Homework 6

Milena Belianovich
Tuesday, April 30

1 Part 1: Uncertainty Visualization of Isocontours

1. Uncertainty in data (adding and removing noise) with step-by-step visualization. First, let's generate some noisy data and visualize the uncertainty. Create a uniform array from 0 to 1 in x with 100 points. Now, let $y = \sin(10\pi x) + \sin(20\pi x)$. Add Gaussian noise with mean of 0 and standard deviation of $\frac{abs(y)}{2}$ to each y-value. Do this for 100 instances of $y(x)$. You should get a 100×100 array plotted individually as lines.

   Now, plot the mean and one standard deviation over the set of noisy data. Include an image of the plots in your report. Describe how your plot represents the uncertainty in the data based on the form of our Gaussian noise.

   Solution.

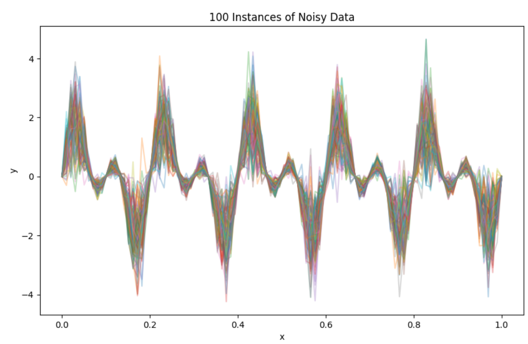


Figure 1: Part 1: 100 Instances of Noisy Data

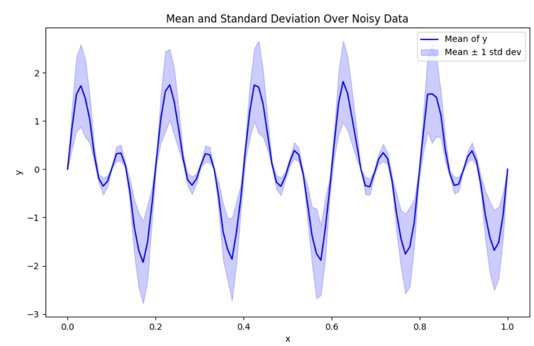


Figure 2: Part 1: Mean and Standard Deviation Over Noisy Data

   100 Instances of Noisy Data (Figure 1): Each line represents one instance of noisy data, where the noise was added to the sinusoidal function based on Gaussian distribution with a mean of 0 and a standard deviation of $\frac{\text{abs}(y)}{2}$. The wide variety of lines, shown in different colors, reflects the varying noise levels dependent on the value of y at each point.

   Mean and Standard Deviation Over Noisy Data (Figure 2): This plot shows the mean of the noisy data calculated across the 100 instances, plotted as a blue line. The shaded area around the mean represents one standard deviation above and below the mean, giving an indication of the spread of the data. The width of this shaded area varies across different x values, representing the uncertainty in y values due to the magnitude-dependent noise. This area is broader where the absolute value of y (and hence the standard deviation) is larger, indicating greater uncertainty.

   The plots and data generation highlight how the form of Gaussian noise (dependent on $\frac{\text{abs}(y)}{2}$) results in uncertainty that is not uniform across the range of x. The uncertainty is higher where the function y has higher absolute values, reflecting the proportional noise scale.

code: https://github.com/milenabel/CS6635-VisforScD

- Extra Credit: Use Scipy or another package

  First, use a Fast Fourier transform (fft) to identify the frequencies. Then apply an appropriate bandpass Butterworth filter to remove signals other than $\sin(10\pi x)$.

  Hint: use scipy.ftt and scipy.signal.butter. Use the bandpass range $[4-6]$ and set the order of the filter to 2. Include an image of the plots in your report. Describe what you learned from removing the noise. Why is the de-noised data different at lower x values?

  Solution.

  Insights:

  The filtering successfully isolated the $\sin(10\pi x)$ component, evident from the "Comparison of Exact Data and Filtered Signal" plot (Figure 4). The filtered signal closely matches the exact $\sin(10\pi x)$ function, demonstrating the effectiveness of the Butterworth filter in extracting relevant frequency components while suppressing others. By observing the mean and standard deviation in the "Mean and Standard Deviation Over Noisy Data" plot (Figure 2), it's clear how much variability the noise introduced at each point across the domain. This spread indicates the level of uncertainty in the measurements or simulations from which the data might have been derived. The "Frequency Spectrum of the Mean Data" plot (Figure 3) shows the dominant frequencies in the data, with significant peaks at frequencies corresponding to the $\sin(10\pi x)$ and $\sin(20\pi x)$ components. This visual representation of the frequency domain is crucial for understanding which components contribute to the signal and guiding the filter design. The "100 Instances of Noisy Data" plot (Figure 1) illustrates the variation between different instances of the noisy data. It highlights how each instance deviates from the mean and provides a visual understanding of the noise level and its effect on the overall data structure.

  Observing the filtered signal at lower x values shows some discrepancies from the exact function. This could be due to several factors inherent in digital signal processing:

  - Filter Initialization: Digital filters require a "warm-up" phase where their internal states adjust to the input data. At the start of the signal, these states are not fully adjusted, which can lead to initial inaccuracies known as transient effects.
  - Edge Effects: Since the filtered signal relies on a window of data points to determine each point in the output, the beginning of the signal, where fewer previous data points are available, might show edge effects. These effects can manifest as distortions that deviate from the expected output.
  - Phase Shifts and Group Delay: The Butterworth filter, being a non-linear phase filter, introduces phase shifts and group delay, especially near the cutoff frequencies. These shifts can affect how the signal aligns temporally with the original data, potentially causing the start of the filtered data to lag behind or lead the actual data.

