A New Technique for Depicting Terrain Relief

Leland Brown

October 15, 2010

Agenda

- Motivation
- Design Goals
- Examples (comparison to hillshading)
- Benefits
- Mathematical Details
- Practical Considerations
- Lots More Examples

Motivation for a New Shading Algorithm

- An intuitive grasp of a region's geography requires a mental framework on which to build a cognitive map
- Examples from my own experience
  - Los Angeles: freeway system
  - México City: metro system
  - San Gabriel Mountains: canyon and ridge network
- Existing cartographic methods for terrain structure
  - Simply labeling trails, roads, and peaks does not support this insight
  - Hillshading does not emphasize the drainage structure
  - Contour lines show the needed information but require interpretation to read the ridges and valleys
  - Stream vectors may lack detail or be too busy, depending on the map scale
  - None of these shows the information well across a wide range of scales

3

This project grew out of my love for hiking in the local mountains north of Los Angeles. As I explored different areas of the mountains, I found it difficult to get a mental picture of the overall layout of the terrain.

I realized that in the city, I know my way around by means of the freeway system, which provides a framework on which to locate places in my mental map. In the mountains, I didn't have such a framework, so I had no way to piece together the various isolated peaks and trails into a coherent mental map.

One day I had the insight that the analog of the freeway network is the network of ridges and canyons in the mountain range. But I didn't find the maps or other information I had to be sufficiently helpful in discerning that structure. So I set out to find a way to visualize the information I was looking for.

Since I'm not artistically inclined, I prefer an algorithmic solution that can automatically produce the type of map I want from digital elevation data.

“Texture Shading” – Design Goals

- Isotropy
  - No directional dependence
- Scale invariance
  - Applying the algorithm on a smaller region will be the same as zooming in on the results from a larger region (“self-generalizing”)
  - Only a rescaling of the vertical axis (grayscale values) is needed
  - At any resolution, large features dominate but small features are present
  - Zooming in brings the small features to the forefront, with even smaller features now visible
- A single family of linear operators meets these two criteria
  - Called “fractional Laplacian” operators
  - Parameter α tunes balance of details vs. major features regardless of scale
- Other, nonlinear solutions also exist – e.g., uniform sky illumination [1]

[1] Kennelly, P., and J. Stewart, “A Uniform Sky Model to Enhance Shading of Terrain and Urban Elevation Models,” *Cartography and Geographic Information Science* 2006, 33(1), 21-36.

I identified two criteria I wanted these maps to satisfy:

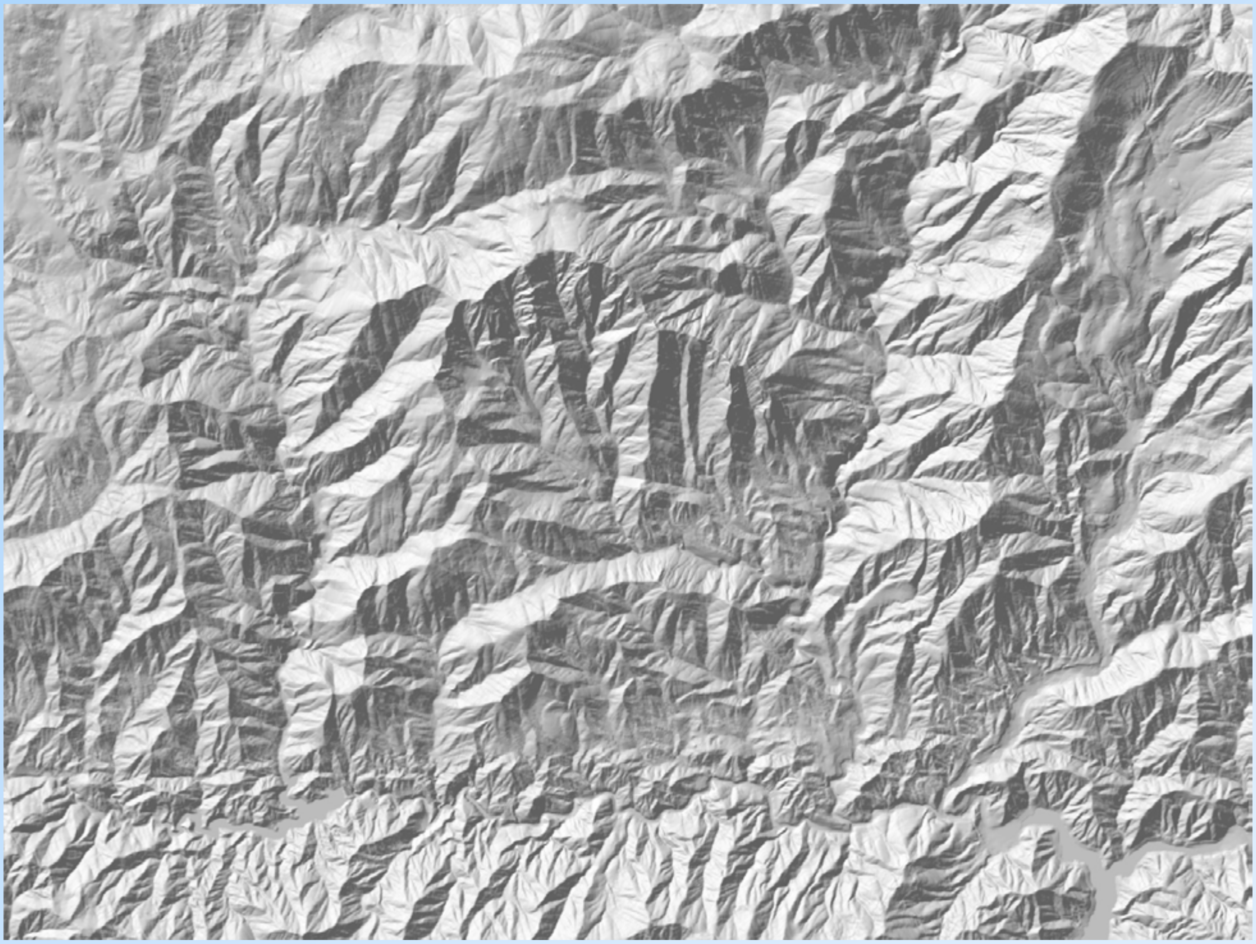
1. Isotropy - that is, the way a particular feature appears should be independent of its orientation on the landscape; and

2. Scale invariance - in the sense that a single image should support a wide variety of scales by simply zooming the image in and out. I refer to this as a sort of “self-generalizing” property.

Among linear operators, a unique family of solutions meets these criteria, called “fractional Laplacian” operators, with a single tuning parameter I’ll call “alpha.” These are related to but different than the regular Laplacian often used in image processing as an edge detector. Based on a Google search, the fractional Laplacian seems to be used primarily in fluid dynamics. As far as I know, this is the first time it’s been suggested for use in terrain representation or image processing.

Besides linear operators, other techniques may satisfy these characteristics, such as the uniform sky illumination model developed by Patrick Kennelly and James Stewart.

Notional Example: Hillshading, not generalized (too much detail)



San Gabriel Wilderness, CA Scale: 11.5 mi. x 8.6 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

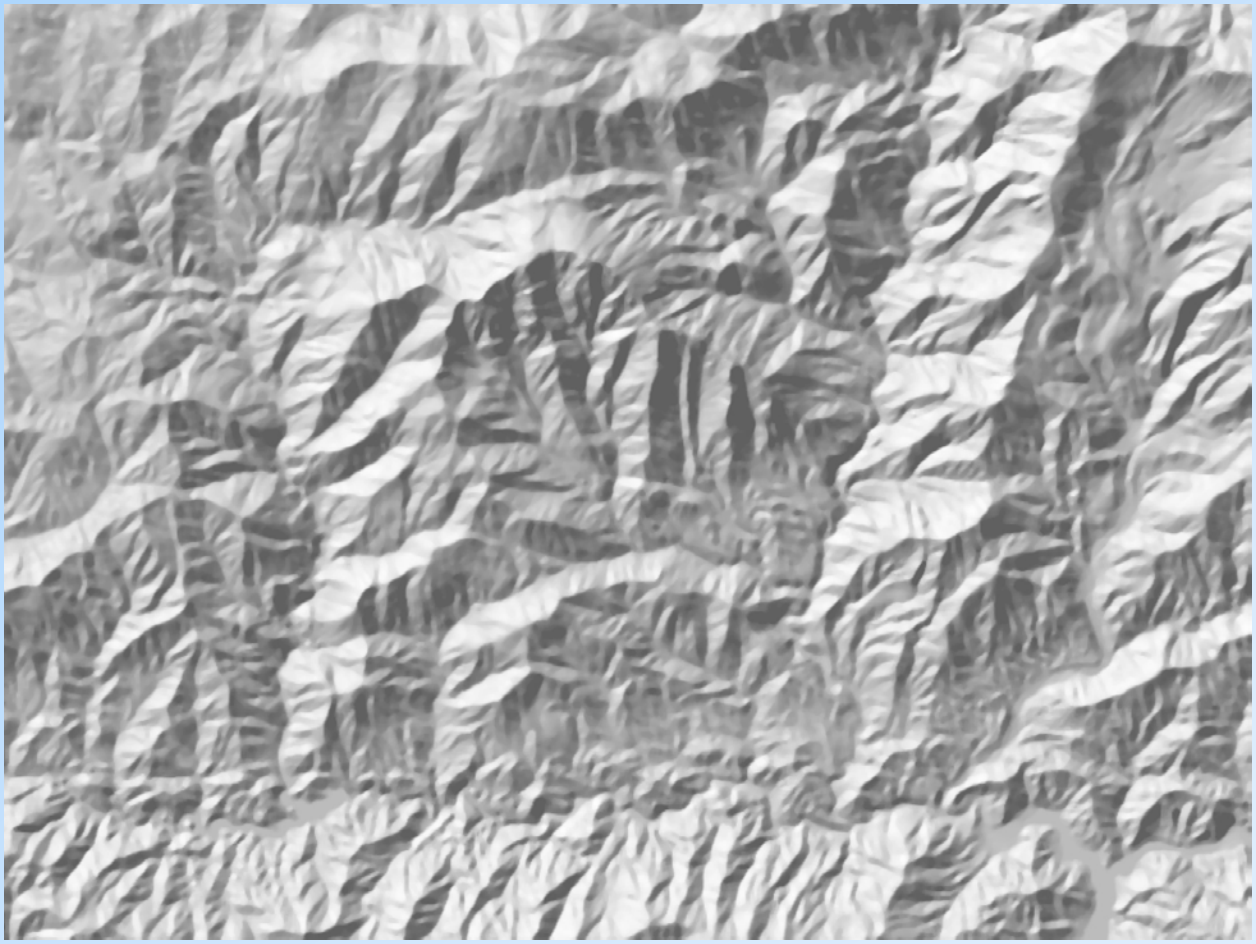
5

This is my simplistic attempt to show an example of standard hillshading. I haven't tried to do anything fancy here, but it will suffice for illustration purposes. This shows a particularly rugged area of wilderness in my local mountains.

