Boxcars on Potatoes: Exploring the Design Language for Tangible Visualizations of Scalar Data Fields on 3D Surfaces

Bridger Herman
Interactive Visualization Lab
University of Minnesota
herma582@umn.edu

Daniel F. Keefe
Interactive Visualization Lab
University of Minnesota
dfk@umn.edu

Abstract: We present a design-based exploration of the potential to reinterpret glyph-based visualization of scalar fields on 3D surfaces, a traditional scientific visualization technique, as a data physicalization technique. Even with the best virtual reality displays, users often struggle to correctly interpret spatial relationships in 3D datasets; thus, we are motivated to understand the extent to which traditional scientific visualization methods can translate to physical media where users may simultaneously leverage their visual systems and tactile senses to, in theory, better understand and connect with the data of interest. This pictorial traces the process of our design for a specific user study experiment: (1) inspiration, (2) exploring the data physicalization design space, (3) prototyping with 3D printing, (4) applying the techniques to different synthetic datasets. We call our most recent and compelling visual/tactile design *boxcars on potatoes*, and the next step in the research is to run a user-based evaluation to elucidate how this design compares to several of the others pictured here.

Why Glyphs on Surfaces?

Glyphs have often been used as a form of visualization to depict complex data in a concise, understandable manner [1]. These symbols are used because they are relatively easy for humans to interpret and learn. This can partially be attributed to the fact that glyph-based visualizations can be processed in parallel. This is often known to the visualization community as “pre-attentive processing,” although cognitive scientists usually prefer the term “efficient search” [2]. This means that conscious effort is not required in order for viewers of the glyph-based visualization to make sense of it and discern patterns [3].

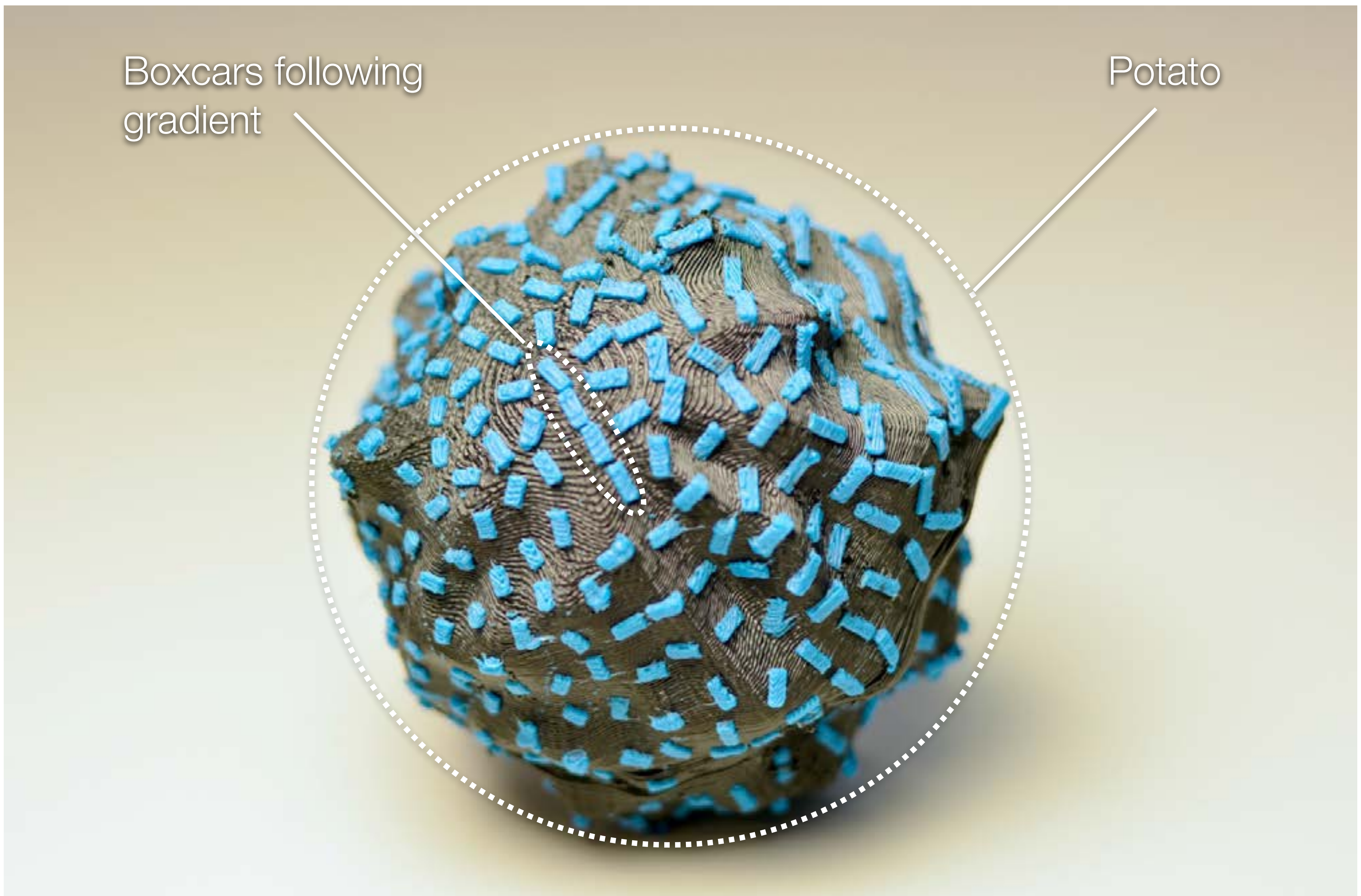
Physical 3D surfaces alone have already been proven to be extremely useful as a visualization tool. For instance, Djavaherpour et al. demonstrated the utility of a 3D printed physicalization of the Earth that can be extended with physical data layers [4]. The authors speculate that this type of physicalization could also be used by people with visual impairments if the right size and shape glyphs are used (in a style similar to tactile thematic maps [4]).

Virtual glyphs on 3D surfaces are common practice in the scientific visualization community. Sometimes the surface glyphs will be represented as images [6], while other times they can be represented as small 3D objects [7]. In our approach, a glyph is considered to be a small 3D object printed upon a physical surface. We hypothesize that, with the correct visual language, physical glyphs on surfaces that can be picked up and manipulated by hand will outperform their virtual counterparts in situations where users must interpret both the data represented via glyphs ***and*** the underlying surface, provided that this surface is sufficiently complex so as to be difficult to understand using traditional computer graphics (e.g., present-day VR/AR displays).

Why Physicalization?