2. Generate a spaghetti plot (uncertainty visualization technique) of isocontours for flow simulation data in ParaView:

   (a) Download the flow simulations flow1.vtk, flow2.vtk, and meanFlow.vtk located in data/ flowData directory. The flow1.vtk and flow2.vtk represent the same flow captured with two different fluid viscosity values. The meanFlow.vtk represents the mean of the two flow simulations (datasets courtesy of the Gerris Flow Solver project.

   (b) Load the meanFlow.vtk, flow1.vtk, and flow2.vtk, and visualize their isocontours with different colors for the isovalue = 40 to generate a visualization (please include the image in your report).

   Solution. After this part, we get the plot presented in Figure 5.

code: https://github.com/milenabel/CS6635-VisforScD

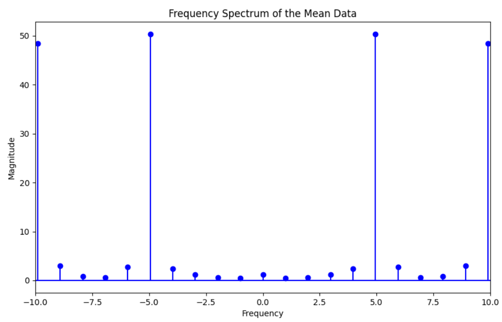


Figure 3: Part 1: Frequency Spectrum of the Mean Data

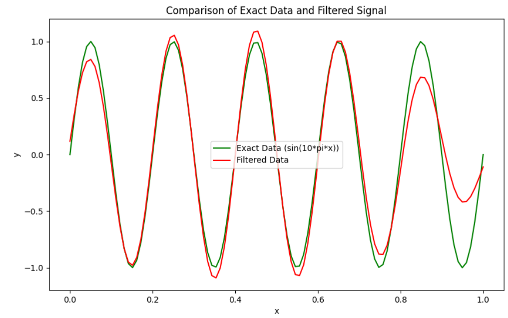


Figure 4: Part 1: Comparison of Exact Data and Filtered Signal

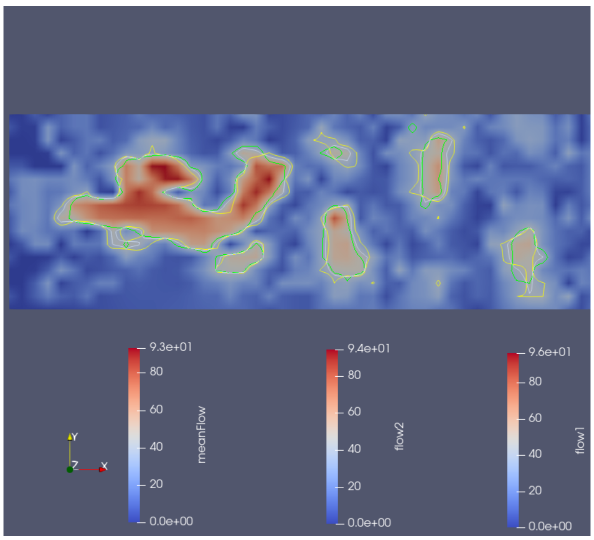


Figure 5: Part 1: Spaghetti Plot

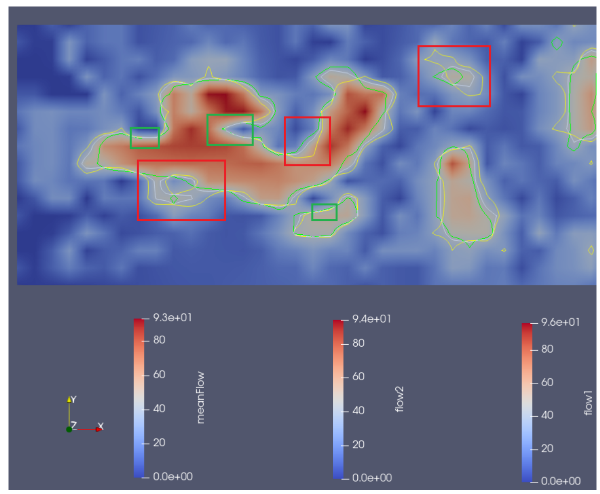


Figure 6: Part 1: Image with Boxes Showing Regions of High(Red) and Low(Green) Variability

(c) Do all isocontours coincide? Describe why they do or do not coincide. Overlay the image from part (b) with three or four boxes showing regions of high/low positional variability in your spaghetti visualization of isocontours. Please include the image in your report.

Solution. After this part, we get the plot presented in Figure 6.

The contours from the three datasets appear to coincide in some regions but significantly diverge in others. This is evident where the contours overlap perfectly and where they are separate or only partially overlap.