This first example is a naïve implementation that simply determines the shading at each point based on the local slope at that point. Since slope is a local operator, it can show rapid variation from pixel to pixel, resulting in an image that's too cluttered with details. It has the appearance of crinkled aluminum foil.

Typically we would generalize this map by smoothing out the terrain to reduce the detail.

Hillshading, generalized



San Gabriel Wilderness, CA Scale: 11.5 mi. x 8.6 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

6

This image shows an example of hillshading after generalization. The detail has been reduced, making the map a bit easier to read. But still I find myself hard-pressed from this map to answer some of these basic questions that I might be interested in:

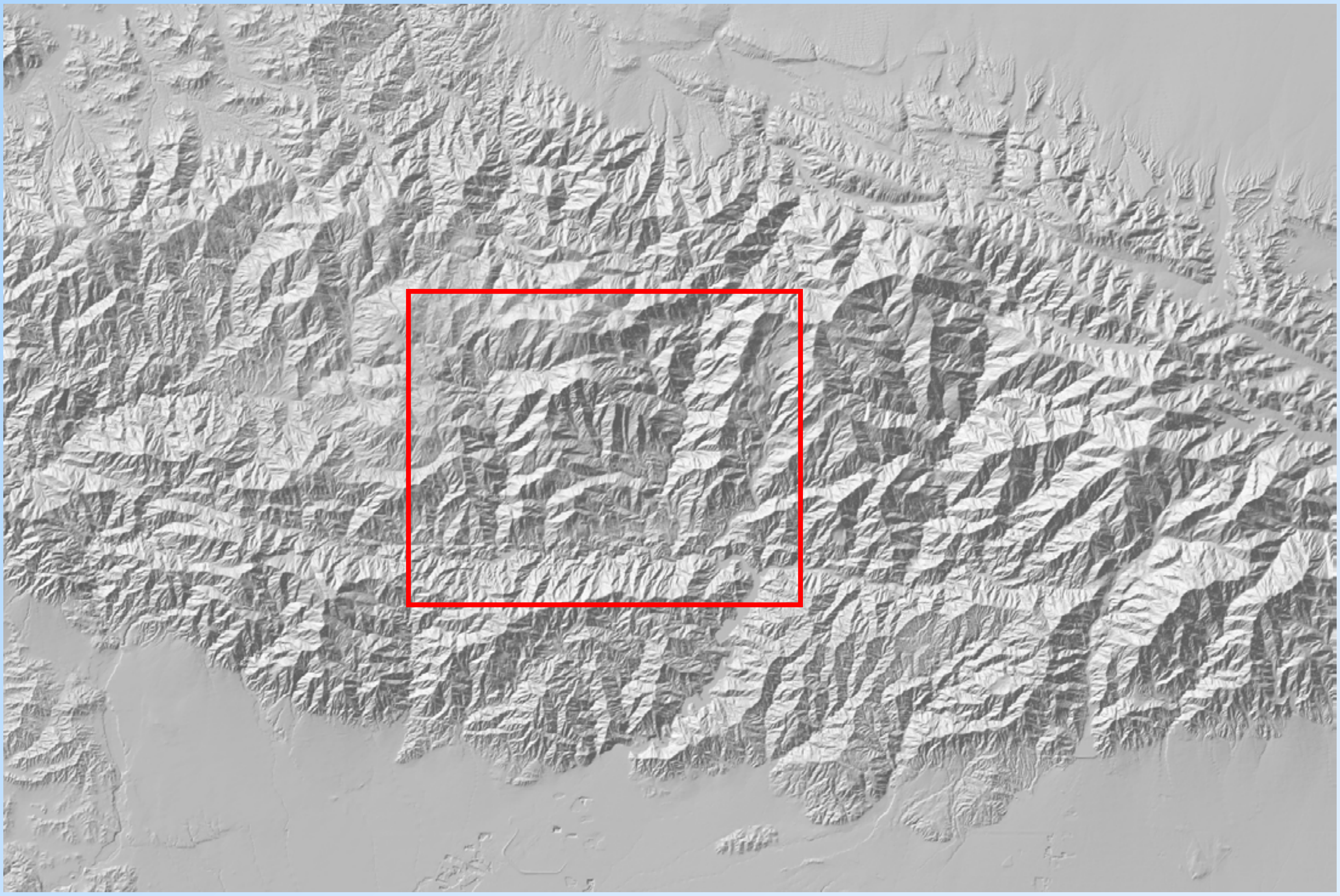
Where are the major canyons?

What is the direction of drainage?

Which are the highest mountains?

Where are the significant saddle points?

Hillshading, zoomed out (too much detail again)



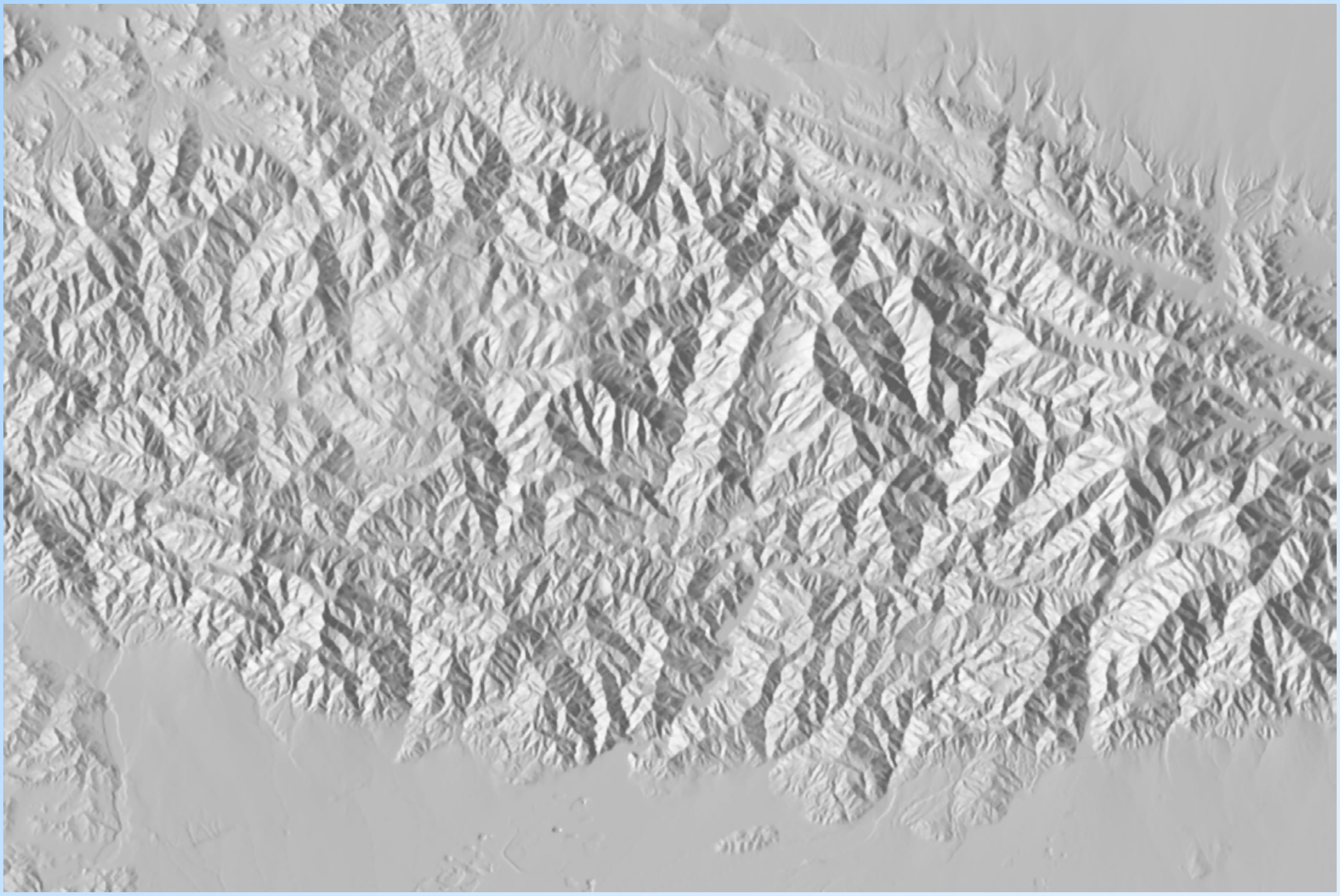
San Gabriel Mountains, CA — Scale: 39 mi. x 26 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

7

Now if we zoom out from the previous image while keeping the scale of detail the same, we find that the map is too cluttered again. We would need to redo the generalization at the new scale, requiring us to redo the shading computation.

A second difficulty with hillshading is the anisotropy, or directional dependence. In this image, with the illumination from the northwest, a major east-west canyon is visible running across just below the center of the image. It actually follows an ancient earthquake fault.

