Why We Use Bad Color Maps and What You Can Do About It

Kenneth Moreland; Sandia National Laboratories; Albuquerque, New Mexico, USA

Abstract

We know the rainbow color map is terrible, and it is emphatically reviled by the visualization community, yet its use continues to persist. Why do we continue to use a this perceptual encoding with so many known flaws? Instead of focusing on why we should not use rainbow colors, this position statement explores the rational for why we do pick these colors despite their flaws. Often the decision is influenced by a lack of knowledge, but even experts that know better sometimes choose poorly. A larger issue is the expedience that we have inadvertently made the rainbow color map become. Knowing why the rainbow color map is used will help us move away from it. Education is good, but clearly not sufficient. We gain traction by making sensible color alternatives more convenient. It is not feasible to force a color map on users. Our goal is to supplant the rainbow color map as a common standard, and we will find that even those wedded to it will migrate away.

Introduction

A pervasive technique in scientific visualization called pseudocoloring is to apply colors that vary based on some numerical variable to an object. Pseudocoloring requires defining a function or map from numerical values to colors. A color map is typically defined by selecting a continuum of colors that map linearly to a range of numeric values. Figure 1 shows a simple example of a color map where numeric values between -1 and 0 map to blue colors of varying brightness and numeric values between 0 and 1 map to red colors.

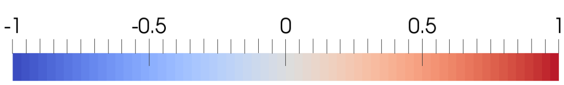

Figure 1. *A simple example of a color map.*

The efficacy of a pseudocolor visualization is contingent on the ability of a human observer to translate the colors back into the numeric values they represent. The choice of colors used in a pseudocolor map can have a major impact on this inverse translation. Consequently, much research has focused on the perception of color and its effect on the visual display of data [6, 17, 20, 23, 24]. As one might expect, the color choice can have a dramatic impact on a viewer's performance in interpreting colors as numbers, and many effective color sets have been designed for this purpose.

With this rich understanding of color perception, one might think that modern visualizations make effective use of color. Unfortunately, many visualizations today use color sets that are known to be extremely problematic. In particular the rainbow color map, so called for its use of the spectrum of colors in the rainbow, is pervasively used in visualization despite the copious evidence that it performs poorly [5, 12, 17, 19, 20, 25].

This paper is a retrospective on why these bad colors are so commonly chosen for visualization and is a position statement on what we as practitioners can do to best promote good color use.

Problems with the Rainbow

Figure 2. *The rainbow color map. Know thy enemy.*

Although there are lots of ways to misuse color, this paper focuses most specifically on using rainbow colors like those shown in Figure 2 for scientific visualization. This is because the rainbow color map is extremely well studied and known to be a very poor representation of data, yet rainbow colors are still used frequently in scientific visualizations.

There are several good publications describing the problems with the rainbow color map [5, 12, 19], so we do not do a thorough analysis here. Instead, this paper gives a very brief overview of the rainbow color map's problems, which fall into three categories: unnatural ordering, irregular perception, and sensitivities to color deficiencies.

The first problem is that the rainbow colors do not follow any natural perceived ordering. Although the order of the hues can be learned, there is no innate sense of higher or lower [25].

The second problem is that the perceptual changes in the rainbow colors are not uniform. The colors appear to change much faster in the yellow region than the green region. This can both obfuscate the data with artifacts that are not in the data and hide important features that are in the data [5].

The third problem with the rainbow color map is that it is sensitive to deficiencies in vision. Although normal human vision can distinguish all of the rainbow's colors, roughly 5% of the population has deficiencies in distinguishing these colors (usually between green and red). These viewers will misinterpret much of the color map [12].

Why We Use Bad Colors

Research shows that subjects tend to overestimate their ability to interpret rainbow colors [4]. That is, users think they are interpreting rainbow colors better than alternatives even though they are in fact doing worse. This is certainly a contributing factor to the proliferation of the rainbow color map, but not the entire reason. Why else would the rainbow color map be so profuse in publications by the visualization community itself, a community that should know better [5]?