Variability in Flow: The differences in alignment between the contours from flow1 and flow2 as compared to meanFlow are indicative of the variability in the flow due to the different fluid viscosity settings used in the simulations. Higher viscosity in one of the simulations could lead to slower movement, resulting in different flow patterns as captured by the isocontours (Figure 5).

code: https://github.com/milenabel/CS6635-VisforScD

3. In your own words, briefly describe contour boxplots, and describe the conceptual similarities and differences between contour box plots and 1D boxplots, e.g., the ones generated in question 1. Refer to the Contour Box plots paper: Whitaker, Ross T.; Mash Mirzargar; Robert M. Kirby (2013). "Contour Box plots: A Method for Characterizing Uncertainty in Feature Sets from Simulation Ensembles". IEEE Transactions on Visualization and Computer Graphics.

   When summarizing, please do it in your own words. Do not write the same thing as in the source material but with only slight changes in phrasing. Do not simply rewrite the same sentences as in the source material. If you are paraphrasing, it cannot be too similar to the source material. You should be able to write your answer without simultaneously looking at the original text.

   Solution.

   Contour boxplots are an advanced statistical tool designed for visualizing and understanding the variability and uncertainty in ensembles of contours, typically derived from simulation data. These plots are particularly useful in fields like meteorology, fluid dynamics, and other areas. Contour boxplots extend the concept of traditional boxplots, which are familiar in statistical analysis for summarizing distributions through medians, quartiles, and outliers.

   In traditional 1D boxplots, the data is represented along a single dimension, showing the distribution's central tendency and spread through a box and whiskers format. This method is quite effective for scalar data but doesn't translate directly to functional or spatial data, where features like shapes or contours are important.

   Contour boxplots, however, offer a way to analyze and visualize such complex data. They use a concept known as "data depth" to organize and represent data. Data depth helps in determining how central or representative a particular contour is within the ensemble, effectively sorting contours from the most representative to the outliers. This method results in a visualization that shows central (median) contours, variability around these contours, and potential outliers.

   Conceptual Similarities:

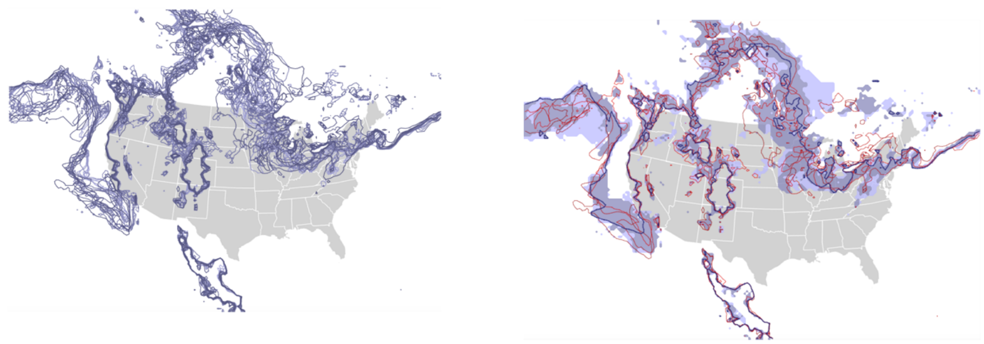
   - Both types of plots aim to provide a visual summary of the data's central tendency—median for boxplots and central contour for contour boxplots.
   - Both methods display the spread of the data, with traditional boxplots showing interquartile ranges and whiskers for potential outliers, while contour boxplots show bands of variability around the median contour.
   - Both plots are used to identify outliers, helping in understanding extremes in the data set.

   Conceptual Differences:

   - Traditional boxplots are limited to scalar data and primarily one-dimensional, while contour boxplots handle multi-dimensional data, particularly useful for spatial data sets like those generated in simulations.
   - Contour boxplots are specifically designed for contour data or isocontours, which are not straightforwardly represented in scalar terms. They consider the geometric properties of data.
   - While traditional boxplots provide a straightforward interpretation of data distribution along a numerical scale, contour boxplots require a more nuanced understanding since they deal with shapes and spatial relationships.

   In conclusion, contour boxplots are a powerful extension of the traditional boxplot concept, adapted to meet the challenges of visualizing uncertainty and variability in complex simulation data.

4. The images in figure below depict the spaghetti and contour box plots for an uncertain temperature field. In your own words, discuss the advantages/disadvantages of contour box plots over spaghetti plots for isocontour uncertainty visualization.



(a)Spaghetti plots (b) Contour box plots

Fig: Uncertainty visualization of isocontours

Median Isocontour Outlier Isocontour

Solution.

The images clearly show the differences between spaghetti plots and contour box plots for visualizing uncertainty in isocontours. Both types of plots aim to show where certain values are predicted to be based on an ensemble of simulations, but they convey this information in distinct ways that have various advantages and disadvantages.

Spaghetti Plots

Advantages:

- Each line in a spaghetti plot represents an individual simulation's result, preserving the specific outcome of each simulation.
- They are straightforward and easy to understand, depicting variability through the dispersion of lines.

Disadvantages:

- As the number of simulations increases, spaghetti plots can become very cluttered, making it difficult to discern patterns or central tendencies.
- Spaghetti plots can sometimes give undue emphasis to outlier simulations because they are as visually prominent as more central outcomes.
- These plots show variability but do not provide a clear statistical summary or identify measures like the median or quantiles without additional processing.