Hillshading, sun from west (E-W canyon now obscured)



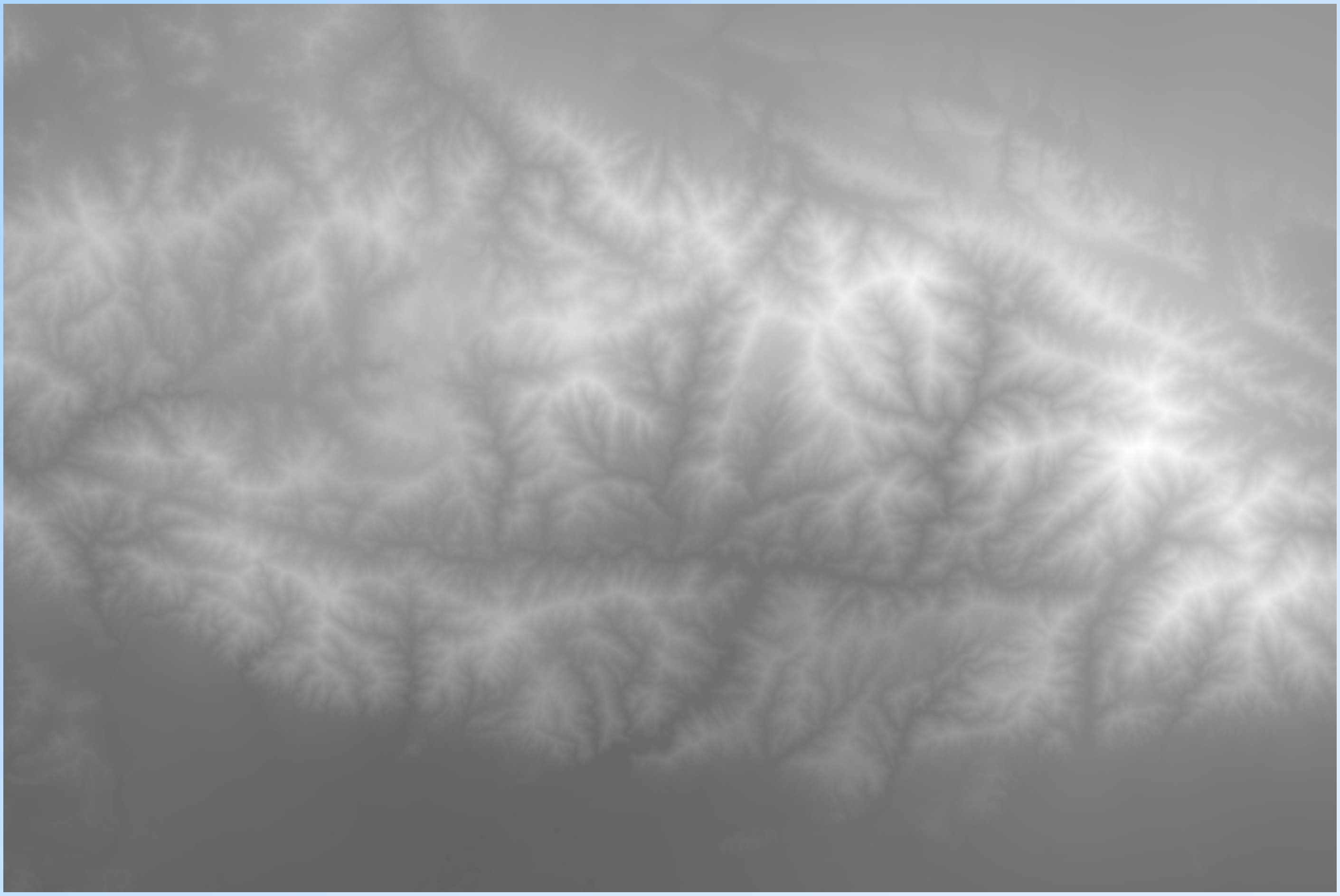
San Gabriel Mountains, CA Scale: 39 mi. x 26 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

8

But if we move the illumination to be from the west, that east-west canyon becomes nearly invisible. And in fact, a northwest-trending canyon near the top of the image becomes apparent, which we didn't see in the previous image.

These are some of the issues I would like to eliminate with the new shading technique. As it's useful to have a name for the new method, I'll refer to it provisionally as "texture shading" (at the risk of creating confusion with "texture mapping," which is a very different thing).

Raw elevation – hypsometric shading (shows large structure)



San Gabriel Mountains, CA (α = 0) Scale: 39 mi. x 26 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

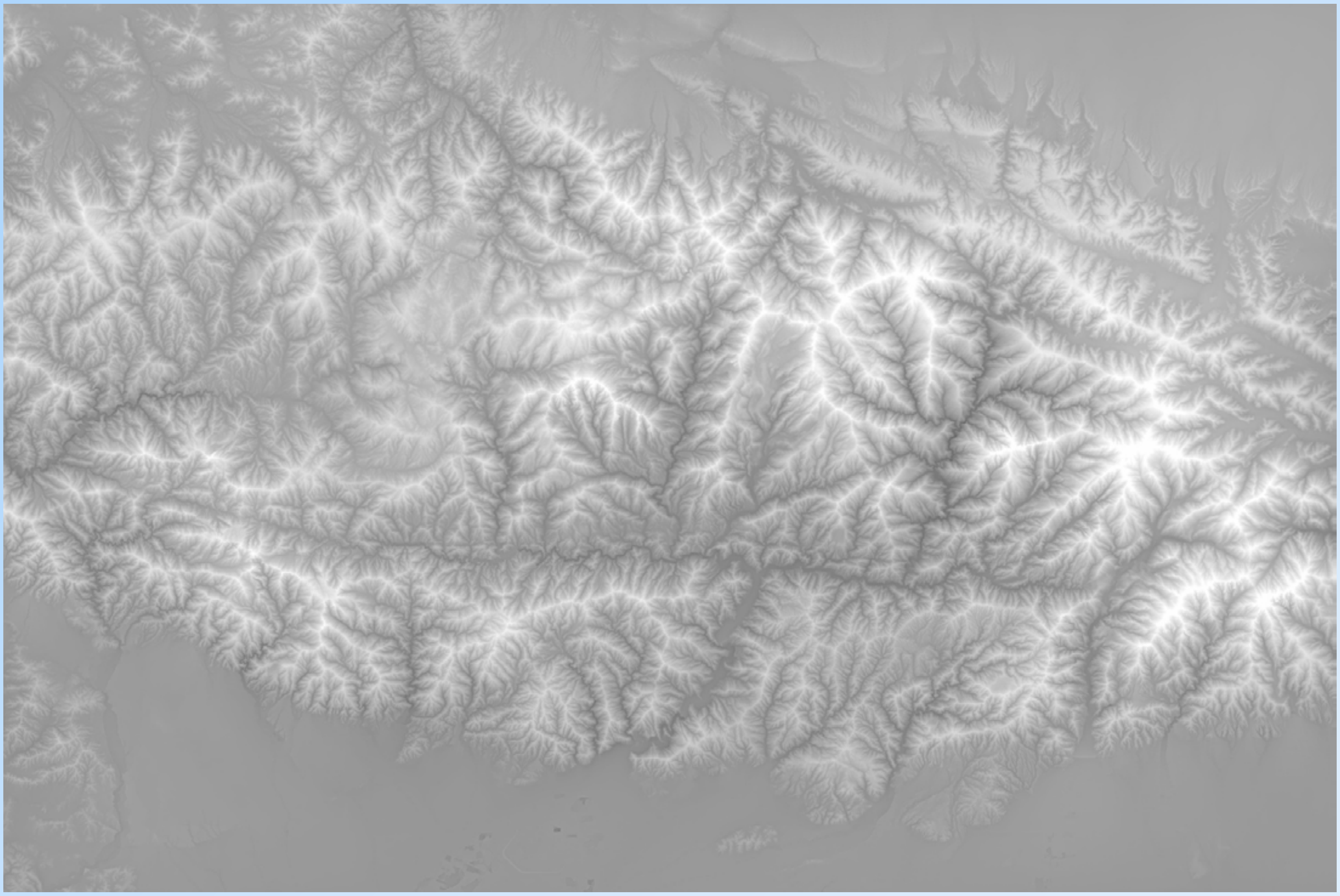
9

Before showing the results of texture shading - this image shows raw elevation data plotted directly on a grayscale, with white for high areas and dark for low areas. Here we can see all the major structures as well as the overall shape of the terrain going up and over the range.

This is in some sense the missing information from the hillshaded images. For that reason, this data is often overlaid on hillshading either with color as hypsometric tints, or using an atmospheric perspective effect.

Texture shading takes a different approach. This image is the starting point for texture shading; from here the problem is not excess detail, but rather not enough detail. Major features are visible, but all the detail is hidden in small variations of gray, giving the image an appearance of being fuzzy or blurred. Visually, texture shading has an effect like contrast enhancement, bringing the details more into focus.

Texture shading with $\alpha = 1/2$ (details enhanced)



San Gabriel Mountains, CA ($\alpha = 1/2$) Scale: 39 mi. x 26 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

10

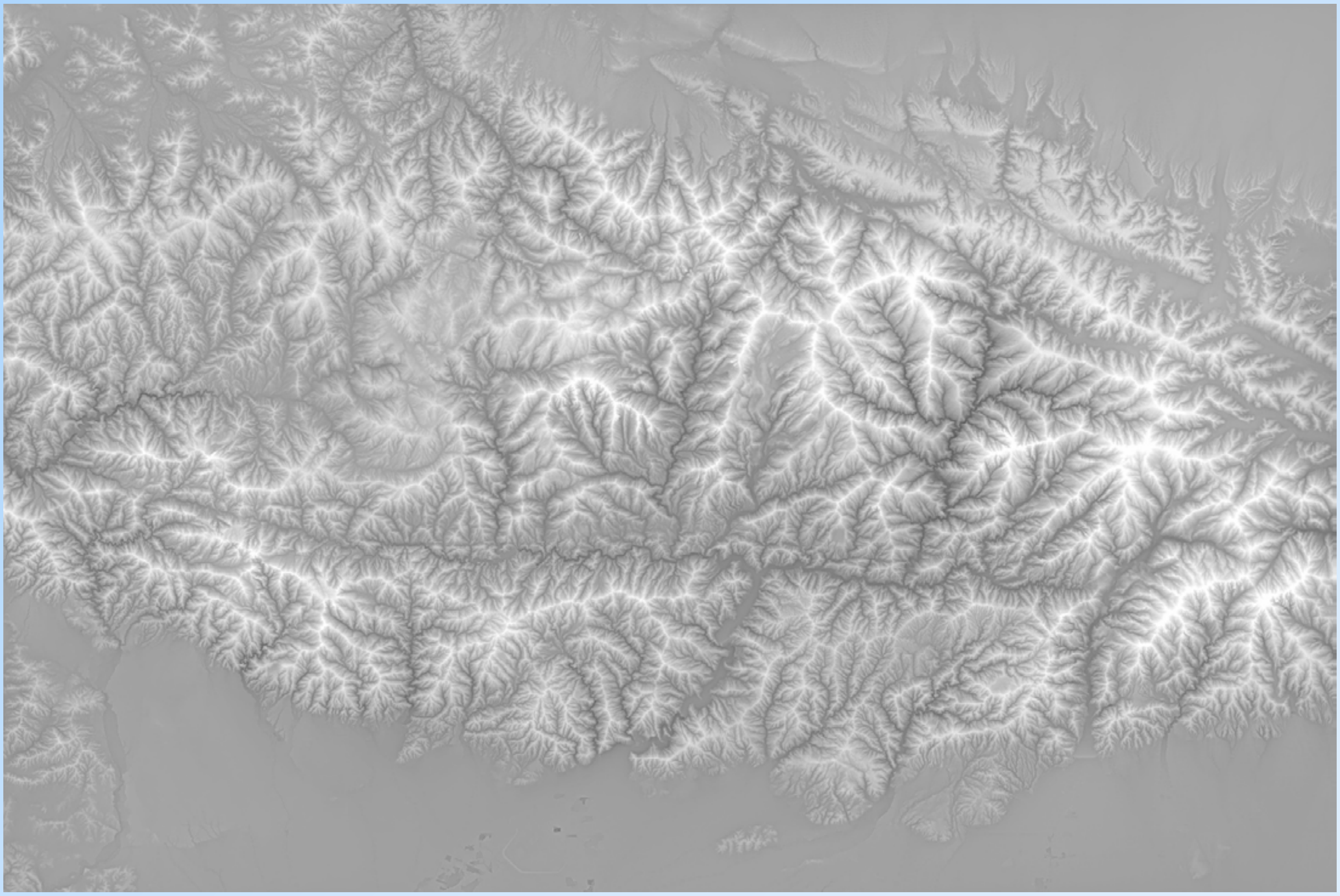
Here is the result of texture shading on this terrain, using an alpha of 1/2. Notice that many more details are visible, yet the major features still stand out. The overall shape of the mountain range is still apparent, though not as clearly as in the previous image. These tradeoffs are controlled by the parameter alpha (as we'll see in the next slide).