Simplicity

One explanation is the sheer simplicity of creating the rainbow color map. Nearly all color selections in computer graphics interfaces are done with RGB (red-green-blue) channels. RGB is a natural choice in computer graphics because the triplet values match the intensity of red, green, and blue light mixed in most display peripherals. An unintended consequence of the RGB color space is that one of the easiest color continua to make is an interpolation between different combinations of fully active and fully inactive channels, which are in fact the rainbow colors. Many computer graphics interfaces also allow colors chosen with HSV (hue-saturation-value) channels. The HSV color space makes rainbow colors even easier: hold saturation and value at maximum while varying hue.

With the simplicity of creating a rainbow spectrum of colors combined with the inclination to accept the colors as a good representation, it is no wonder that the rainbow colors are often the first implemented as a visualization package gets built and are likely to become the default. These initial poor color choices quickly become ingrained in the software as regression testing and backward compatibility must be maintained. Further software layers continue to accept this default and software applications are likely to expose the rainbow color map as its default choice for end users. Few users will have either the knowledge or the inclination to change the default colors used, and thus the rainbow color map becomes featured throughout visualizations.

Aesthetics

However, simplicity is not the only reason rainbow colors are so widely used. If that were the case, simply making a better alternative would eradicate the use of bad colors. But this, at least anecdotally, is shown not to be the case. Consider, for example, bug number 7024 for the ParaView scientific visualization application.[1] A user raised this bug soon after the default color map in ParaView was changed from the rainbow color map to the map shown in Figure 1, which is designed to be reasonably similar to the rainbow color map it supplants but with better perceptual characteristics [13]. Despite this movement to make the better color map easier, this bug report is an artifact of the user's difficulty as he went through the extra motions to go back to the rainbow color map.

The overestimation of the efficacy of the rainbow color map might be a motivator to spend effort to go back to it, but it is unlikely to be a very strong motivator. In fact, other anecdotal incidents suggest that even users that are aware of the rainbow's flaws still have an affinity to use it. Consider the example visualization in Figure 3 created by one of my colleagues at Sandia National Laboratories. He and I have had many spirited conversations about color use and still he finds occasion to use the rainbow's colors. His argument is that this particular visualization is not used for scientific study but rather for communicating and engaging with non-experts, and these eye-grabbing colors are the best for this purpose.

Ultimately it is the eye-grabbing nature of the rainbow colors that keeps us coming back to them over and over again. Regardless of how we can interpret these colors, the bright and varying pure colors are certainly captivating (if not distracting), and it takes a good deal of time and expertise to beat these colors on aesthetics alone.

[1]http://www.paraview.org/Bug/view.php?id=7024

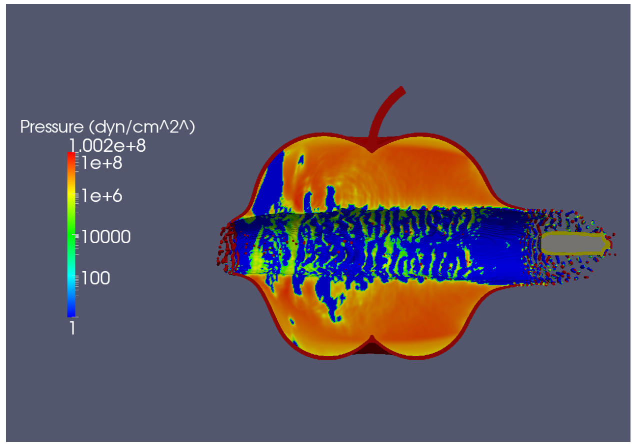

Figure 3. *Despite having years of experience and being well aware of the problems with rainbow color maps, the designer of this visualization chose these colors over others offered with better perceptual properties.*

Inertia

Finally, a major contributor to the persistence of the rainbow color map is that it has become entrenched in scientific visualization. Take for example VTK [22], the popular visualization library. Despite the fact that better color maps have been used with and contributed to VTK, the default coloring still reverts to the rainbow map. The problem is that although it is easy to add a new feature to VTK, changing a core feature like the default colors is difficult. VTK contains hundreds of regression tests that could break if the default colors are changed, and updating them can be very time consuming. Furthermore, such a change will necessarily break backwards compatibility and so has to be approved by the larger VTK community, another time consuming process. These software development minutiae create inertia in changing color map usage.