Contour Box Plots

Advantages:

- Contour box plots provide a clear visualization of the median or most representative isocontour, quantiles, and potential outliers, offering a compact statistical summary.

code: https://github.com/milenabel/CS6635-VisforScD

- By focusing on statistical boundaries rather than individual outcomes, contour box plots can simplify the visualization, making it easier to interpret in the presence of many simulations.
- These plots are designed to highlight outliers explicitly, which are crucial for risk assessment and decision-making.

Disadvantages:

- While reducing visual complexity, contour box plots might obscure specific details of individual simulations, which could be important in certain analytical contexts.
- Understanding contour box plots and their statistical implications can be more complex and may require a higher level of statistical literacy.

Conclusion.

Contour box plots offer a more structured and statistically informative way to visualize uncertainty in isocontour data compared to spaghetti plots. They provide a clearer picture of central tendencies and variability, which is particularly valuable in making decisions based on the data. However, for a detailed examination of each possible scenario, spaghetti plots might still be preferred due to their detailed preservation of individual simulation outcomes. In practice, the choice between these visualization techniques should consider the specific needs for analysis, the audience's statistical understanding, and the complexity of the data.

2 Part 2: Uncertainty Visualization of a Vector Field

Download the uncertain wind field data wind1.vtk and wind2.vtk located in data/windData. The mean vector field is represented by meanWind.vtk.

1. Visualize each vector field using Arrow Glyphs with different colors using the Solid Colors option to generate three images and put them in your report. Now overlay these three images with a color mapped surface to visualize the uncertainty in vector directions as depicted below. Put the resulting image/images in your report.

   Solution. The result of this step can be seen in Figure 7.

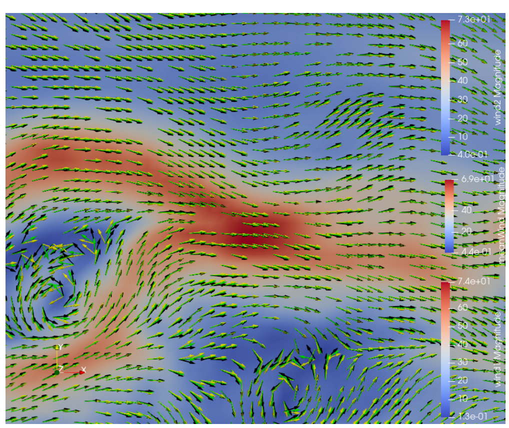


Figure 7: Part 2: Glyph Images with Color Mapped Surface

code: https://github.com/milenabel/CS6635-VisforScD

2. Now change the color map of the surface to visualize the angle between wind1 and wind2 vector fields. You will need the Append Attributes filter to combine the fields and the Calculator filter to compute the angle. Note that the Calculator's "mag" function computes the vector's length, which is commonly called the vector norm, and the "norm" function normalizes the vector. Then, enable log scale for color maps and use the color map's control points(located below the color map's opacity line editor) to enhance contrast. Then, make the wind1 and wind2 glyphs smaller than the meanWind to make it easier to discern. Show an image of the whole field and one of a turbulent region and put them in your report.

   Solution. The results of this step can be seen in Figures 8 and 9.

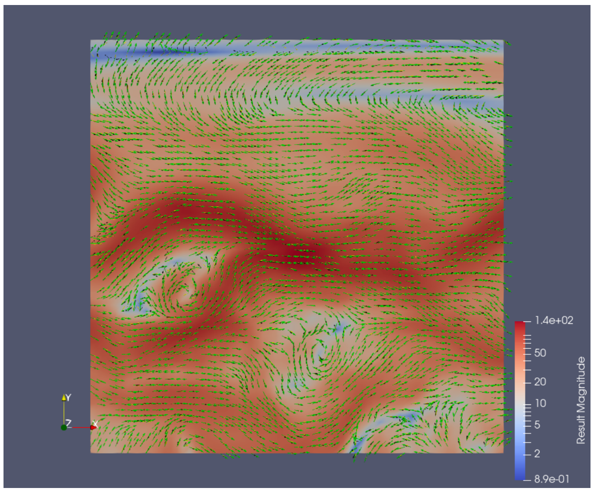


Figure 8: Part 2: Image of a Whole Field

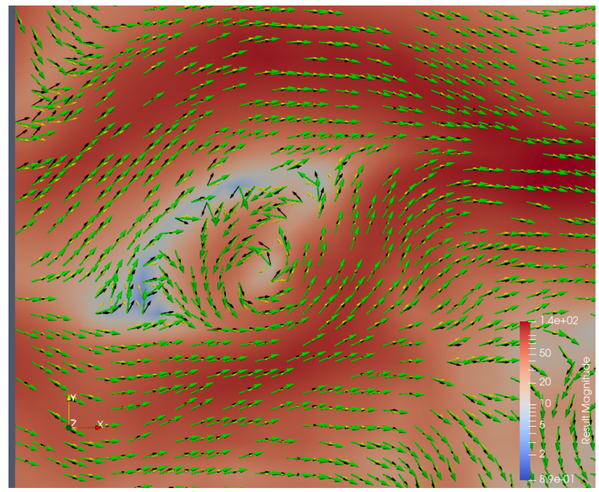


Figure 9: Part 2: Turbulent Region of the Whole Field

3 Part 3: Reading questions

These questions pertain to Chapter 39 of The Visualization Handbook: Extending Visualization to Perceptualization.