Physicalization has taken a new turn in the past decade with the growing availability of affordable 3D printing, and has recently been used for creating tactile models of surfaces representing data [8]. Current virtual data representations lack one of the most important human senses – touch. We're motivated to explore an alternative form of visualization that makes it possible to explore three-dimensional surfaces and scalar data fields via physicalizations that can be held in the hands.

Where might this be useful? One of the driving current scientific visualization problems studied in our lab is simulated blood flow in the heart. Here, scientists must analyze scalar fields, such as pressure and stress magnitude relative to the shape of the heart wall. Abstracting from this problem to come up with a more general visualization and task that is appropriate for user study, we arrived at a "bumpy potato." Like scientists analyzing the heart, the task we plan to study requires users to interpret the shape of the surface of the potato, similar to the shape of the heart walls, ***and*** the glyphs on the surface, which depict a scalar data field, similar to pressure or stress on the heart walls. Thus, our exploration of the design space has focused on identifying legible physical glyphs that do not obstruct (and perhaps even enhance) legibility of the underlying 3D surface.

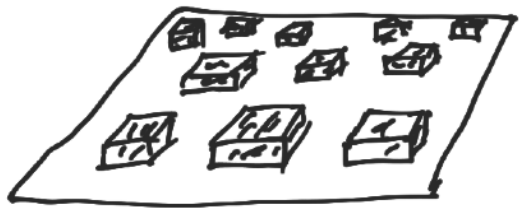


Ideation Process

Glyph Shape

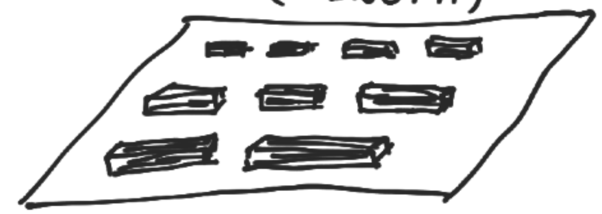
SPHERES

CUBES



BOXES (HEIGHT)

BOXES (LENGTH)



Glyph Alignment

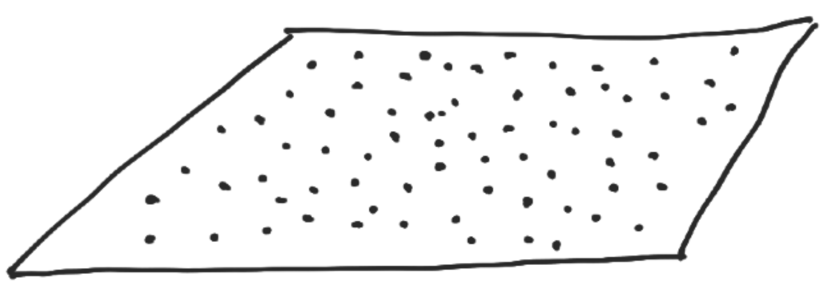
AXIS-ALIGNED

WITH NORMAL

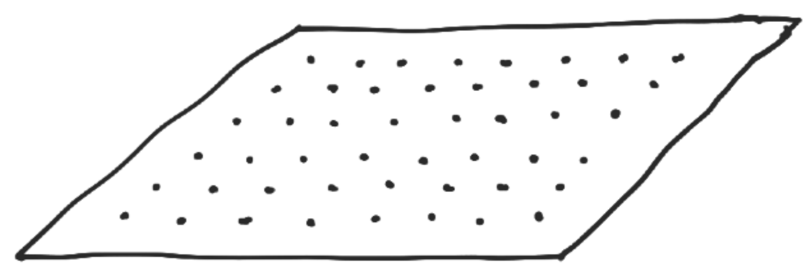
NORMAL AND GRADIENT

Glyph Distribution

POISSON-DISC



GRID



Fabrication Process

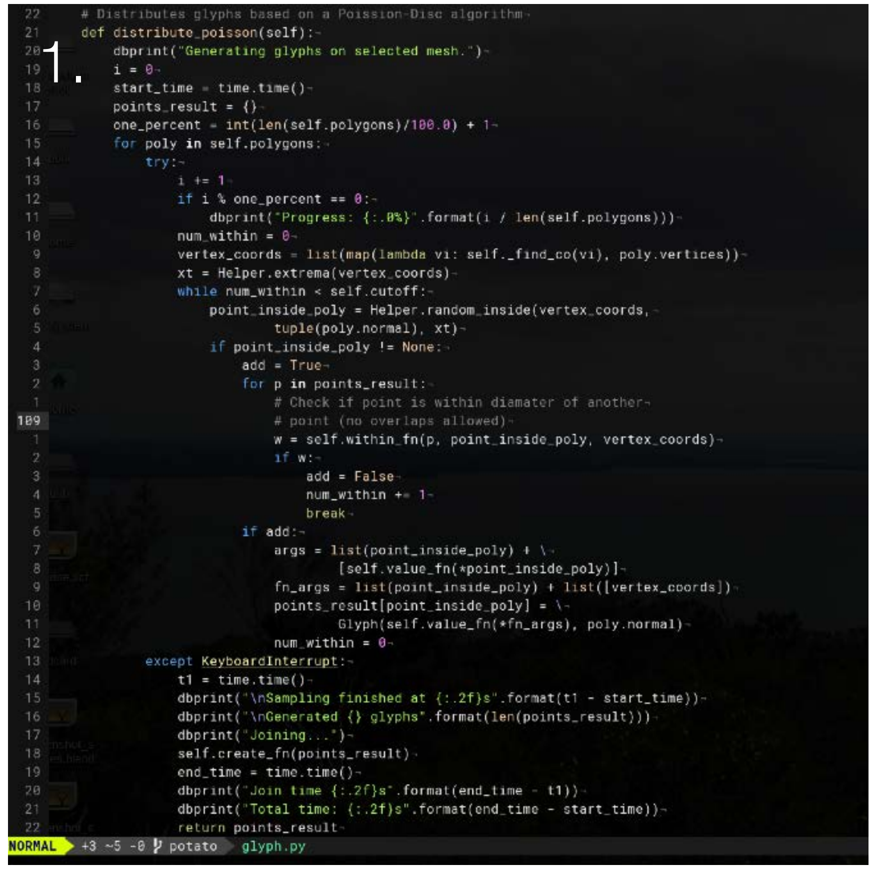
1. Blender scripts* to generate 3D meshes from synthetic surface data and place glyphs on top based on a synthetic scalar data field. Synthetic data are used because we wish to develop a range of datasets with known properties for a user study.