The same type of inertia forms within scientific communities using visualization tools. Often scientists compare current results with previous results. If previous scientific results are only available with the rainbow color map, the community is stuck with these color choices.

How We Can Promote Good Color Use

Providing and encouraging good color use in visualization is an ongoing battle. There are many activities the visualization research and development community can do, and likely all will be required to some degree.

Education

Much research has shown that subjects tend to overestimate the efficacy of the rainbow color map [4]. Thus, it is critical that visualization practitioners are educated on the appropriate use of color.

Fortunately, the visualization community has been vigilant in recent years to train its own practitioners in good color usage, and there is evidence that we have made progress. Borland and Taylor [5] report that 52% of papers in the 2005 IEEE Visualization conference proceedings displaying data with a pseudocoloring use a rainbow color map. In contrast in the 2014 IEEE Visualization conference proceedings, only 29% of papers (8 out of 28) featuring the pseudocoloring of a 3D scalar field use a rainbow color map, and only 16% of papers (10 out of 62) featuring the color representation of any sequential data use a rainbow color map. Progress is being made.

However, education has its limits. Although it might be feasible to educate visualization practitioners, it is unlikely we can reach every potential visualization user. Visualization is an integrated part of computational science, and many tools are designed with the understanding they will be used by the visualization layman. Furthermore, as we have seen previously it is often the case that even after having enough education to know the rainbow color map is bad, users still tend to apply it. Education is necessary, but not sufficient.

Admonishment

When education is not quite enough to prevent bad color usage, sometimes a little push is in order. When we see colors that we know are bad, we should take the time to attempt getting the creator to fix them. Sometimes this is simply informing the person. Sometimes it involves some admonishment.

Apart from face-to-face meetings with colleagues, there are many opportunities to redirect visualizations that go astray. One of the best such openings is the review of publications. One of the responsibilities of a peer reviewer is to ensure that the document, including its figures, conveys information effectively and honestly. As such, it is wholly appropriate to reprimand graphical displays that use perceptually misleading colors, and as a barrier to publication, you as a reviewer can add some motivation.

In addition to scrutinizing individual visualization examples, it is good to examine the software tools used to generate visualizations. Does the software attempt to produce good colors, or does it simply generate rainbow colors by default, thereby further encouraging its use? If the latter, then users and observers should point out this fault. One such mechanism is to raise bug or feature requests with the development team. Another mechanism is to point out the deficiency in a public setting the developers may participate in. After all, it was the public admonishment in the publication by Borland and Taylor [5] more than anything that motivated the ParaView development team to remove the rainbow color map as the default.

And on that note, **VisIt** [8], **EnSight** [9], **VTK** [22], **VAPOR** [15], and **MayaVi** [16], consider yourselves admonished for continuing to use the rainbow color map as the default.

Simplification

As stated previously, one of the main factors making the rainbow colors a primary choice in visualization is the sheer simplicity of creating them. However, the simpler it is to create good colors, the more likely a user is going to do so. Ideally, we make good coloring become easier to apply than poor coloring.

The first step is for the experts designing color for visualization to make the colors they create accessible. Creating examples and publishing literature is important, but to make a real impact on the field of visualization, the coloring must be delivered in a format that can be used by non-experts.

An excellent example of making color research accessible is the work by Brewer, et al. [7]. This work has produced a plethora of color schemes that can be used for mapping values to colors. But the real value for visualization practitioners is that Brewer provides a web page [1] shown in Figure 4 with a simple and intuitive interface for choosing a collection of colors. Once selected, the site provides color values that are easily imported into any program, API, or web interface. The work of creating and maintaining this site is well above and beyond any of the publications, but the effect this web site has on the field of visualization is well beyond anything the publications can offer. This repository of colors is so effective that although the design was originally targeted specifically for cartography, the Color Brewer website is a standard reference for all types of visualization.