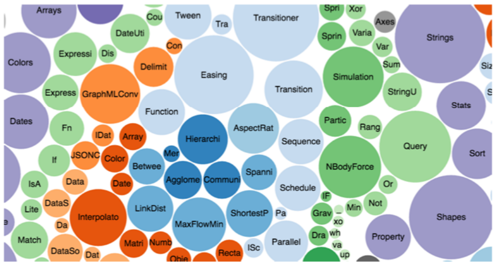
1. What is preattentive processing, and why is important? Please answer in your own words.

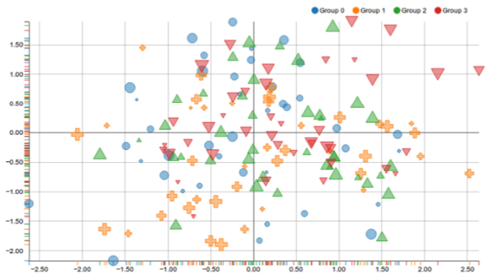
   Solution.

   Preattentive processing refers to the brain's ability to quickly assimilate and process visual information from the environment without conscious effort. This occurs almost instantaneously, allowing us to recognize certain visual elements that stand out due to their distinct properties. Preattentive processing is essential because it enables rapid decision-making and understanding by helping us focus on the most important or anomalous parts of a visual scene. This process enhances efficiency in tasks that require quick data interpretation, such as scanning dashboards for anomalies or navigating complex environments.

code: https://github.com/milenabel/CS6635-VisforScD

2. What types of preattentively processed features are used in the following two visualizations? Please answer separately each image. Try to identify as many relevant types/classes as possible.





Solution.

(a) The bubble chart visualization uses various preattentive features:

- Color: Different colors help in distinguishing between various categories or groups without needing to process each element individually.
- Size: Varying sizes of the bubbles immediately draw attention to larger (or smaller) values, indicating magnitude or importance at a glance.
- Spatial Positioning: The arrangement allows the eye to compare data across the horizontal (X-axis) and vertical (Y-axis) planes efficiently.

(b) Preattentive features in the scatter plot include:

- Shape: Different shapes (circles, triangles, squares) are used to represent different groups or categories within the data, allowing for quick differentiation.
- Color: Like in the bubble chart, color here also differentiates between groups, aiding in quick categorization and comparison across the visualization.
- Size: Varying sizes of the data shapes, just as for bubble chart, draw attention to different sizes of values, indicating magnitude at a glance.
- Position along common scales: The placement of marks along the X and Y axes allows for immediate assessment of value relationships and distributions, harnessing our ability to quickly assess spatial differences.

Conclusion

These visualizations effectively use preattentive processing to enhance data comprehension. The bubble chart, with its use of size and color, leverages human visual capabilities to quickly convey information about data magnitude and category. The scatter plot uses shape, size and color to categorize data while enabling fast comparison of positions and distributions, crucial for identifying trends or anomalies quickly. These tools show how understanding and applying preattentive features can make complex data more accessible and understandable.

3. We have created many visualizations in previous assignments. Select one visualization that you have created from a previous assignment that may not be perceptually significant. State what you can do to improve the perceptualization and try to re-visualize it in ParaView. (Images of your previous visualization and the new design should be in your report.)

Solution.

From the various assignments we have worked on, the visualizations that stand out as possibly lacking perceptual significance might be the problem from assignment 2, where we worked on ”Visualization

code: https://github.com/milenabel/CS6635-VisforScD

of 2D images" using a single threshold to capture specific features like the Grand Canyon's riverbed, could benefit from enhanced perceptualization techniques. The final result in this task might be more perceptually significant if it incorporated better depth perception and more detailed differentiation of features.

To improve the perceptualization of this 2D scalar field:

- Incorporating Shading: Introduce artificial shading to enhance depth perception. This could make elevation differences more apparent and give a 3D-like effect to the 2D image.
- Layered Visualization: Overlay transparent layers of data where higher thresholds can be rendered in semi-transparent colors to provide a sense of depth and complexity.

Now, we try to re-visualize this in ParaView using these suggestions:

First, load the original data file in ParaView. Then, enhance depth perception and overlay contours with annotations to guide viewer attention.

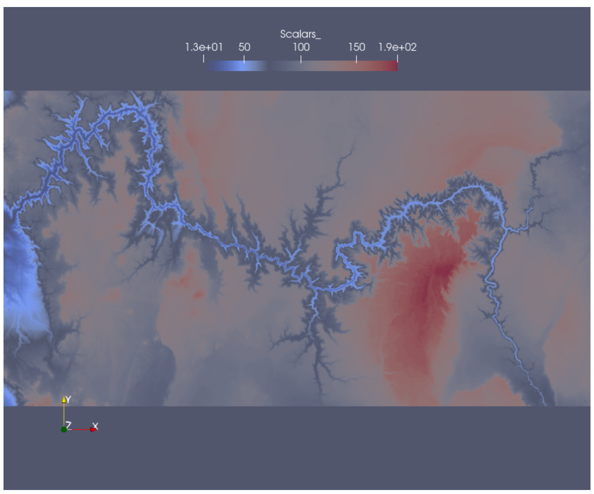


Figure 10: Part 3: Riverbed Visualization with Opacity Changes

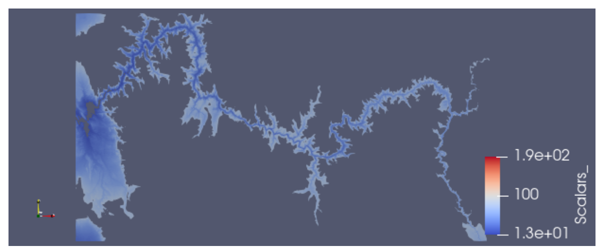


Figure 11: Part 3: Previous Riverbed Visualization

In Figure 10, you can see the result of using a transfer function for color and opacity which enhances the features of the riverbed more and highlights the desired data.