2. Models generated in Blender ready for 3D printing. Each potato is printed in two halves.

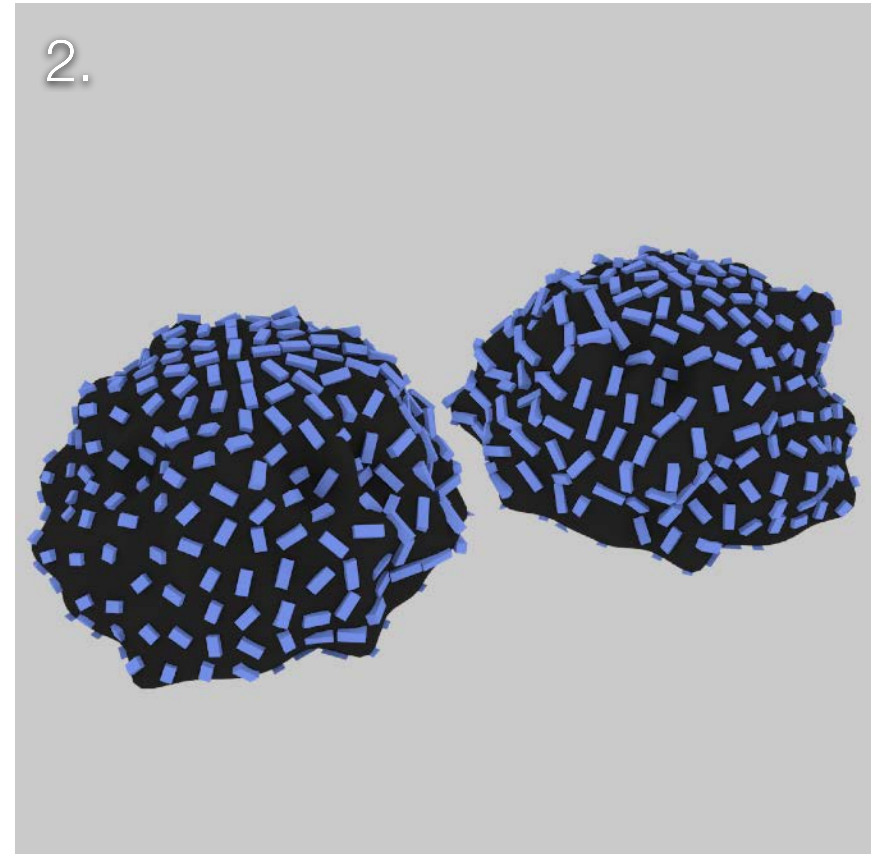
3. 3D printing using a MakerBot printer with two colors.

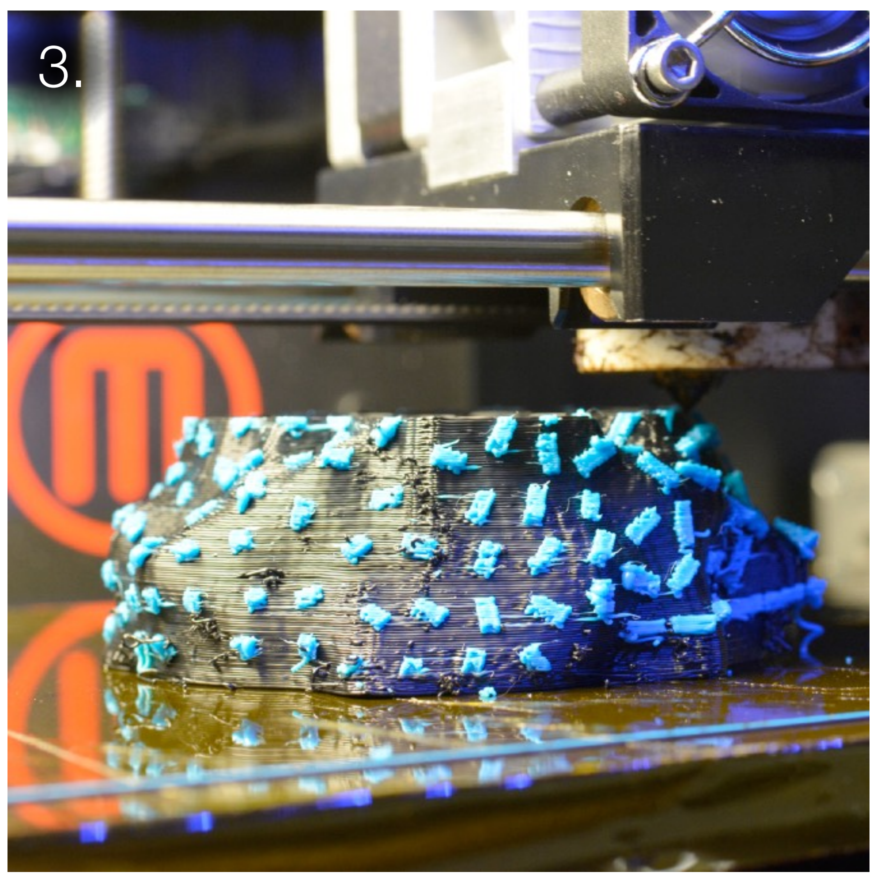
4. After dusting off any 3D printing artifacts and securing the two halves together.

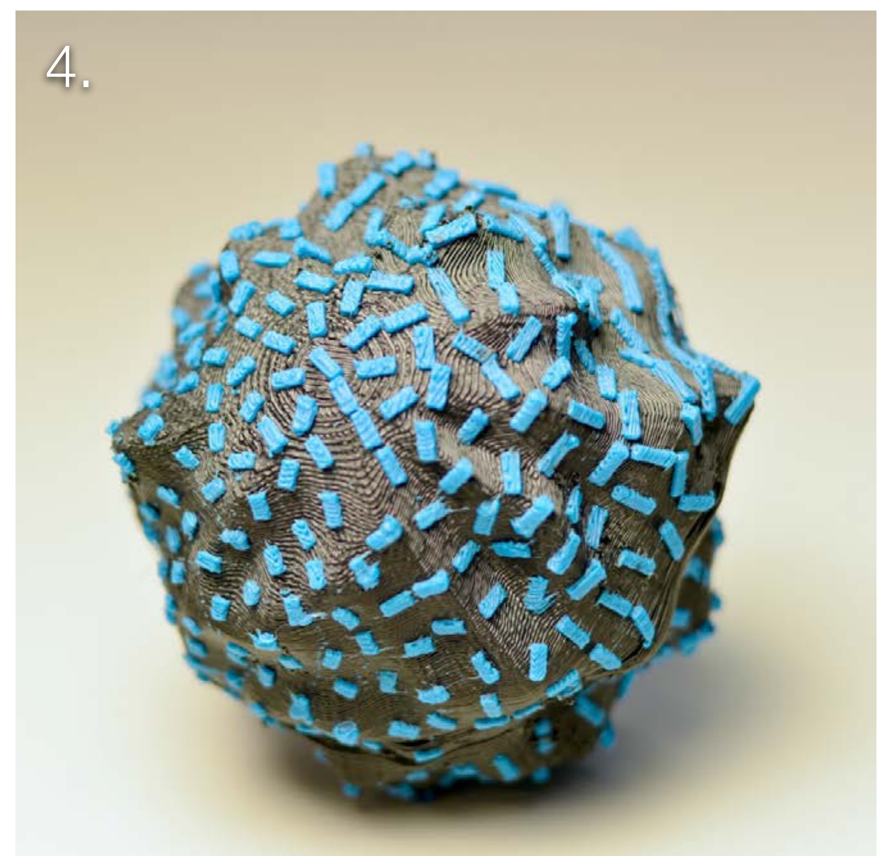
*Code used to generate the models can be found at https://github.com/ivlab/BoxcarPotatoes