What texture shading does, in essence, is show the *relative* height of each point, relative to terrain nearby - where "nearby" is defined as a weighted average over all distances, giving the most weight to closer terrain but also considering data out to very large distances, to a lesser and lesser extent (following a power law).

Thus, light shading primarily indicates points high relative to things around - i.e., ridges - but also includes some indication of absolute elevation. Likewise, the darkest points generally indicate the deepest canyons. Minor ridges and canyons show up as more subtle variations on the local shading, in a manner reminiscent of the resolution bumping technique developed by Tom Patterson of the U.S. National Park Service.

With this map I've accomplished my original goal - to have a better understanding of the topography of the mountains where I hike.

Texture shading with $\alpha = 2 \,/\, 3$ (more details)

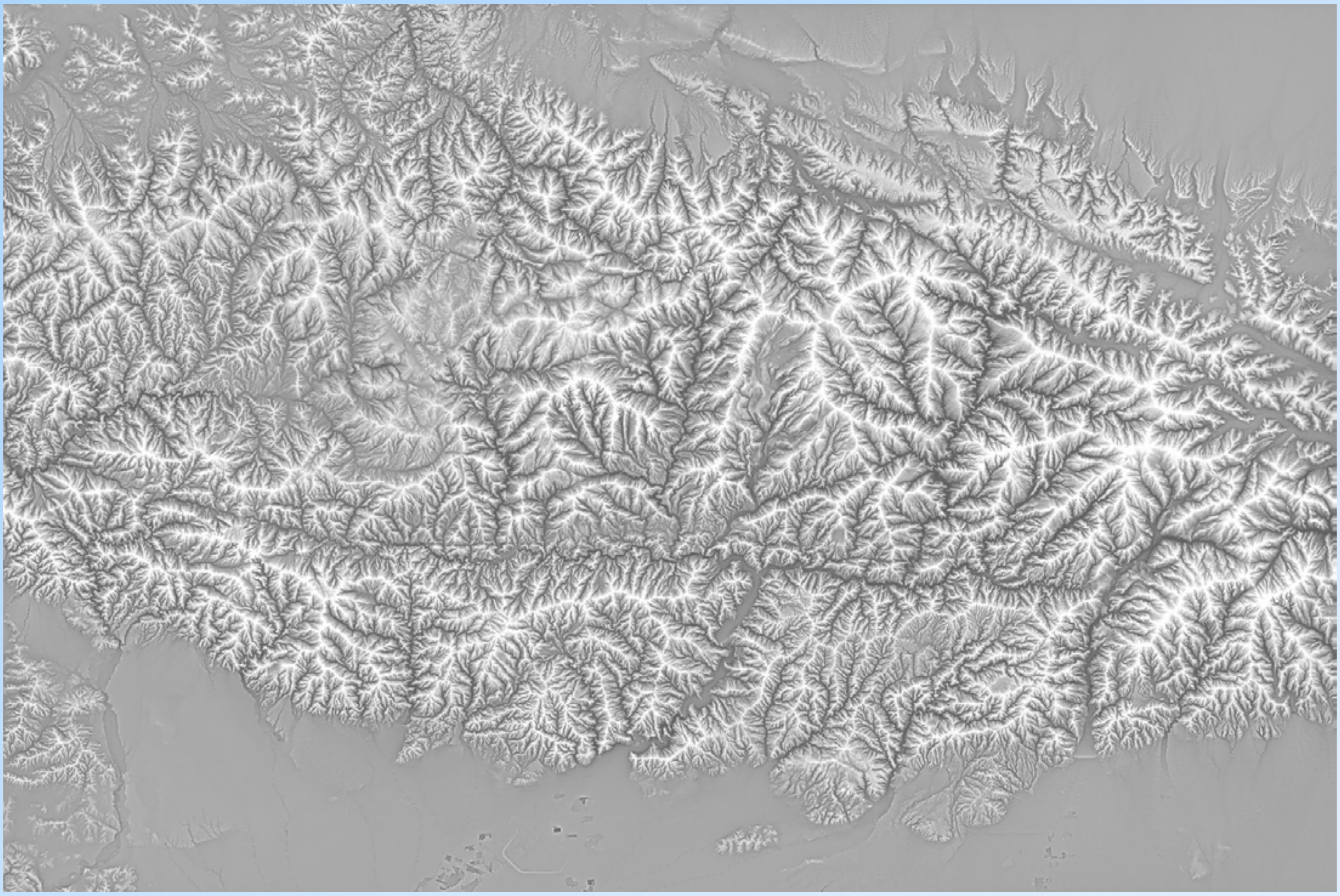


San Gabriel Mountains, CA ($\alpha = 2 \,/\, 3$) Scale: 39 mi. x 26 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

11

Here is texture shading with a higher alpha of 2/3. What this does essentially is turn up the gain on the details. The image is a bit more cluttered, and the overall shape has a flatter appearance, but it also looks more sharp and crisp. The choice between these two values of alpha is a matter of preference, and of which information is desired to be highlighted in a particular map.

Texture shading with α = 1 (details overwhelm)



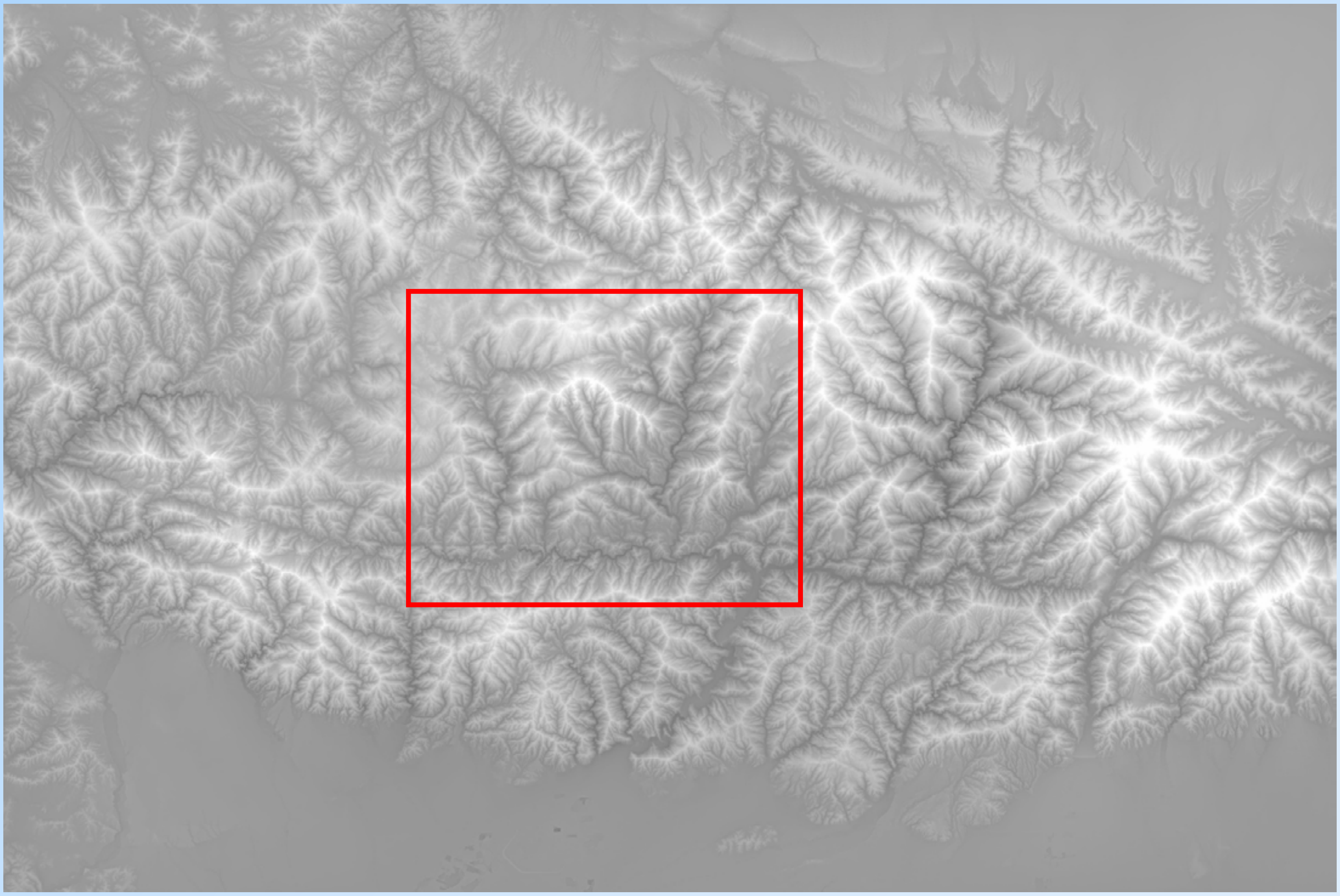
San Gabriel Mountains, CA (α = 1) Scale: 39 mi. x 26 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

12

In principle, we can drive alpha even higher - this shows alpha = 1. Now the overall shape looks very flat, and the details dominate so as to create clutter. This is only for illustration - it would not likely make a useful map.

If we were to increase alpha all the way up to 2, we would get an ordinary Laplacian operator. As mentioned before, it's an edge detector, and since this is not a photographic image but a rugged terrain, it would find "edges" on nearly every pixel, resulting in so much clutter as to look like random noise or static.