In comparison, Figure 11 shows the previous result of isolating the riverbed using a threshold, which does enhance the riverbed but also has rough outlines and looks not very appealing.

In Figure 12, we are adding the isocontour lines that enhance the features of the terrain even more.

In Figure 13, we can see the isolated isocontour lines. Another added change for this visualization was the change in ambiance from 0 to 0.7, which visibly enhanced the riverbed pronunciation.

Lastly, we add isocontour values as seen in Figure 14, but enhance the color of the text and make the font bold in order to visualize the values more coherently. For that reason we also zoomed in onto the image much more compared to the previous visualization as seen in Figure 15.

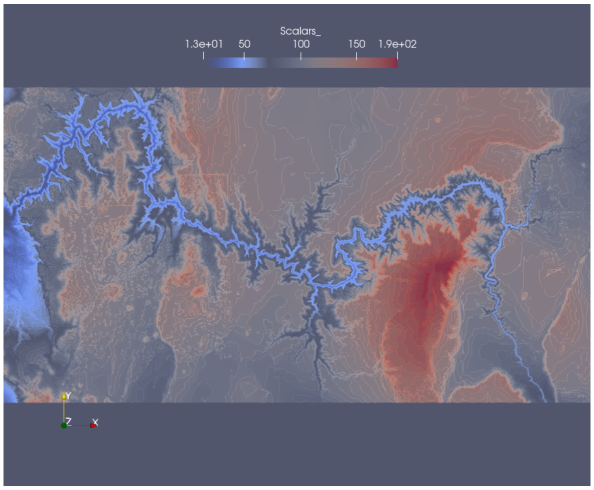


Figure 12: Part 3: Riverbed Visualization with Isocontours

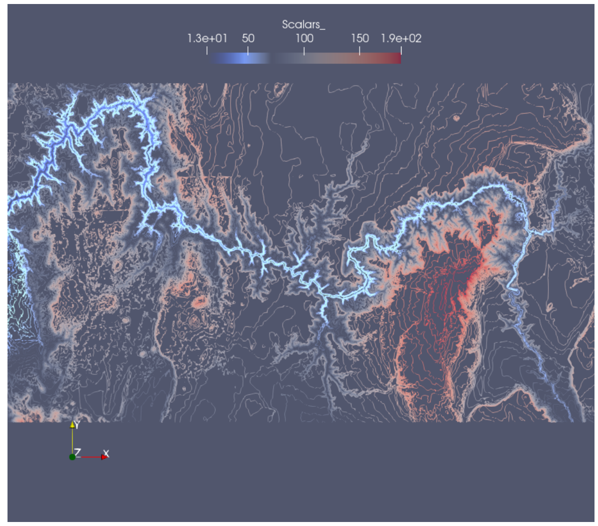


Figure 13: Part 3: Riverbed Visualization with Only Isocountours

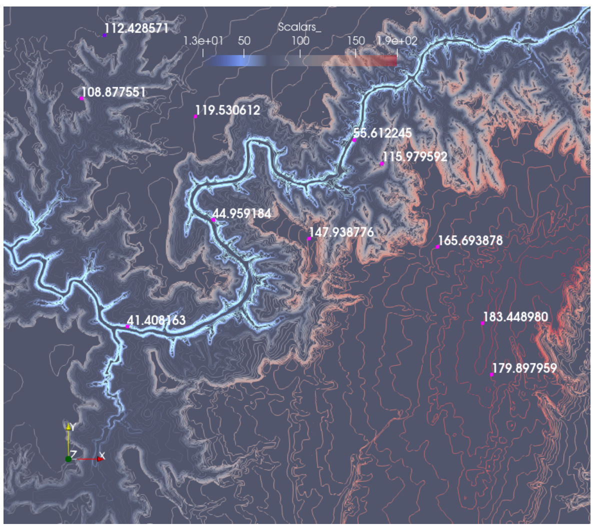


Figure 14: Part 3: Riverbed Visualization with Isocontour Values

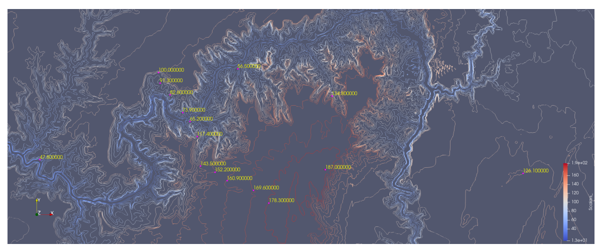


Figure 15: Part 3: Previous Riverbed Visualization with Isocontour Values

4 Part 4: Color Maps

- Color maps are very important in conveying information in data visualization. Load the data under the part4/ directory in ParaView, visualize it using different colormaps provided in Paraview: "Cool to Warm" and "Rainbow Desaturated".

  Solution. The results of this step can be seen in Figures 16 and 17.