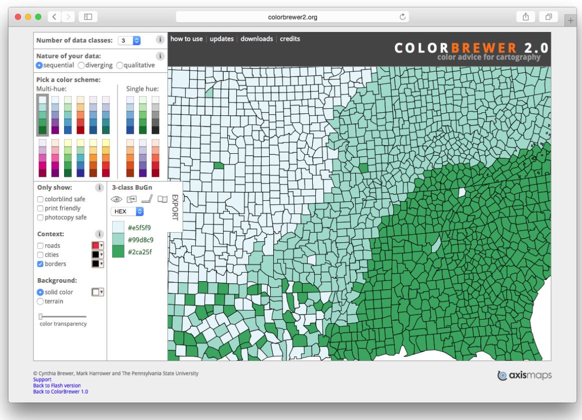

Figure 4. The ColorBrewer web site.

We cannot all be as cool as Color Brewer, but even simple ready-to-play resources can make a difference. Consider my earlier paper on diverging color maps [13]. When I published this paper I posted a simple companion web site providing example color values that are easily imported in software as well as the software and tools used in constructing the paper [2]. I continue to maintain the site and have additionally added contributions from several readers. Although this page requires some effort to build and maintain, it has immeasurably helped the research take hold in practice, which both helps the visualization community as a whole and helps increase visibility of my personal visualization work.

Ultimately, almost all visualizations are made through some form of software tools, and we can make the biggest positive impact by designing these tools to make good use of color. Tools like ParaView [3] and MATLAB have recently changed their default coloring to move away from rainbow colors. These changes engender improvements in a large quantity of visualizations produced.

Time

Because tools and communities have inertia in how they create visualizations, we cannot expect color usage to change immediately. It takes patience, but with time and pressure we can make a difference. Progress is being made, and visualization experts must continue their education, admonishment, and simplification.

Simple Practical Advice

The remainder of this paper is spent administering simple, practical advice for novice users. Engineering good color maps is complex as it requires considerable knowledge of color theory and perception, involves the resolution of conflicting goals, and is aided by some artistic ability. Choosing good colors is time consuming even for experts.

This advice attempts to circumvent all these complexities by instead leveraging ready-made colors. As there is no perfect color map for all instances we consider multiple options, but attempt to keep the number of options small for simplicity.

Color Brewer

The first resource for any color choice should be the aforementioned excellent Color Brewer web page [1] shown in Figure 4. The collection of color maps provided is very well organized, making it easy to find a set of colors appropriate for whatever visualization it is applied to. The extensive collection of color maps makes it likely that one is available for whatever visualization needs you have. And since each set of colors is designed by experts, they are efficacious as well as attractive.

3D Visualization

A common operation in scientific visualization is the application of pseudocoloring on a 3D surface or volume. When using pseudocoloring in a 3D environment, extra care should be taken in the color choices as the shading of colors provides important spatial cues. The simultaneous coloring and shading should not interfere with each other.

Some practitioners propose using isoluminant color maps so that the psuedocoloring does not alter the shading in any way. However, isoluminant colors tend to make poor linear maps because of their low contrast [14, 18, 25]. Also the visual system is quite adept at distinguishing shading caused by textures and shading caused by lighting conditions as demonstrated in Figure 5.

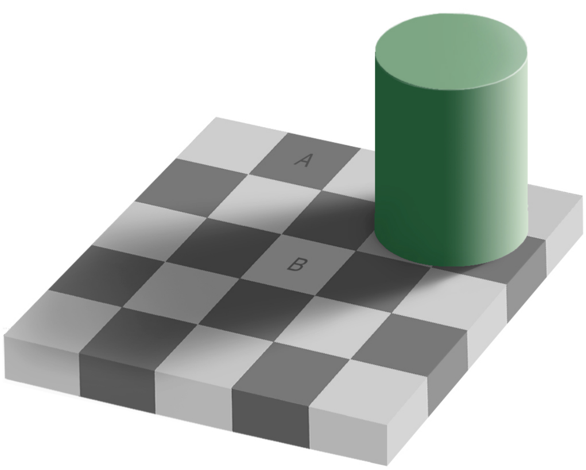

Figure 5. *The checker shadow illusion first published by Edward H. Adelson. Although the squares marked A and B measure the exact same brightness, the visual system has no trouble distinguishing the shading of the checkerboard texture on the floor from the shadow cast by the cylinder.*

That said, 3D shading requires surfaces that reflect a good amount of light to be effective. Dark surfaces appear amorphous, so a color map used in 3D should retain some brightness throughout. Color maps that drop to black are unusable on 3D shapes.