Texture shading – zoom in

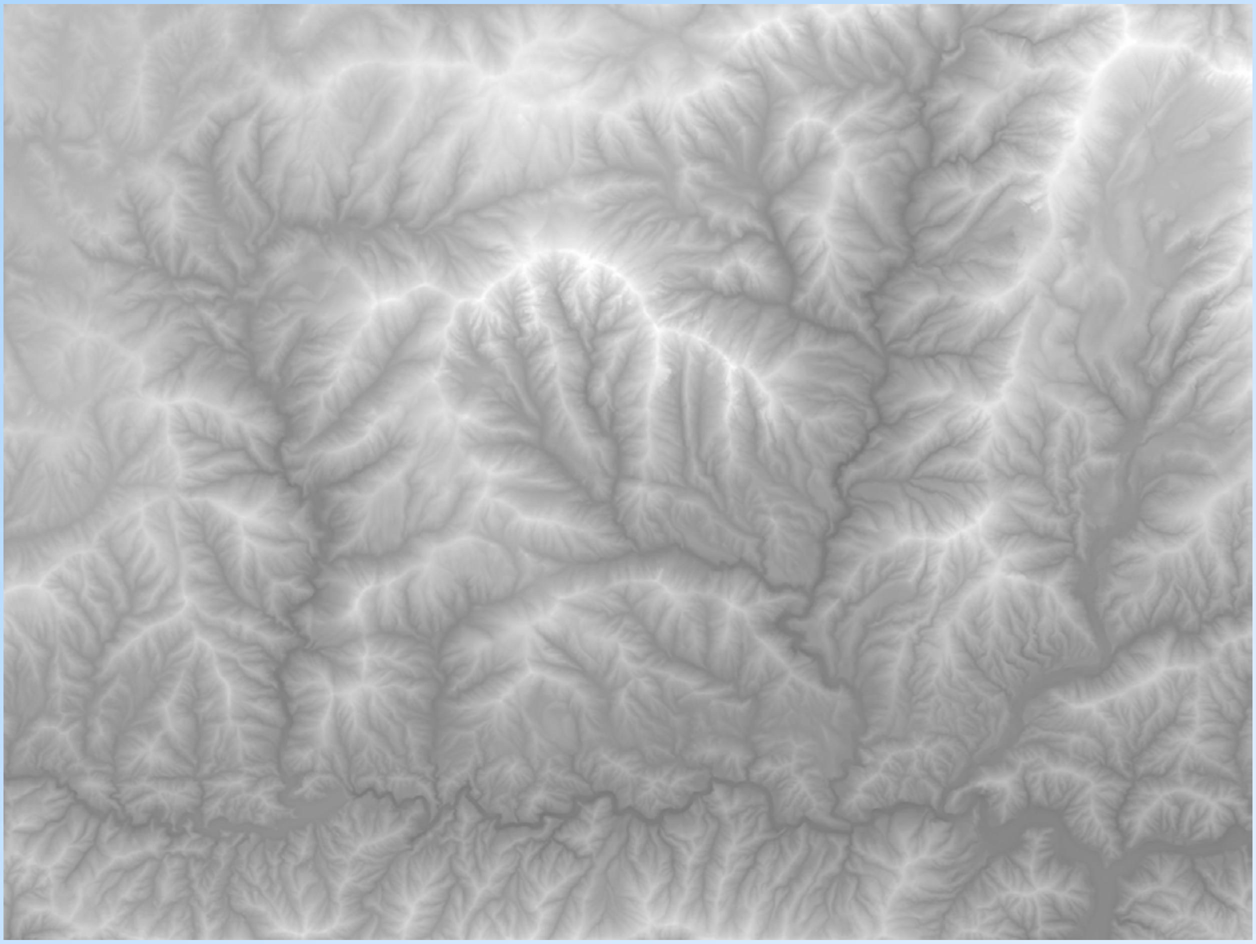


San Gabriel Mountains, CA ($\alpha = 1/2$) Scale: 39 mi. x 26 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

13

Going back to the example with alpha = 1/2, let’s zoom in on the region we started with.

Texture shading with $\alpha = 1/2$



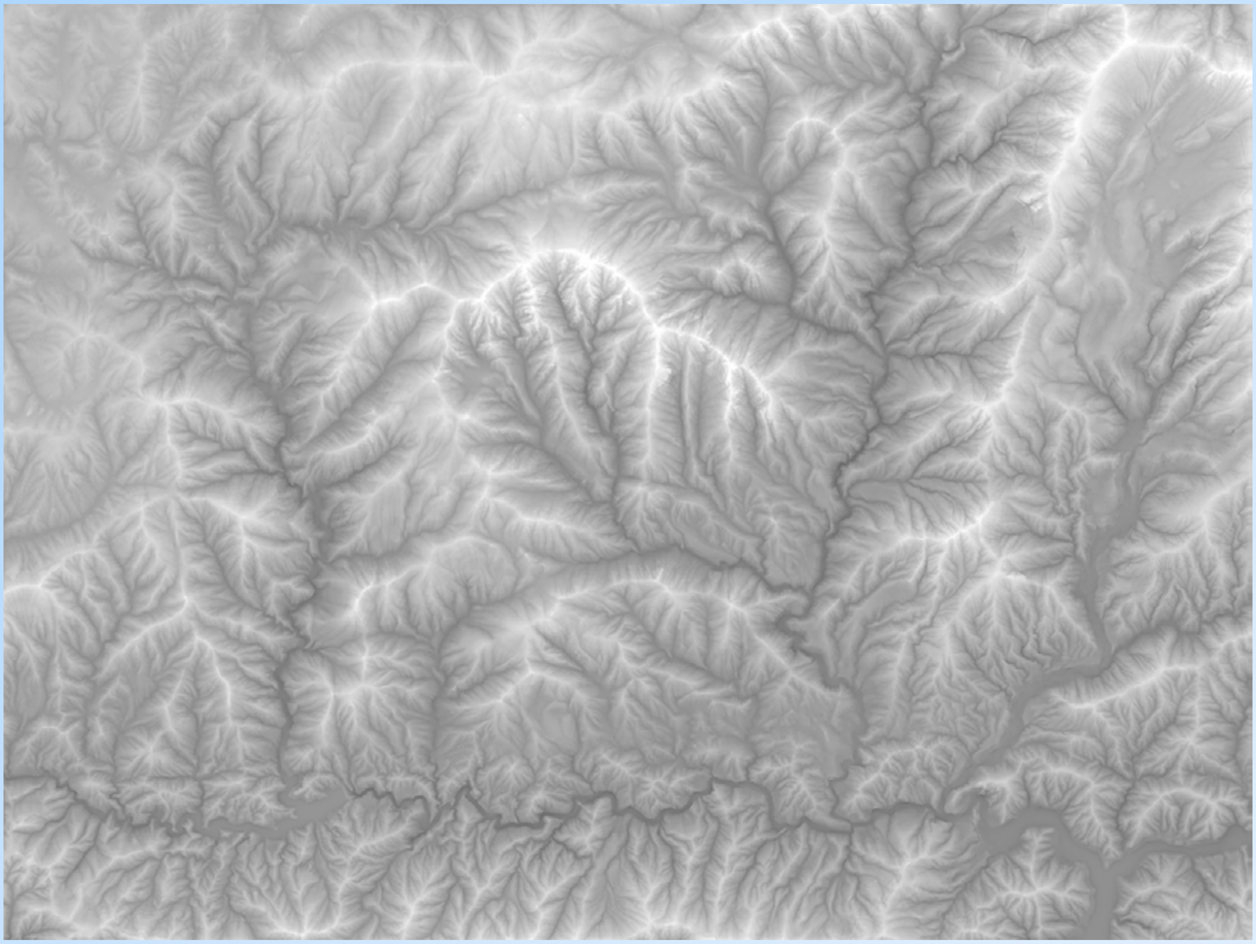
San Gabriel Wilderness, CA ($\alpha = 1/2$) Scale: 11.5 mi. x 8.6 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

14

Now what were details in the previous image become the prominent features at this scale, while even smaller details become visible. This balance of emphasis is maintained, without having to change the generalization or do any recalculation - we simply zoom the image.

We can continue this process as long as we want, up to the limit of resolution of the underlying data. If we zoom in on an area of less relief, we'll probably want to change the grayscale limits to use the full range of light to dark instead of just middle grays (and this will also bring more subtle details into view). Other than that rescaling, however, no changes or recalculation would be needed.

Texture shading with $\alpha = 2/3$



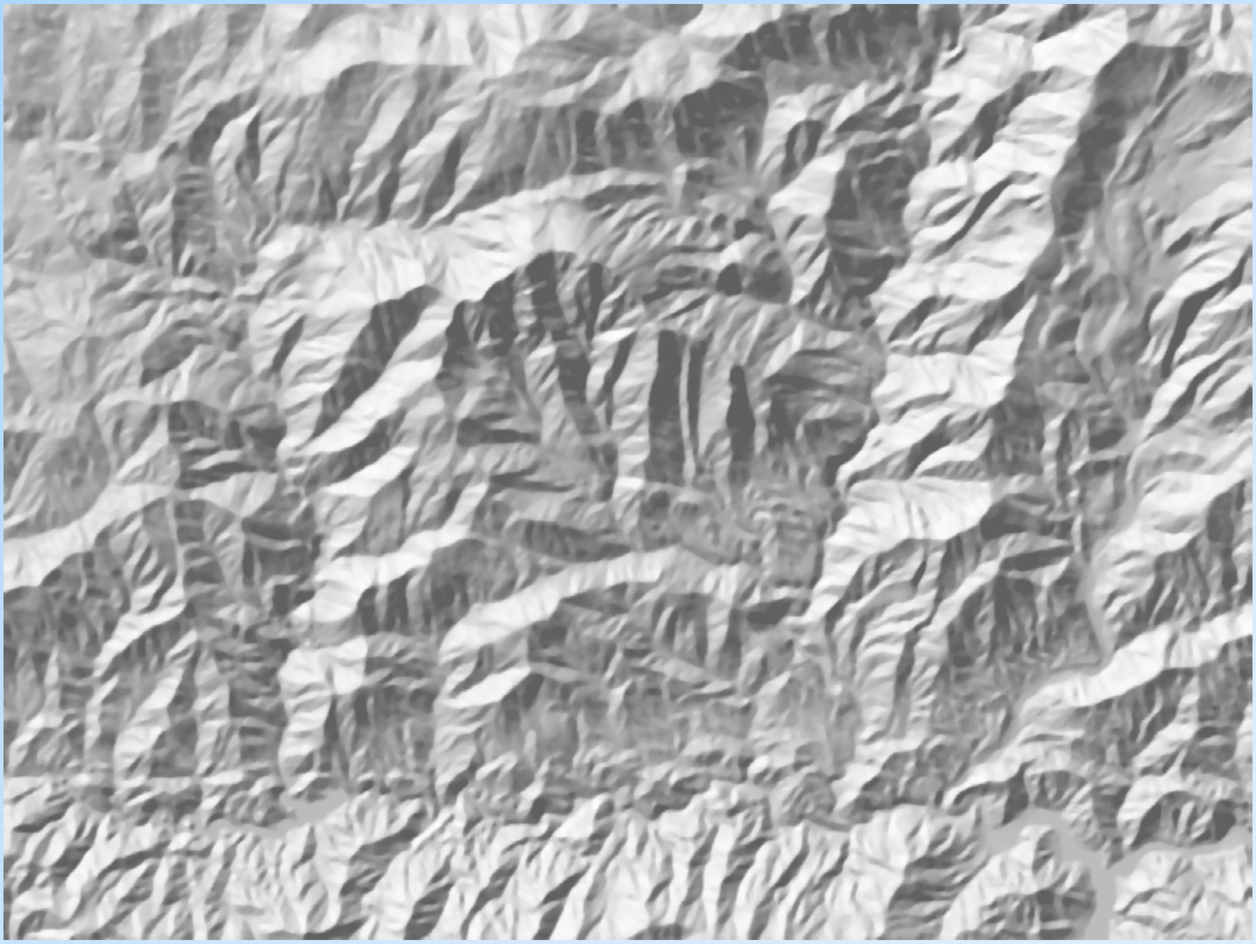
San Gabriel Wilderness, CA ($\alpha = 2/3$) Scale: 11.5 mi. x 8.6 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