1.

```
    # Distributes glyphs based on a Poission-Disc algorithm
    def distribute_poisson(self):
        dbprint("Generating glyphs on selected mesh.")
        i = 0
        start_time = time.time()
        points_result = {}
        one_percent = int(len(self.polygons)/100.0) + 1
        for poly in self.polygons:
            try:
                i += 1
                if i % one_percent == 0:
                    dbprint("Progress: {:.0%}".format(i / len(self.polygons)))
                num_within = 0
                vertex_coords = list(map(lambda vi: self._find_co(vi), poly.vertices))
                xt = Helper.extrema(vertex_coords)
                while num_within < self.cutoff:
                    point_inside_poly = Helper.random_inside(vertex_coords,
                            tuple(poly.normal), xt)
                    if point_inside_poly != None:
                        add = True
                        for p in points_result:
                            # Check if point is within diamater of another
                            # point (no overlaps allowed)
                            w = self.within_fn(p, point_inside_poly, vertex_coords)
                            if w:
                                add = False
                                num_within += 1
                                break
                        if add:
                            args = list(point_inside_poly) + \
                                    [self.value_fn(*point_inside_poly)]
                            fn_args = list(point_inside_poly) + list([vertex_coords])
                            points_result[point_inside_poly] = \
                                    Glyph(self.value_fn(*fn_args), poly.normal)
                            num_within = 0
            except KeyboardInterrupt:
                t1 = time.time()
                dbprint("\nSampling finished at {:.2f}s".format(t1 - start_time))
                dbprint("\nGenerated {} glyphs".format(len(points_result)))
                dbprint("Joining...")
                self.create_fn(points_result)
                end_time = time.time()
                dbprint("Join time {:.2f}s".format(end_time - t1))
                dbprint("Total time: {:.2f}s".format(end_time - start_time))
                return points_result
NORMAL  +3 ~5 -0  potato  glyph.py
```

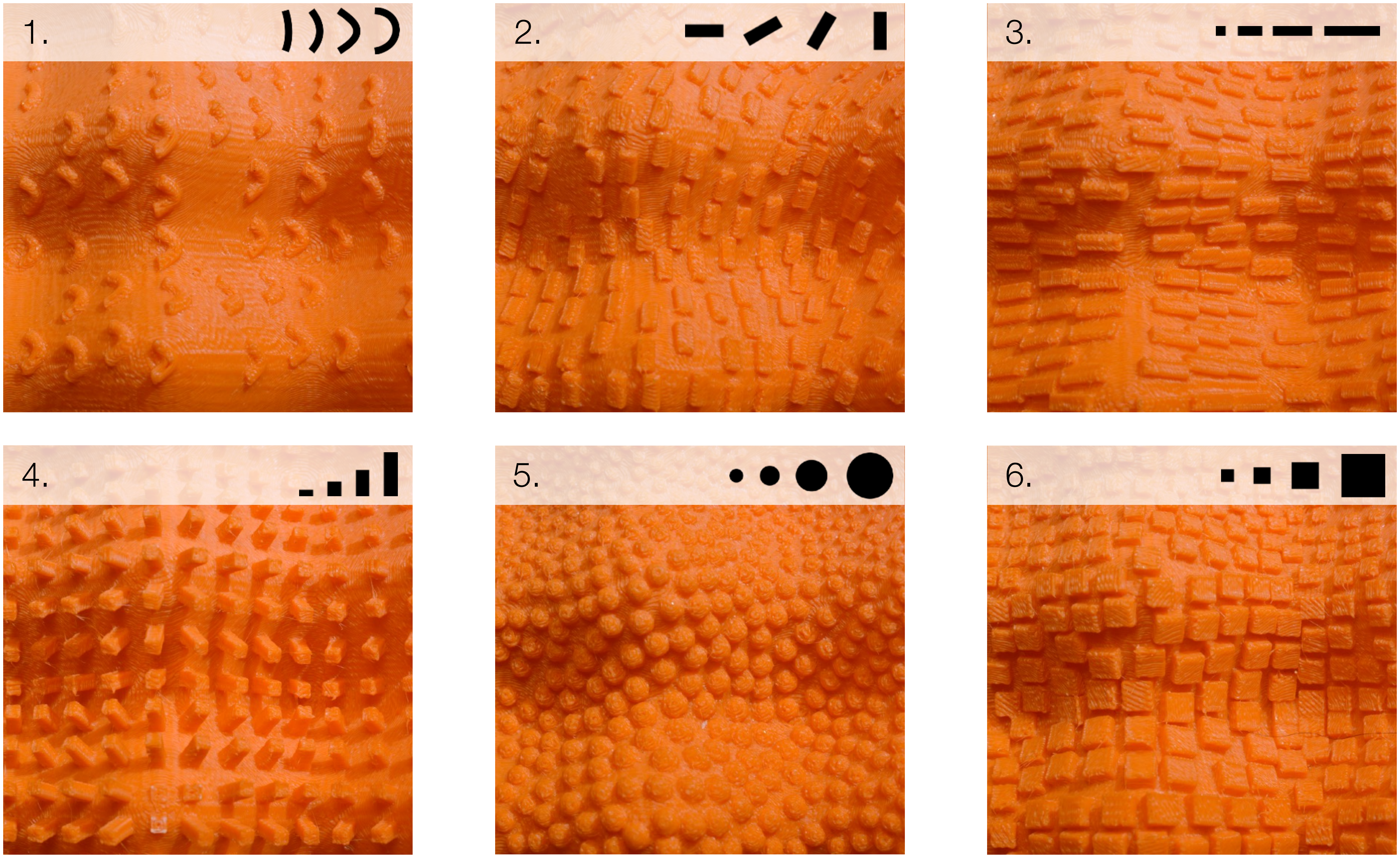

2.