Common advice for mapping a scalar field with no special middle value is to use a color map with monotonically increasing brightness because brightness is a good indication of order and provides high contrast. The maximum range of brightness that can be achieved is to go from black to white (with any number of hues used in between). However, because the darkest colors interfere with 3D shading, the darker colors cannot be used, reducing the perceptual resolution of the color map.

An alternative to a map with monotonic brightness is a diverging color map, which is a double-ended map containing colors with different hues at each end and meeting with a bright neutral color in the middle. Diverging color maps are traditionally designed for displaying scalars that have a value of special significance in the middle (such as sea level for elevation or the freezing point for temperature).

Early advice condemned the use of diverging color maps for showing a uniform range of values because they do not have monotonic brightness. However, recent works suggest using diverging color maps with hues having low/high cues, such as cool and warm colors, that naturally convey the relative values [13]. Recent perceptual studies have shown the diverging color maps to be efficacious [4, 21].

The Color Brewer web site [1] contains several diverging color maps that can work well in 3D. There are also resources for diverging color maps that smoothly interpolate throughout [2] as shown in Figure 6.

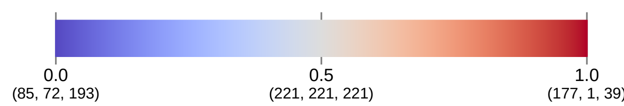

Figure 6. *Smooth diverging color maps for scientific visualization. Colors are interpolated in the polar Msh color space [13].*

Flat Scalar Fields

Some use cases of scientific visualization involve the pseudocoloring of fields on flat 2D surfaces rather than in 3D. Common use cases include geographic fields, dense arrays or matrices, and two dimensional functions. In this case, the pseudocoloring can be applied to the flat surface of the image, a technique sometimes known as a heat map. Because the 2D surface requires no shading, it does not have the color limitations of 3D data.

Thus, to maximize the perceptual resolution of a flat field, it is sometimes desirable to use a color map that goes from no brightness (black) to maximum brightness (white). Although a gray scale ramp satisfies this criteria, simultaneous contrast tends to make it perceptually inaccurate [25]. Inserting changing hues helps remove the problems with simultaneous contrast.

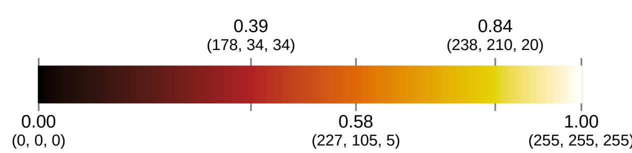

Figure 7. *The black body radiation color map. The provided colors can be interpolated in the CIELAB color space.*

One effective color map going from black to white is known as black body radiation, shown in Figure 7. This color map is inspired by the color of light emitted by a body held at different temperatures. The changes in red, orange, and yellow hues help distinguish colors with different backgrounds and the progression of hues helps reinforce the interpretation of the colors.

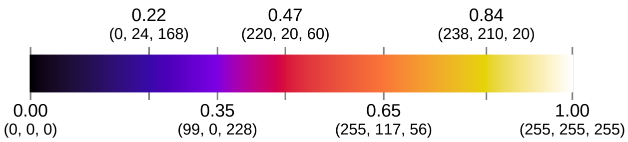

Figure 8. *A perceptually linear map incorporating blue and purple hues in addition to the reds and yellows of the black body radiation. The provided colors can be interpolated in the CIELAB color space.*

Although perceptually effective, many observers find it more appealing to have more hues. The color map shown in Figure 8, which borrows from the gnuplot default color palette, mixes blue and purple hues to make more appealing colors.