15

This is the same area with an alpha of 2/3.

The next slide shows the original generalized hillshading of the same area for comparison …

Hillshading for comparison

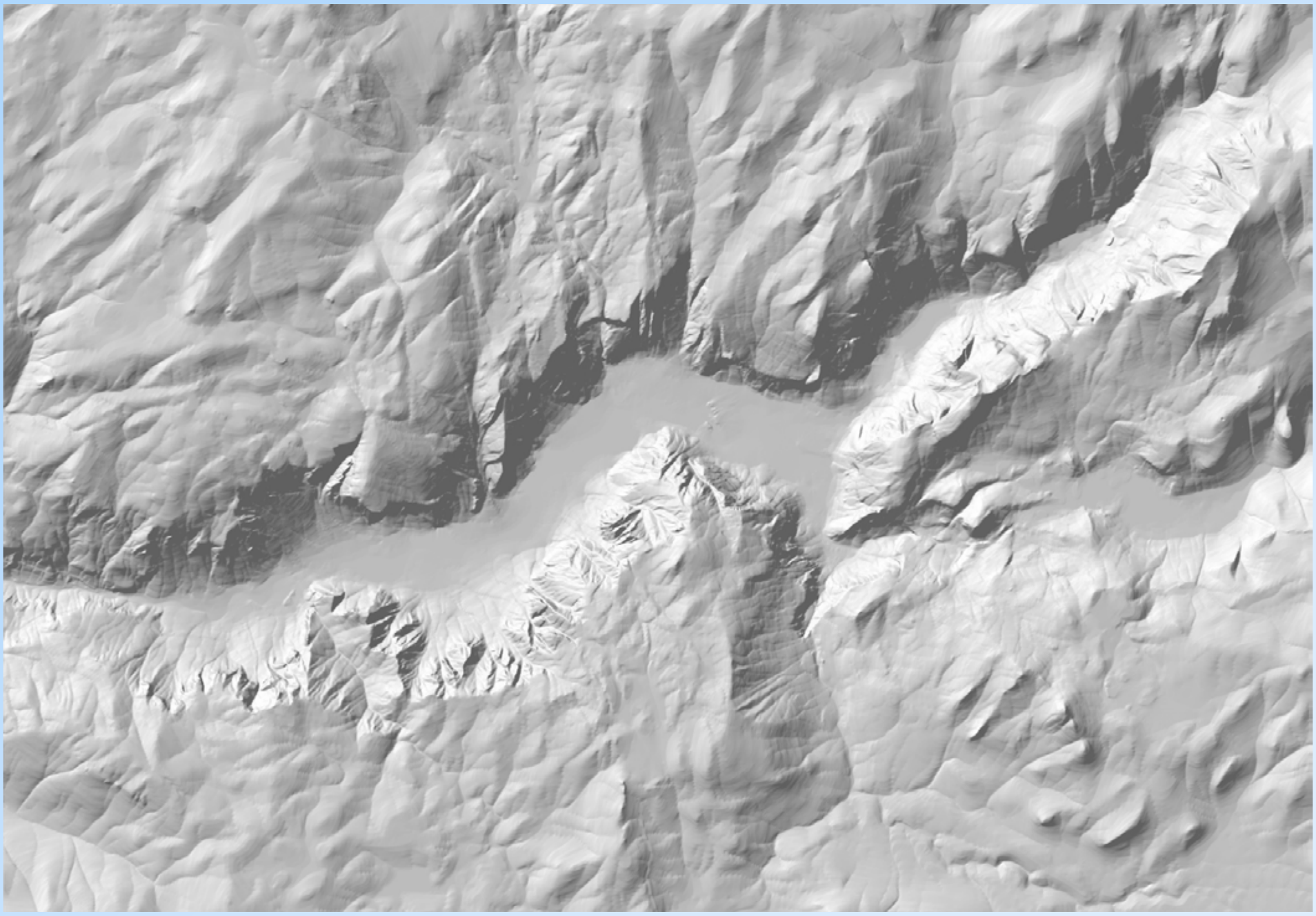


San Gabriel Wilderness, CA Scale: 11.5 mi. x 8.6 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

16

I think it's clear here that the information exhibited by the two approaches is very different.

Yosemite Valley – Hillshading: Can you find Half Dome?



Yosemite Valley, CA Scale: 12.4 mi. x 8.6 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

17

Here's another location that's more well-known: Yosemite Valley. This image is rendered using standard hillshading (without generalization).

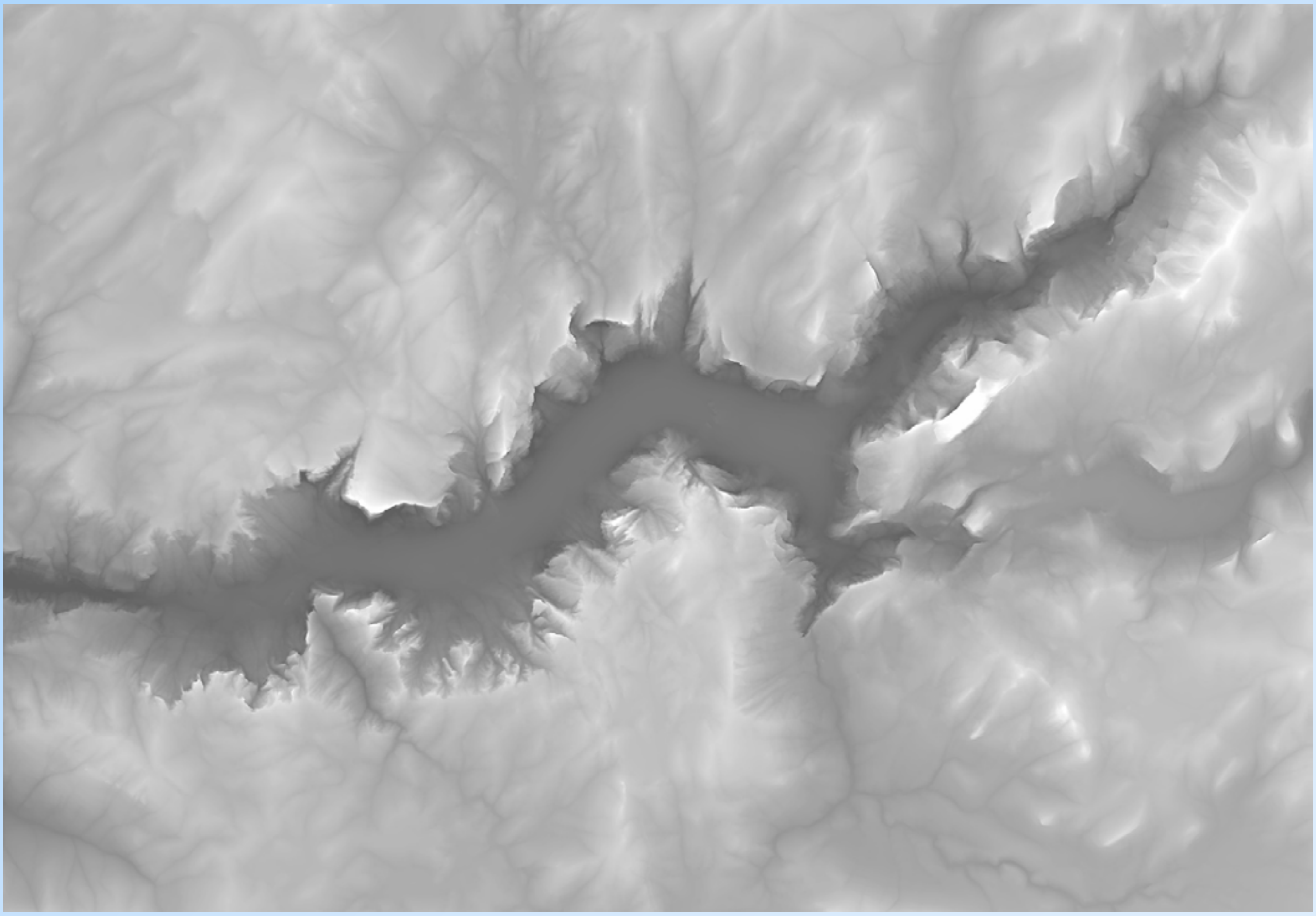
Besides the valley itself, by far the most prominent feature in this terrain is Half Dome. But in this image, it's almost impossible to find Half Dome even if you know where to look for it. There are two reasons for this:

First, the sheer face of Half Dome faces northwest, toward the light source - so it's illuminated similarly to much of the more horizontal terrain nearby. Thus, there is very little contrast on the face.

Second, since the face is nearly vertical, its width on the map is very narrow - perhaps only a single pixel, or even zero pixels wide here. So even if there were more contrast, it would only show up as a very narrow feature or not at all.

(Adding hypsometric tinting to this map would likely make Half Dome more visible, but that would also mean the color space has been used up and no longer available for displaying other information on the map.)