3.


4.

Experimentation with Glyph Shapes

One of the most intuitive ways to represent glyphs is with simple shapes. While many multidimensional datasets require more complex glyph shapes that vary with the data they are representing, the most understandable glyphs are usually primitive shapes like spheres and cubes [9]. We explored several different glyph types, following the guide to visual channel effectiveness posited by Munzner [10]. Visual channels such as curvature (1), angle (2), length (3), height (4), and area (5, 6) were tested on spherical and box-shaped glyphs. Ultimately, the current design uses the length of boxes to represent the data, because boxes reflect the normal and gradient of the underlying surface, whereas the other glyph types listed below do not. Additionally, when examining surfaces with spherical glyphs, some people experienced discomfort. This may have been triggered by trypophobia, the fear of holes [11].

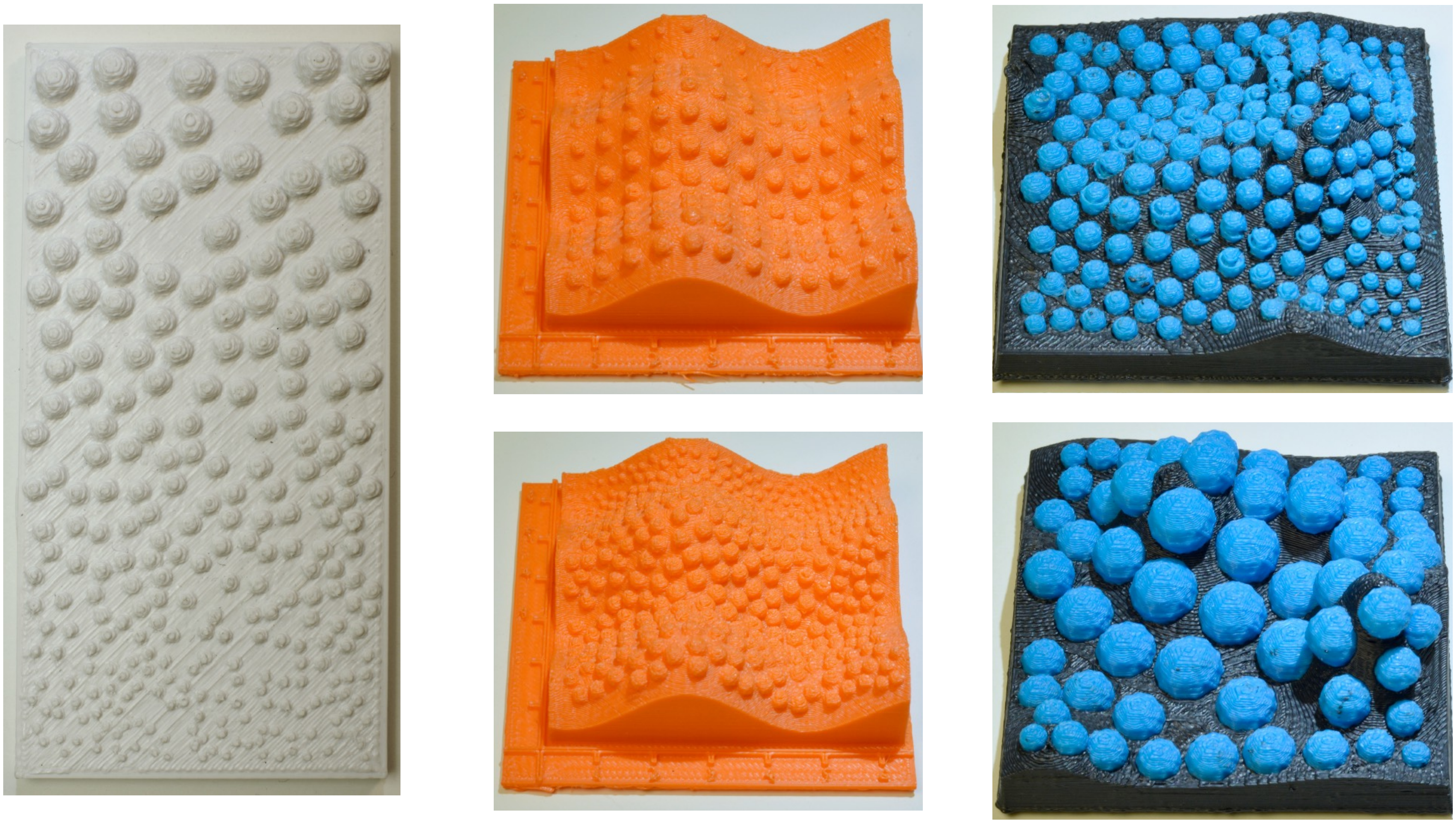


Experimentation with Glyph Sizes

Initially, data was directly mapped onto glyph radius. However, this direct mapping can be misleading, because radius is not necessarily perceived linearly [1]. It turns out that this is especially true with physicalizations, as shown by Jansen and Hornbæk [14]. Additionally, it was shown that certain sizes of glyph tend be the easiest to perceive.

For 2D scatter plots, glyphs should have visual angles between 0.072° and 0.573° [15]. Interpreting these guidelines at a comfortable examination distance of 25cm, our final glyphs range between 0.314mm and 2.5mm.