- Now make a custom colormap using the website https://sciviscolor.org/colormovesapp/. To use the tool, you will need to create a reference image of the dataset which you drag and drop into the ColorMoves window. To create the reference image, you will need to follow this guide: https://sciviscolor.org/colormoves/flo files/instructions/

  Solution. The slice of a dataset can be seen in Figure 18.

code: https://github.com/milenabel/CS6635-VisforScD

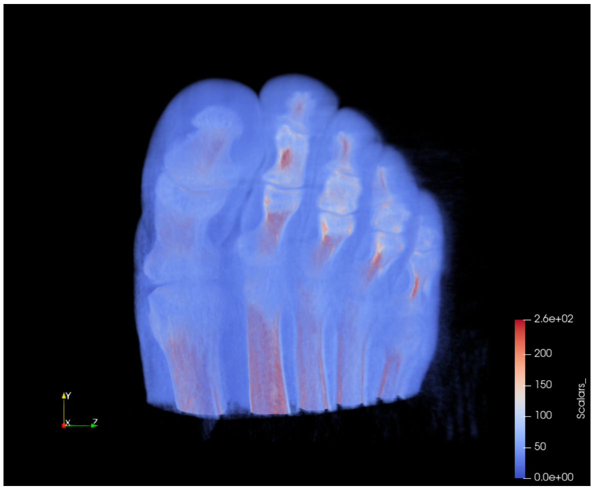


Figure 16: Part 2: Data Visualization with "Cool to Warm" Colormap

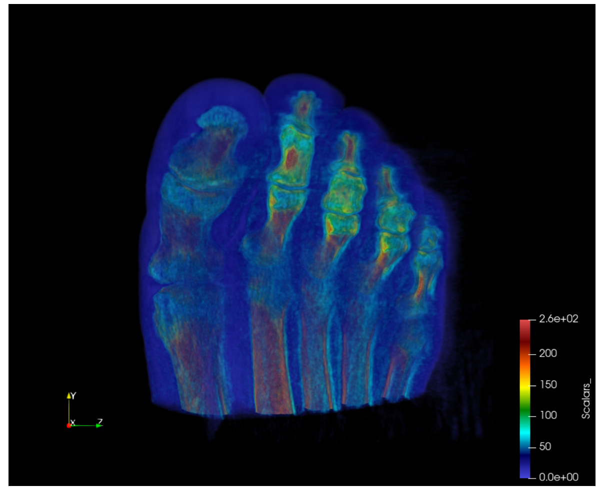


Figure 17: Part 2: Data Visualization with "Rainbow Desaturated" Colormap

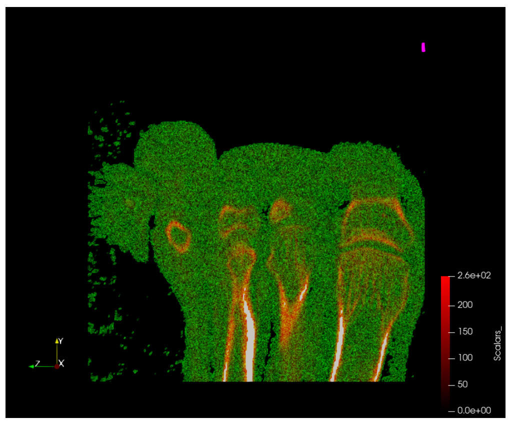


Figure 18: Part 2: Data Visualization Slice

- The first one should be white and by also modifying the transfer function in ParaView and the background, you should be able to just show the bones similar to this.

  Solution. The result of this step can be seen in Figure 19 (I had trouble with using gradient for white or black on the website for a perfect picture, so I used light gray instead).

- Now, make a color map with two color maps within it, one colormap for the bone and another for the flesh.

  Solution. The result of this step can be seen in Figure 20.

- More details can be found here if needed https://sciviscolor.org/colormoves/overview/. Save your colormaps as an XML file and import them to ParaView, there is an Import button under Choose Preset and it will come up at the bottom when searching for all.

- Compare differences of visualization of all four colormaps. State pros and cons of these colormaps. Include images in your report to support your conclusions.

  Solution.

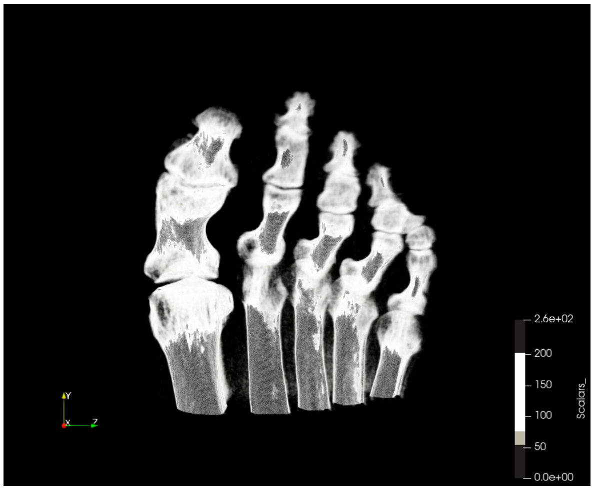


Figure 19: Part 2: Data Visualization with “White and Black” Colormap Generated Through Website

