	45.57%	58.13%
Recall	Input-dataset	94.87%	97.44%
	Paper-dataset	78.15%	76.47%

Table 1: Accuracy of identifying correct drugs. The finetuned model improves performance in precision for both datasets but lowers recall for the paper-dataset.

Accuracy	Dose	Route	Timing_seq	Cycle UB	Cycle Length
Baseline	97.29%	86.49%	64.86%	83.33%	58.33%
Finetuned	97.37%	94.74%	83.78%	100%	91.67%

Table 2: Accuracy for dose, route, timing_seq, cycle_ub, cycle_length of 2 models. Both models perform well in identifying dose, route. Finetuned model improves timing_seq and cycle_length greatly.

5.2 Open Problems

Through survey feedback and discussion with our collaborators, we have identified several open problems.

Evaluating our LLM with safety-specific metrics can demonstrate greater in-clinic utility.

Incorporating solid cancer training data and visual encodings would increase the applicability of browser tools. Elements like radiation cleanly fit into drug route representations, while more volatile treatments (e.g. surgery) mandate design changes.

Encoding treatment duration is novel and clinically applicable. For example, some IV treatments can stretch across multiple days, surprising patients.

User study results demonstrate room for UI/UX improvements, visual appeal, and display of clinical information like side effects and efficacy of chemotherapy. Encoding and automatic synthesis of clinical factors may improve the personalization of our tool.

In-clinic cancer patient experience studies are a promising direction for visualization and informativeness evaluation.

Acknowledgements

The authors wish to thank their collaborators at HemOnc and Brown Alpert Medical School, especially Sandeep Jain, Jeremy Warner, and Sanjay Mishra. We also thank our user study participants for their valuable feedback. Finally, we thank Professor David Laidlaw for overseeing and guiding us through the course that led to the development of this project.

References

[1] ChemoExperts Foundation, Inc. Chemoexperts treatment tracker®, 2024. Retrieved October 14, 2024.

[2] G. P. Chua, H. K. Tan, and M. Gandhi. What information do cancer patients want and how well are their needs being met? *Ecancermedicalscience*, 12:873, 2018.

[3] H. Lam, E. Bertini, P. Isenberg, C. Plaisant, and S. Carpendale. Empirical studies in information visualization: Seven scenarios. *IEEE Transactions on Visualization and Computer Graphics*, 18(9):1520–1536, 2012.

[4] J. Warner, P. Yang, and G. Alterovitz. Automated synthesis and visualization of a chemotherapy treatment regimen network. *Studies in Health Technology and Informatics*, 192:62–66, 2013.

[5] J. L. Warner, D. Dymshyts, C. G. Reich, M. J. Gurley, H. Hochheiser, Z. H. Moldwin, R. Belenkaya, A. E. Williams, and P. C. Yang. Hemonc: A new standard vocabulary for chemotherapy regimen representation in the omop common data model. *Journal of Biomedical Informatics*, 96:103239, 2019.

6 Yang Xiang - Individual Contributions

6.1 Intellectual Contributions

- Discussed regimen generation and optimization in meetings with collaborators
- Designed form of datasets based on Hemonc database
- Designed part of the visualization forms

6.2 Technical Contributions

- Designed frontend framework, including website construction and implementation of dateline, regimen tree, different panels
- Implemented backend data pipeline, including server backend apis and the baseline to convert input to structured regimen info
- Trained model after gathering and processing datasets
- Deployed the web app frontend at github page and the backend at render.com

7 Brendan Leahey - Individual Contributions

7.1 Intellectual Contributions

- Discussed clinical UX multiple times a week with Sandeep Jain
- Designed user study
- Researched baseline comparison methods
- Conceptualized timeline visualization and visual encoding forms building on calendar-based representations planned in my proposal

7.2 Technical Contributions

- Implemented frontend framework in collaboration with Yang
- Validated and debugged JSON and DateLine outputs
- Developed drug and date encodings/legend and novel visualization methods such as uncertainty visualization
- Developed supplementary visualizations, user study, and tool walk-through
- Wrote and formatted the majority of text deliverables such as presentations and papers