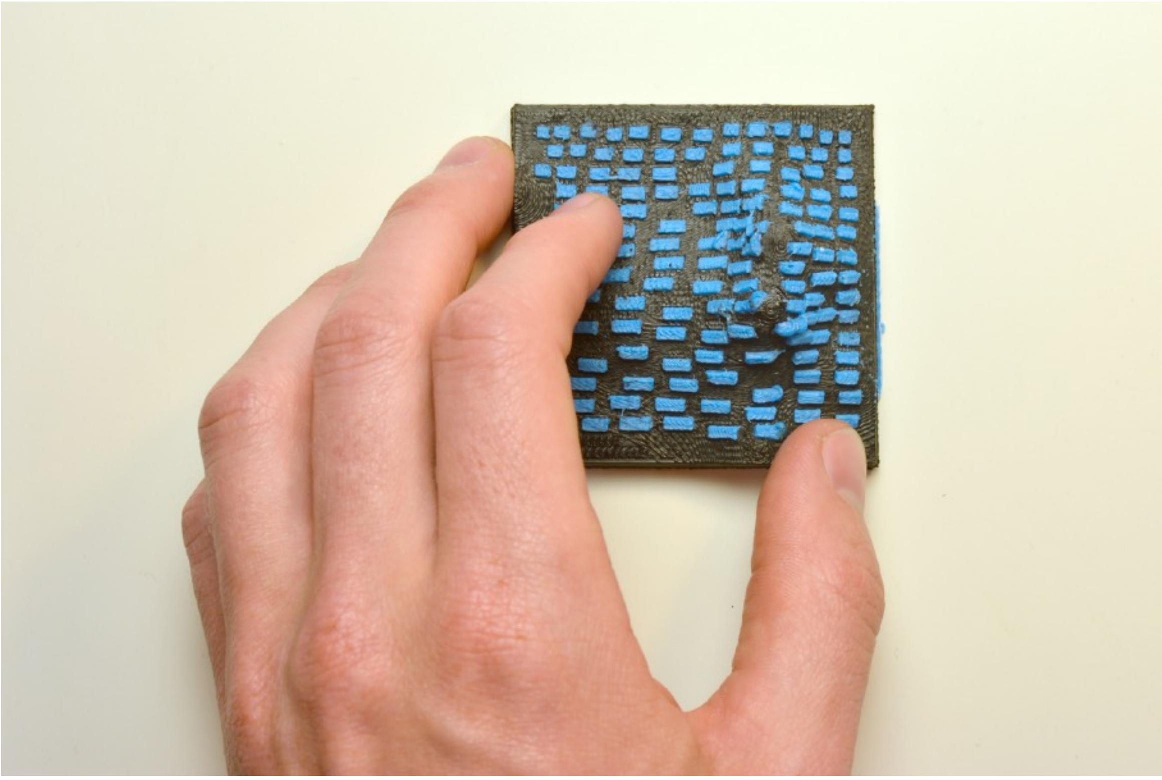
Experiments with Spherical Glyph Sizes



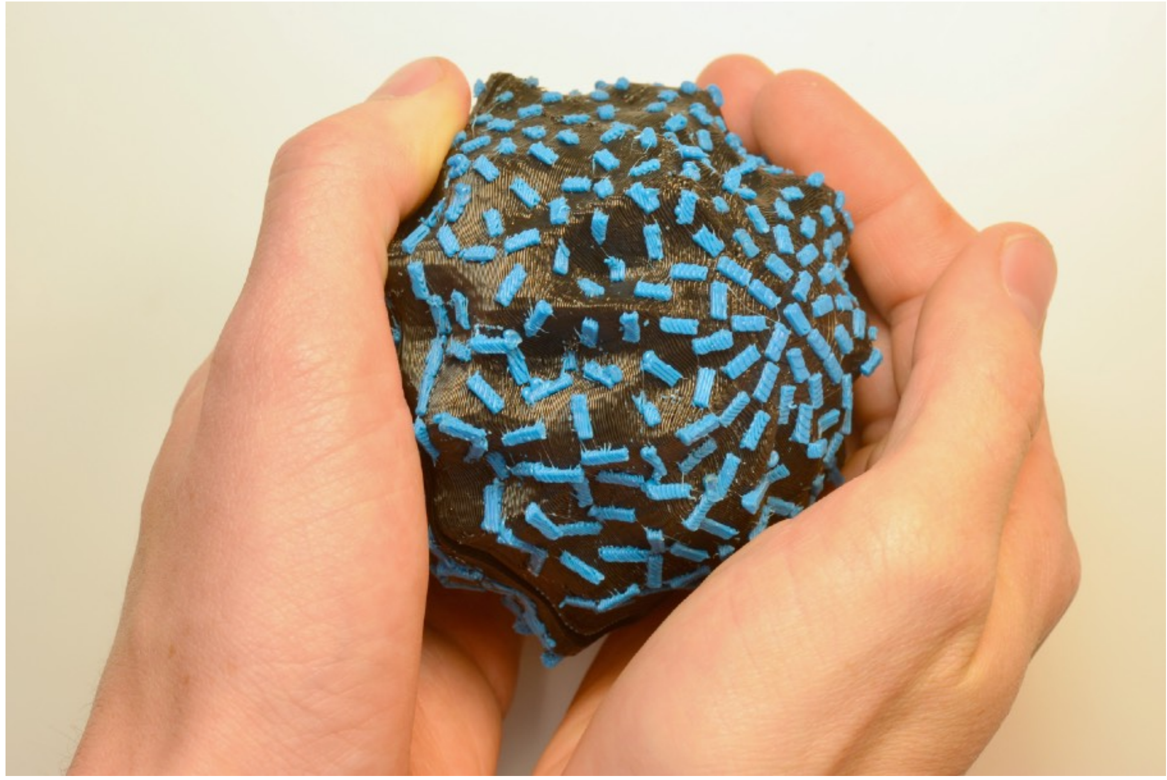
Refining the Surface

The design process leading up to the present *boxcars on potatoes* required several iterations. As an initial proof-of-concept, glyphs were first printed on simple planar (1) and quadratic (2) surfaces. These surfaces ended up being fairly uninteresting, because they did not feature any concavities. The next phase was a sinusoidal surface (3), which sometimes caused glyphs to intersect, prompting further glyph design work. For the penultimate iteration, glyphs were placed upon a randomly generated sum-of-Gaussians surface (4).

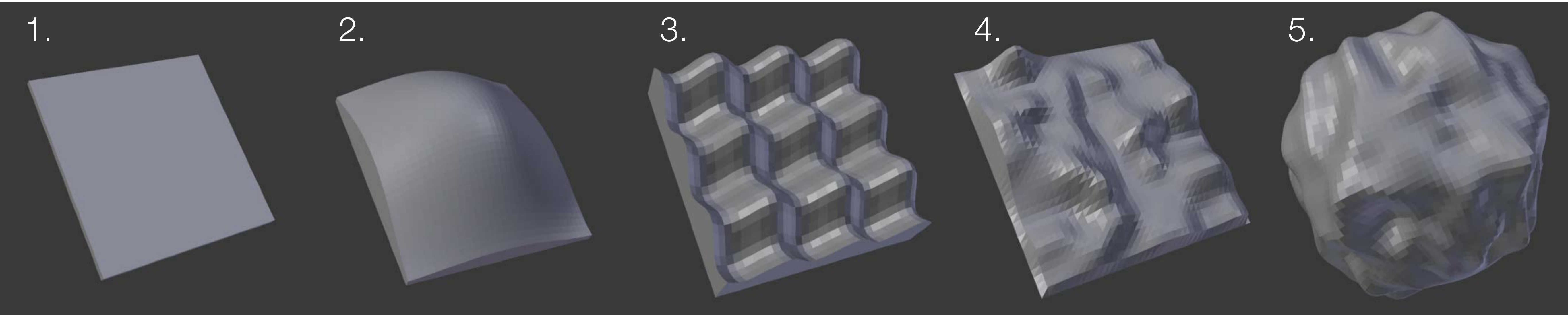
However, all of the aforementioned surfaces are essentially just 3D height maps, which don't necessarily encourage users to pick them up and examine them. With the goal of making the objects more appealing for users to pick up, we turned to a spheroid (5) as the final design. The spheroids (*potatoes*) were inspired by the randomly-generated blobs used in a virtual reality perceptual study titled "The Great Potato Search" [12], from which this paper takes part of its name.