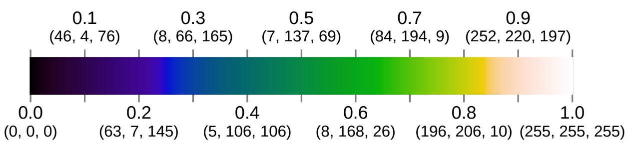

Figure 9. *The Kindlmann color map [11]. The provided colors interpolate reasonably well in the CIELAB color space.*

Alternatively, we could use a maximum change of hue between black and white to even better distinguish colors in the map. One way to do this is to spin the hue much like you would for a rainbow color map, but adjust the brightness of each color to match its placement in the color map as shown in Figure 9. This color map is often referred to as the Kindlmann color map as it was first proposed in a paper by Kindlmann, et al. [11]. Similar variations such as the cubehelix color map [10] also exist. This type of color map performs well, especially in comparison to the rainbow color map.

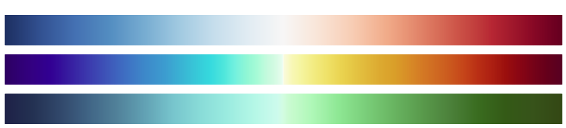

Figure 10. *A collection of diverging color maps designed by Francesca Samsel [21]. The darker colors at the endpoints give a larger perceptual range for 2D visualization, but the dark colors will not work well with shading on 3D surfaces.*

Another approach is to use a diverging color map similar to what is discussed for 3D visualization. However, for 2D flat fields it is possible to extend the range of colors to be darker. Figure 10 demonstrates some color maps that can be used in 2D visualizations that have a larger range than their 3D counterparts.

Conclusion

Despite its many flaws, the use of rainbow colors still persists in many scientific visualizations. However, there are simple things we can do to discourage its use. We must continue to educate users on basic color usage and admonish those that misuse color.

Most importantly, we should make using good color maps as easy as possible. This paper provides some simple advice with clear examples for color maps that generally work well within scientific visualization. To supplement this information and make these color maps even more accessible, I have established the following web site to post these color maps in a format that is easily added to visualizations.

`http://kennethmoreland.com/color-advice`

As we make visualization tools and libraries, we should use reference material like this to provide good default colors to end users.

Acknowledgments

Thanks to Francesca Samsel and Terry Turton for providing their color map designs. Thanks to Jason Wilke for providing simulation data. Also thanks to W. Alan Scott for his visualization examples (even though I promised not to name him).

This material is based in part upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, under Award Numbers 14-017566 and 12-015215. This work was also funded by the U.S. Department of Energy, National Nuclear Security Administration, Advanced Simulation and Computing Program. Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

References

[1] `http://colorbrewer2.org`.

[2] `http://www.kennethmoreland.com/color-maps/`.

[3] James Ahrens, Berk Geveci, and Charles Law. Paraview: An end-user tool for large data visualization. In *Visualization Handbook*. Elesvier, 2005. ISBN 978-0123875822.

[4] Michelle A. Borkin, Krzysztof Z. Gajos, Amanda Peters, Dimitrios Mitsouras, Simone Melchionna, Frank J. Rybicki, Charles L. Feldman, and Hanspeter Pfister. Evaluation of artery visualizations for heart disease diagnosis. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2479–2488, December 2011. DOI 10.1109/TVCG.2011.192.

[5] David Borland and Russell M. Taylor II. Rainbow color map (still) considered harmful. *IEEE Computer Graphics and Applications*, 27(2):14–17, March/April 2007. DOI 10.1109/MCG.2007.323435.

[6] Cynthia A. Brewer. *Designing better MAPS: A Guide for GIS Users*. ESRI Press, 2005. ISBN 1-58948-089-9.

[7] Cynthia A. Brewer, Geoffrey W. Hatchard, and Mark A. Harrower. Colorbrewer in print: A catalog of color schemes for maps. *Cartography and Geographic Information Science*, 30(1):5–32, 2003. DOI 10.1559/152304003100010929.

[8] Hank Childs, Eric Brugger, Brad Whitlock, Jeremy Meredith, Sean Ahern, David Pugmire, Kathleen Biagas, Mark Miller, Cyrus Harrison, Gunther H. Weber, Hari Krishnan, Thomas Fogal, Allen Sanderson, Christoph Garth, E. Wes Bethel, David Camp, Oliver Rübel, Marc Durant, Jean M. Favre, and Paul Navrátil. VisIt: An end-user tool for visualizing and analyzing very large data. In *High Performance Visualization: Enabling Extreme-Scale Scientific Insight*, pages 357–372. October 2012.