Yosemite Valley – Texture Shading ($\alpha = 1/2$)

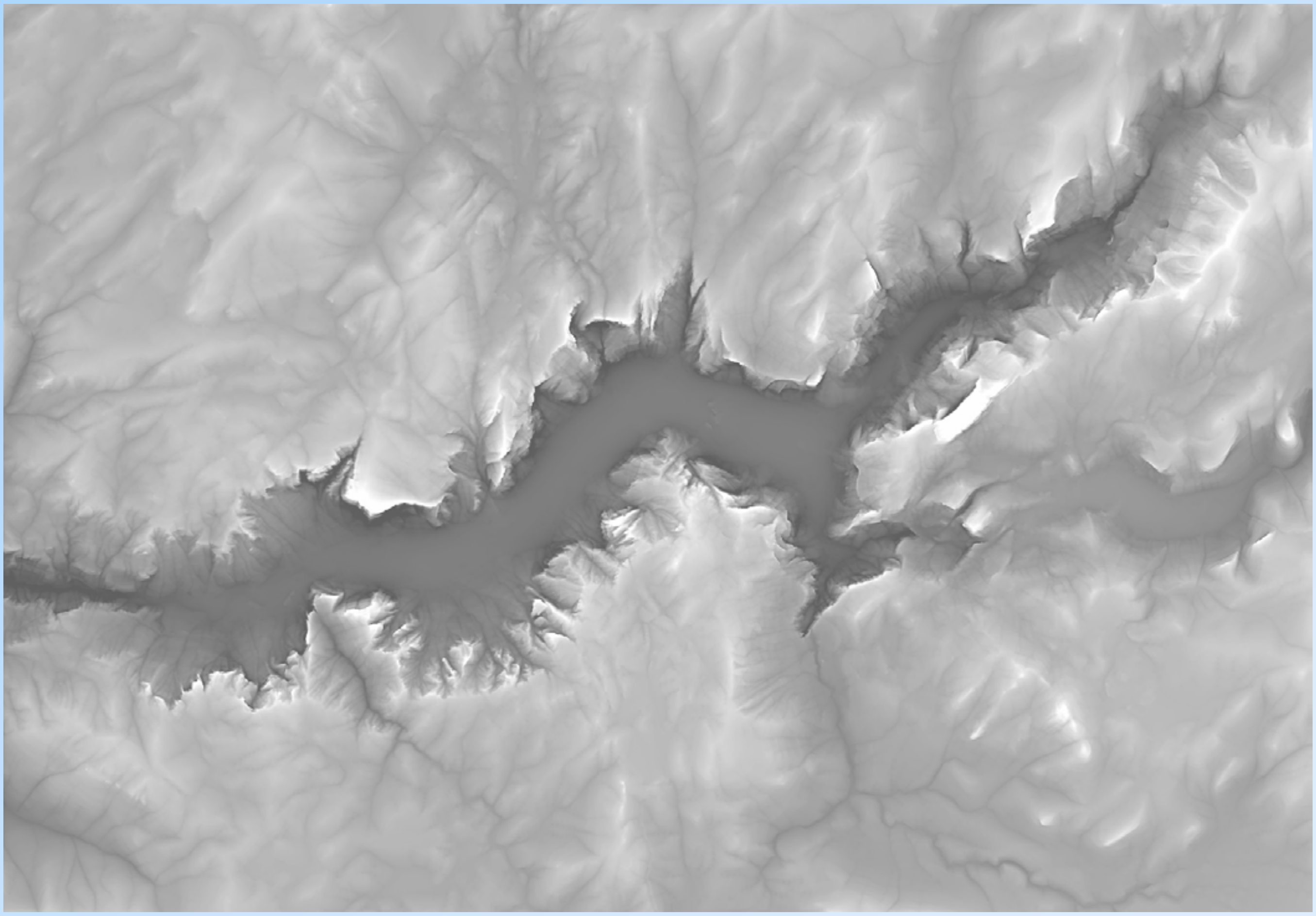


Yosemite Valley, CA ($\alpha = 1/2$) Scale: 12.4 mi. x 8.6 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

18

With texture shading, Half Dome becomes easier to see (a diagonal white band near the eastern end of the valley). The texture shading algorithm guarantees that any abrupt change in elevation will show up as a sharp contrast between dark and light regions in the vicinity of the edge.

Yosemite Valley – Texture Shading (α = 2⁄3)



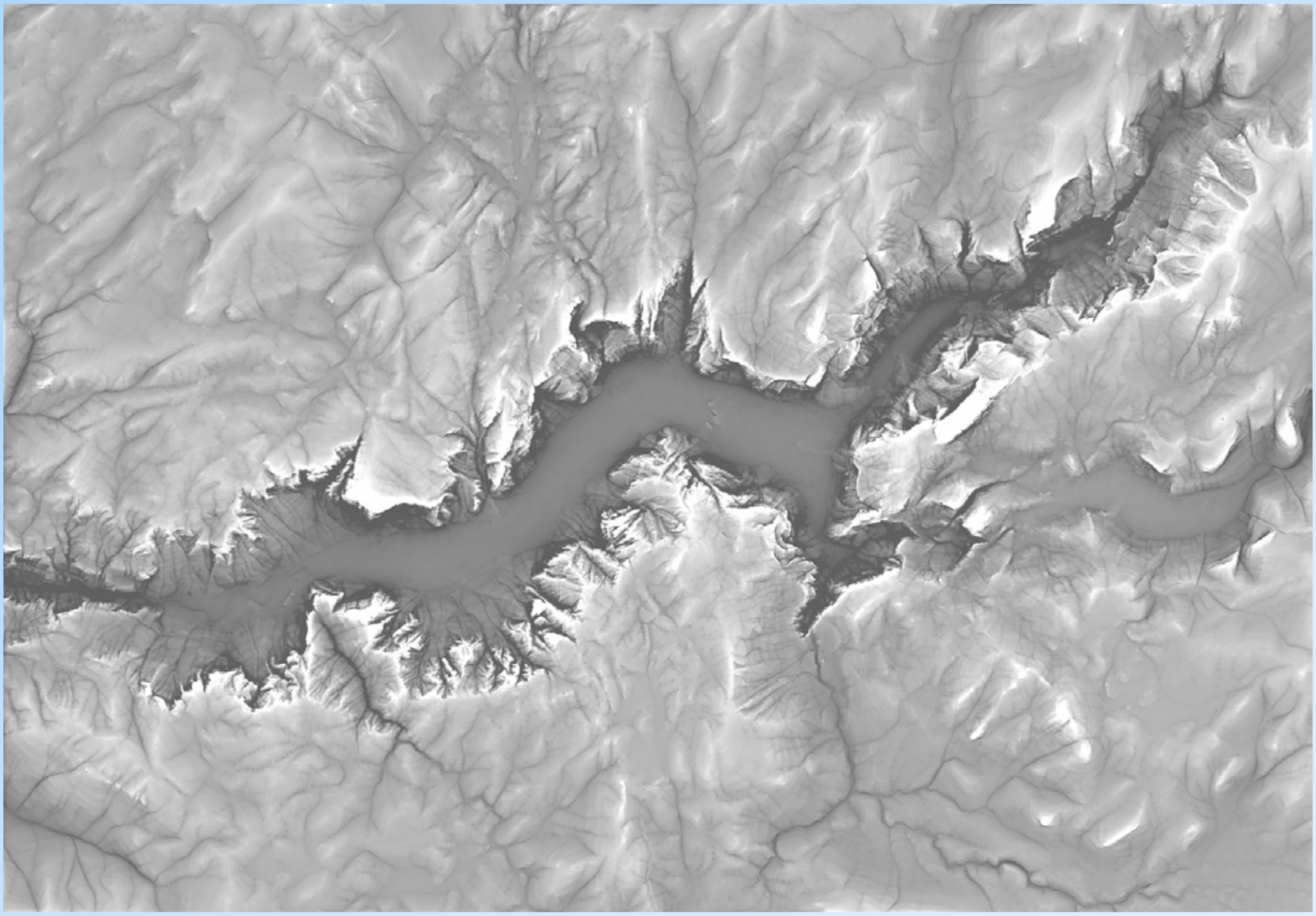
Yosemite Valley, CA (α = 2⁄3) Scale: 12.4 mi. x 8.6 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

19

Here is the same terrain rendered with alpha = 2/3.

It’s interesting to note here that Bridalveil Creek (running into the valley near the western end, from the south) is the darkest, most clearly delineated tributary to the valley in this image. This is likely a clue that it has the greatest water flow, and in fact Bridalveil Fall is the one waterfall (apart from the main canyons) that runs all year.

Yosemite Valley – Texture Shading (α = 1)