Users were content to leave the flatter surfaces on the table while examining them.



Users felt compelled to pick up the bumpy potato-shaped objects while examining them.

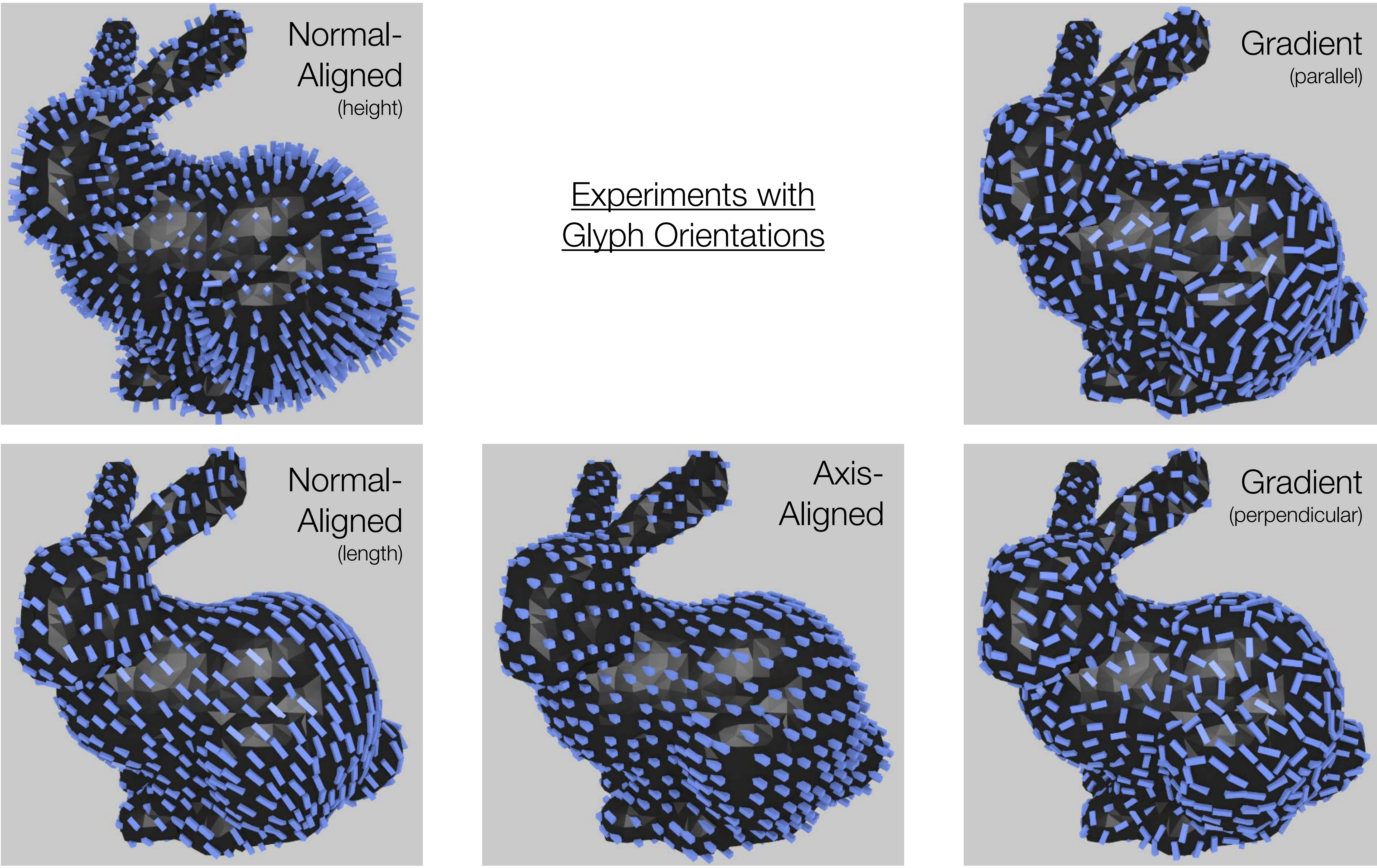


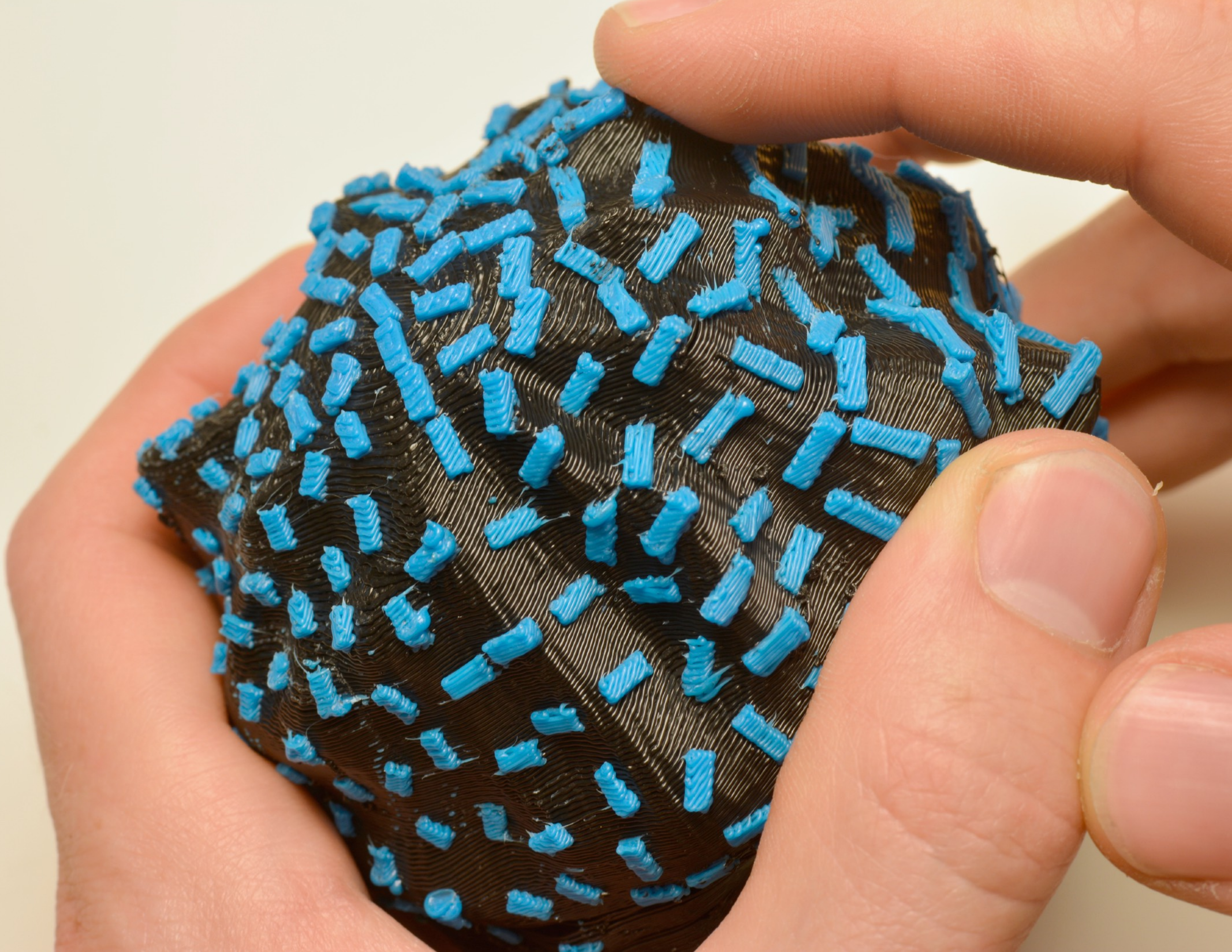
Refining Glyph Orientation

Out of simplicity, the first alignment method used during the course of this project was axis-aligned glyphs. These glyphs look fine when applied to flat, axis-aligned surfaces, but when the surface has any curvature or slant, this technique fails spectacularly.

Next, glyphs were rotated to relay the surface normal at the location where they are placed. This technique may actually enhance understanding of the underlying surface data.

For the final *boxcar* glyphs, the surface gradient was used in conjunction with the normal. A technique inspired by Kim et al. was used to compute the gradient in the principal direction [13]. Using a similar approach, glyphs can also be placed so they are rotated perpendicular to the gradient.





Outlook and Conclusions

Our medical collaborators already use 3D printing to better understand anatomical surfaces, such as the patient-specific shape of the heart, but the utility of these prints is limited relative to VR visualizations because only the surface is physicalized. We foresee that as 3D printers become even more common, printing additional data on top of these surfaces will prove to be an insightful and engaging addition to the visualization toolbox. To reach this point, we must first understand the design language for this new form of data physicalization.

The next step in our process is to design and conduct a formal user study. However, in this work-in-progress, we can already state several findings that will be relevant to the visualization community. Firstly, some styles of glyphs plainly didn't work for this application — namely, the curvature- and angle-based styles. Through informal user trials with lab members, we determined that box-shaped glyphs performed the best. We hypothesize that this is due to the fact that these shapes reflect additional information about the surface underneath them. We also found that the gradient-based orientation of glyphs worked well, for the same reason. To our surprise, one user had a serious aversion to the spherical glyphs; additional research revealed that this may be triggered by trypophobia.

We also found that in order to take full advantage of this form of physicalization, the underlying surface must be complex enough to compel users to pick up the visualization, touch it, and rotate it in their hands for closer inspection. It wasn't until the physical *potatoes* were printed that users felt the need to pick up the model and examine it — with the rest, they were content to let them sit on the table.