[9] Randall Frank and Michael F. Krogh. The EnSight visualization application. In *High Performance Visualization: Enabling Extreme-*

Scale Scientific Insight, pages 429–442. CRC Press/Francis–Taylor Group, 2013.

[10] D. A. Green. A colour scheme for the display of astronomical intensity images. *Bulletin of the Astronomical Society of India*, 39:289–295, 2011. arXiv:1108.5083.

[11] Gordon Kindlmann, Erik Reinhard, and Sarah Creem. Face-based luminance matching for perceptual colormap generation. In *Proceedings of IEEE Visualization*, pages 299–306, October 2002. DOI 10.1109/VISUAL.2002.1183788.

[12] A. Light and P.J. Bartlein. The end of the rainbow? Color schemes for improved data graphics. *EOS, Transactions, American Geophysical Union*, 85(40):385, 391, October 2004.

[13] Kenneth Moreland. Diverging color maps for scientific visualization. In *Advances in Visual Computing (Proceedings of the 5th International Symposium on Visual Computing)*, volume 5876, pages 92–103, December 2009. DOI 10.1007/978-3-642-10520-3_9.

[14] Kathy T. Mullen. The contrast sensitivity of human colour vision to red–green and blue–yellow chromatic gratings. *The Journal of Physiology*, 359:381–400, February 1985.

[15] Alan Norton and John Clyne. The VAPOR visualization application. In *High Performance Visualization: Enabling Extreme-Scale Scientific Insight*, pages 415–427. CRC Press/Francis–Taylor Group, 2013.

[16] Prabhu Ramachandran and Ga el Varoquaux. Mayavi: 3D visualization of scientific data. *Computing in Science & Engineering*, 13(2):40–51, March/April 2011. DOI 10.1109/MCSE.2011.35.

[17] Penny Rheingans. Task-based color scale design. In *Proceedings of Applied Image and Pattern Recognition '99*, pages 35–43, October 1999. DOI 10.1117/12.384882.

[18] Bernice E. Rogowitz and Alan D. Kalvin. The "which blair project": A quick visual method for evaluating perceptual color maps. In *Proceedings of the Conference on Visualization '01*, pages 183–190, October 2001.

[19] Bernice E. Rogowitz and Lloyd A. Treinish. Data visualization: the end of the rainbow. *IEEE Spectrum*, 35(12):52–59, December 1998. DOI 10.1109/6.736450.

[20] Bernice E. Rogowitz, Lloyd A. Treinish, and Steve Bryson. How *not* to lie with visualization. *Computers in Physics*, 10(3):268–273, May/June 1996. DOI 10.1063/1.4822401.

[21] Francesca Samsel, Mark Petersen, Terece Geld, Greg Abram, Joanne Wendelberger, and James Ahrens. Colormaps that improve perception of high-resolution ocean data. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '15)*, pages 703–710, 2015. DOI 10.1145/2702613.2702975.

[22] Will Schroeder, Ken Martin, and Bill Lorensen. *The Visualization Toolkit: An Object Oriented Approach to 3D Graphics*. Kitware Inc., fourth edition, 2004. ISBN 1-930934-19-X.

[23] Samuel Silva, Beatriz Sousa Santosa, and Joaquim Madeira. Using color in visualization: A survey. *Computers & Graphics*, 35(2):320–333, April 2011. DOI 10.1016/j.cag.2010.11.015.

[24] Colin Ware. Color sequences for univariate maps: Theory, experiments, and principles. *IEEE Computer Graphics and Applications*, 8(5):41–49, September/October 1988. DOI 10.1109/38.7760.

[25] Colin Ware. *Information Visualization: Perception for Design*. Morgan Kaufmann, 2nd edition, 2004. ISBN 1-55860-819-2.

Author Biography

Dr. Kenneth Moreland is a principal member of technical staff at Sandia National Laboratories. He received the BS degrees in computer science and electrical engineering from New Mexico Tech in 1997. He received MS and Ph.D. degrees in computer science from the University of New Mexico in 2000, and 2004, respectively. Dr. Moreland specializes in large-scale visualization and plays an active role in several HPC products including ParaView, VTK, IceT, Dax, and VTK-m.