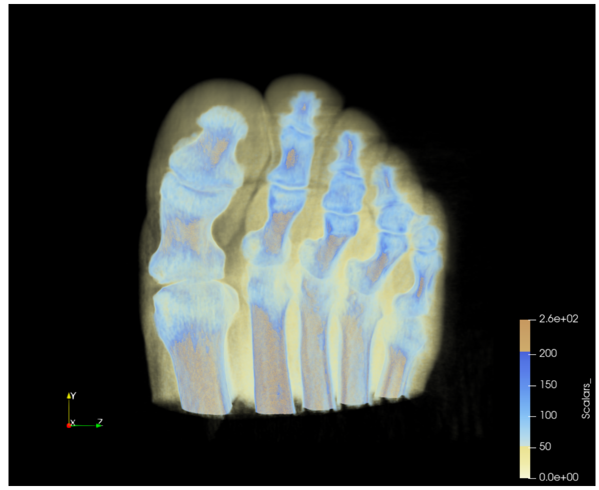


Figure 20: Part 2: Data Visualization with 2 Colormaps Generated Through Website (Yellow and Blue Gradients)

Analysis for each color map based on their visual effectiveness and appropriateness for the data presented.

1. ”Cool to Warm” Colormap

   Pros:

   – This colormap provides a good gradient for viewing details within a continuous range of data values, which is particularly useful for medical imaging like we are presented with.
   – It minimizes visual distortion by using colors that transition smoothly from cool to warm, avoiding the sharp breaks that can mislead in data interpretation.

   Cons:

   – The use of a gradient from blue to red might not be intuitive for all viewers, particularly in distinguishing between critical values unless they are familiar with this specific color mapping.
   – In some contexts, the cool-to-warm transition might not highlight specific features of interest as effectively as more targeted color choices might.

2. ”Rainbow Desaturated” Colormap

   Pros:

   – The desaturated rainbow palette can highlight variations across a broad range of data values, which can be useful for identifying unique features or anomalies in data.
   – This colormap can be visually appealing and can make needed distinct data regions stand out due to varied hues.

   Cons:

   – Rainbow colormaps can create artificial boundaries in the data where none exist, due to the abrupt changes in hue.
   – They can be challenging for people with color vision deficiencies to interpret.

3. ”Black and White” Colormap

   Pros:

   - Maximizes contrast, which is excellent for detecting fine details and variations in density.
   - Universally understandable, does not require viewers to interpret colors and is ideal for colorblindness.

   Cons:

   - Does not utilize color to differentiate between different ranges, which can limit the amount of information conveyed in one glance.
   - May oversimplify the visualization, losing some nuances that color can provide.

4. Dual Colormap (Yellow and Blue Gradients)

   Pros:

   - Using two different colormaps for different aspects of the data (for us: bone and soft tissue) can clarify the separation between different types of materials or conditions in the scan.
   - Allows for tailored color usage that can highlight specific features or anomalies relevant to particular needs.

   Cons:

   - May require additional explanation to ensure that the viewer understands what each color represents, particularly if the colormaps encode different types of information.

Conclusion.

Each colormap offers unique benefits and drawbacks, making them suitable for different visualization needs.

5 Conclusion

In this assignment, I delved deeply into the challenges and techniques of uncertainty visualization across various types of data fields, including isocontours, vector fields, and exploring the impact of perceptualization and color mapping on data interpretation. The insights gained and the practical applications of these visualization techniques significantly enhanced my understanding and skills in handling complex visualization scenarios.

Uncertainty Visualization of Isocontours and Vector Fields:

The creation of spaghetti plots and the utilization of contour box plots provided a profound understanding of the uncertainty inherent in simulation datasets. These techniques allowed for the detailed portrayal of variability and inconsistency within the data, showcasing where models agree or diverge significantly.

In vector field visualization, the use of arrow glyphs and color-mapped surfaces illuminated the directionality and magnitude differences within the wind data. The application of different color maps to represent vector angles further deepened the perceptual impact of these visualizations, making subtle variations more noticeable and interpretable.

Reading and Application of Visualization Techniques:

The exploration of preattentive processing emphasized the importance of visual features that allow rapid, subconscious data processing, which is crucial for effective data visualization where quick insights are often necessary.

The examination and implementation of various color maps underscored their significance in enhancing or diminishing the perceptual quality of visual data. Through practical application, I experienced firsthand how different color maps affect the readability and interpretive accuracy of data, leading to a better appreciation of their strategic use in visualizations.

Perceptualization Enhancements:

By revisiting previous visualizations, I identified opportunities for improving perceptual significance, such as enhancing depth perception and feature differentiation in 2D scalar fields. The re-visualization efforts in ParaView using advanced color mapping and shading techniques demonstrated substantial improvements, offering clearer, more intuitive visual representations.

Synthesis of Visual Elements Across Multiple Fields:

The assignment facilitated a comprehensive integration of techniques ranging from noise filtering and frequency analysis to advanced glyph representation and contour enhancements. This synthesis not only demonstrated the robust capabilities of ParaView and similar tools but also solidified my understanding of when and how to apply different visualization strategies to reveal crucial insights into complex data sets.

The cumulative knowledge and skills developed through this assignment are invaluable, extending beyond academic exercises to practical applications in real-world data analysis. This assignment has profoundly expanded my toolkit for data visualization and sparked further interest in exploring new and emerging techniques in this dynamic field.

code: https://github.com/milenabel/CS6635-VisforScD