Acknowledgments

This work was supported in part by the National Science Foundation (IIS-1251069, IIS-1704604, IIS-1704904) and the National Institutes of Health (1R01EB018205-01)

References

[1] R. Borgo, J. Kehrer, D. H. Chung, E. Maguire, R. S. Laramee, H. Hauser, M. Ward, and M. Chen, "Glyph-Based Visualization: Foundations, Design Guidelines, Techniques and Applications.," in *Eurographics (STARs)*, pp. 39–63, 2013.

[2] J. Duncan and G. W. Humphreys, "Visual Search and Stimulus Similarity.," in *Psychological Review*, pp. 433–458, 1989.

[3] C. Healey and J. Enns, "Attention and Visual Memory in Visualization and Computer Graphics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 7, pp. 1170–1188, 2012.

[4] H. Djavaherpour, A. Mahdavi-Amiri, and F. F. Samavati, "Physical Visualization of Geospatial Datasets," *IEEE Computer Graphics and Applications*, vol. 38, no. 3, pp. 61–69, 2017.

[4] M. M. Lawrence and A. K. Lobben, "The Design of Tactile Thematic Symbols.," *Journal of Visual Impairment & Blindness*, vol. 105, no. 10, 2011.

[6] A. Rocha, U. Alim, J. D. Silva, and M. C. Sousa, "Decal-Maps: Real-time Layering of Decals on Surfaces for Multivariate Visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 821–830, 2017.

[7] T. Ropinski, M. Specht, J. Meyer-Spradow, K. H. Hinrichs, and B. Preim, "Surface Glyphs for Visualizing Multimodal Volume Data.," in *VMV*, pp. 3–12, 2007.

[8] M. C. Thrun and F. Lerch, "Visualization and 3D Printing of Multivariate Data of Biomarkers," 2016.

[9] A.E.Lie, J. Kehrer, and H.Hauser,"Critical Design and Realization Aspects of Glyph-Based 3D Data Visualization," in *Proceedings of the 25th Spring Conference on Computer Graphics*, pp. 19–26, ACM, 2009.

[10] T. Munzner, Visualization Analysis and Design. AK Peters/CRC Press, 2014.

[11] G. G. Cole and A. J. Wilkins, "Fear of Holes," *Psychological Science*, vol. 24, no. 10, pp. 1980–1985, 2013.

[12] C. D. Jackson, D. B. Karelitz, D. H. Laidlaw, and S. A. Cannella, "The Great Potato Search: The Effects of Visual Context on Users' Feature Search and Recognition Abilities in an IVR Scene," Interactive Poster.

[13] S. Kim, H. Hagh-Shenas, and V. Interrante, "Showing Shape with Texture: Two Directions Seem Better Than One," in *Human Vision and Electronic Imaging VIII*, vol. 5007, pp. 332–340, International Society for Optics and Photonics, 2003.

[14] Y. Jansen and K. Hornbæk, "A Psychophysical Investigation of Size as a Physical Variable.," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 1, pp. 479–488, 2016.

[15] J. Li, J.-B. Martens, and J. J. van Wijk, "A Model of Symbol Size Discrimination in Scatterplots," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2553–2562, ACM, 2010.