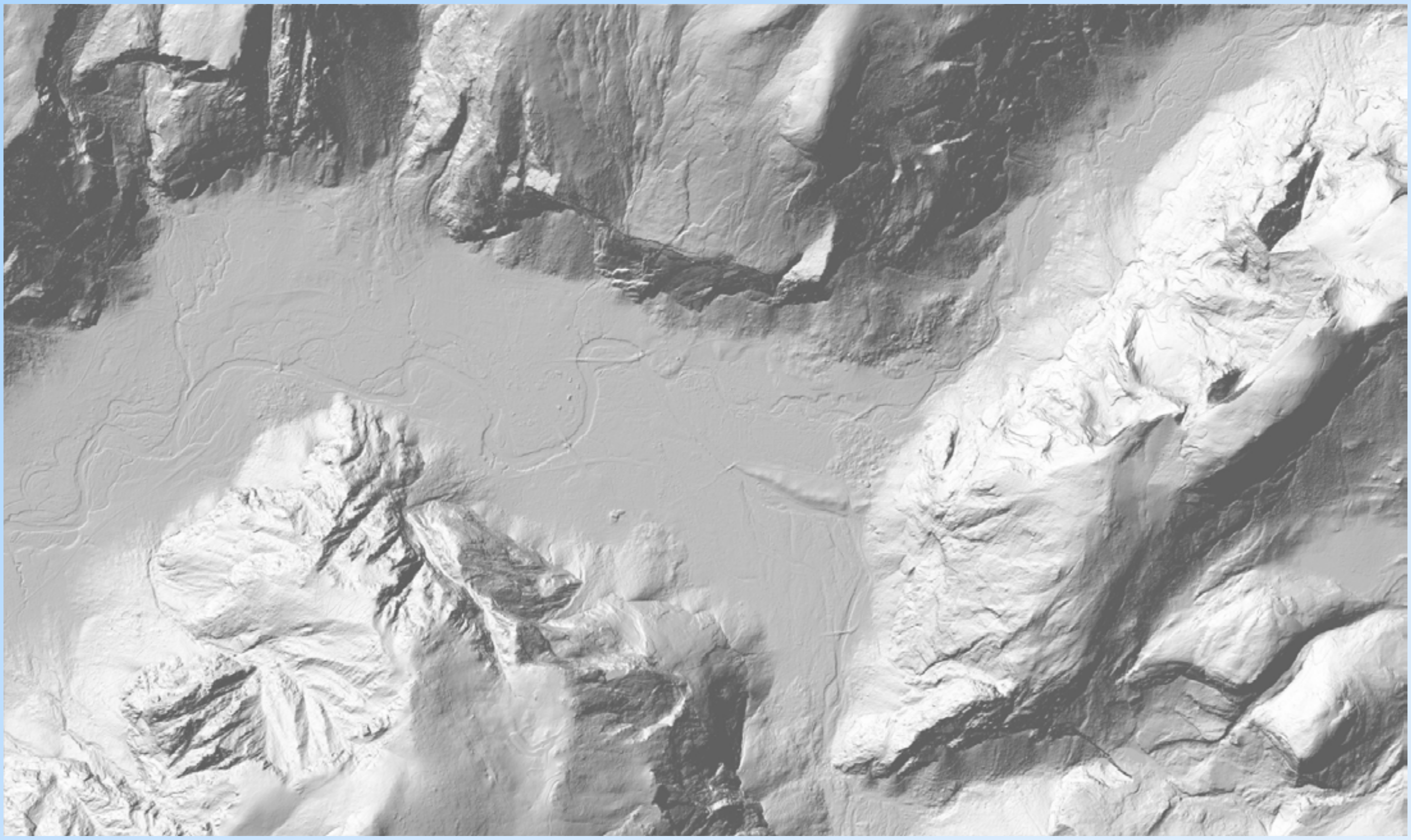


Yosemite Valley, CA (α = 1) Scale: 12.4 mi. x 8.6 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

20

Yosemite Valley is something of a special case, because the one major feature here - the valley itself - so dominates all the other terrain variation, in comparison to either the flat valley floor or the surrounding landscape. This allows us to move alpha all the way up to 1, without causing the details to overwhelm the image or the valley to lose its appearance of depth. The result gives us a lot more detail on the canyon walls, as well as some along the valley floor.

Yosemite Valley – Hillshading: Where is Half Dome?

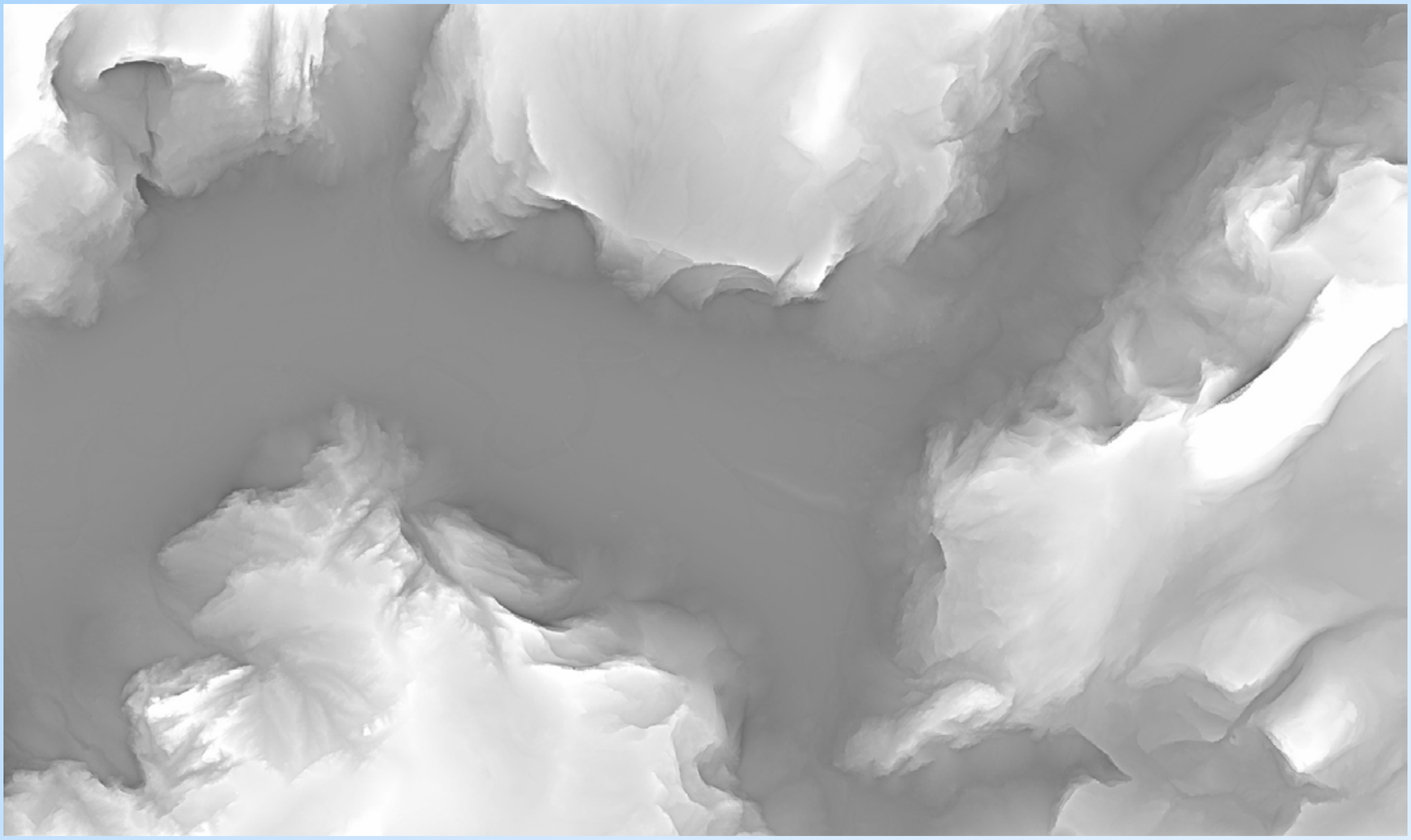


Eastern Yosemite Valley, CA Scale: 4.2 mi. x 2.5 mi.
Source of elevation data: National Park Service (filtered LIDAR DEM)

21

This is the eastern end of Yosemite Valley, derived from 1-meter LIDAR data provided by the National Park Service. Once again, it's very hard to see the face of Half Dome (though its shadowed back side can be seen).

Yosemite Valley – Texture Shading ($\alpha = 1/2$)



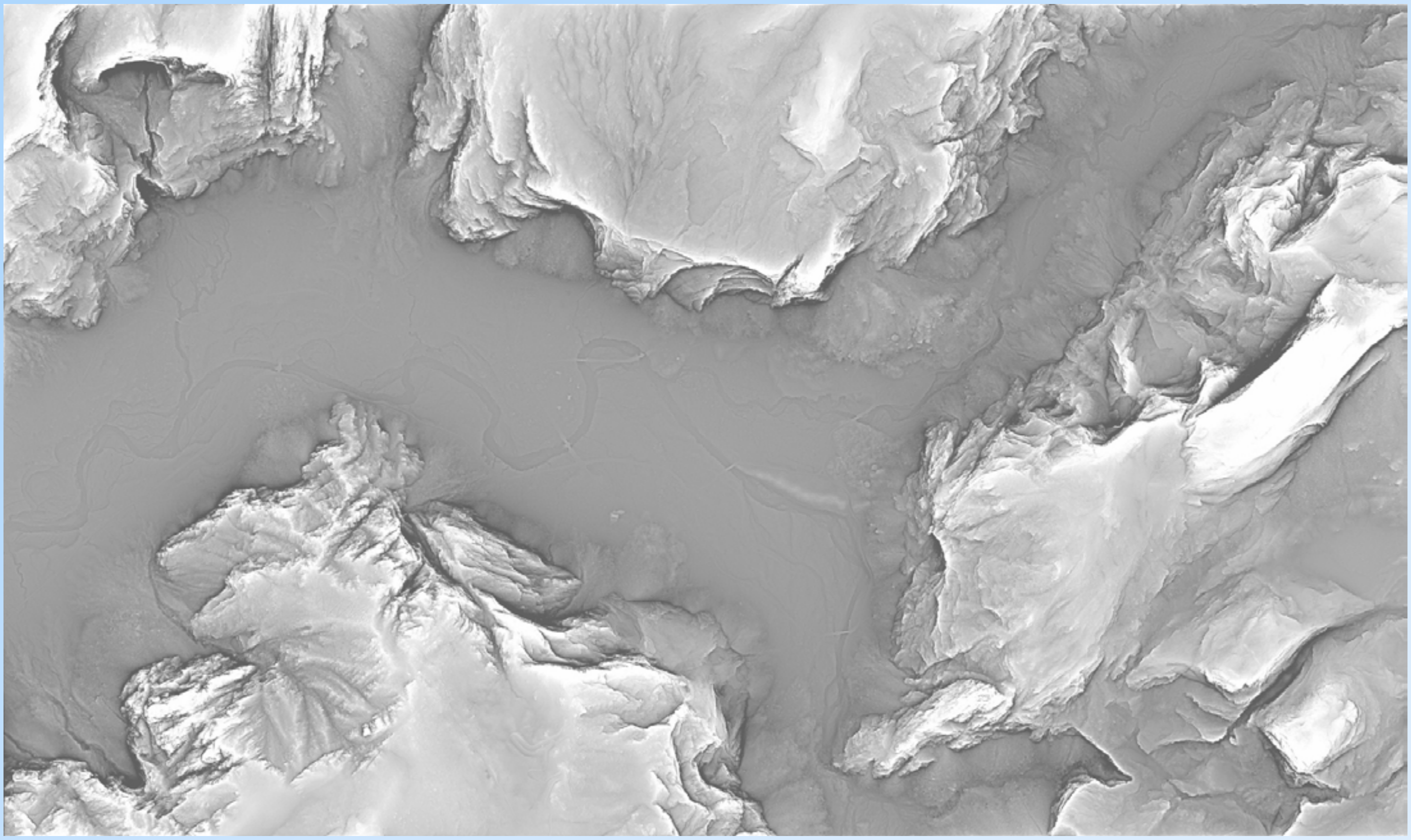
Eastern Yosemite Valley, CA ($\alpha = 1/2$) Scale: 4.2 mi. x 2.5 mi.
Source of elevation data: National Park Service (filtered LIDAR DEM)

22

With texture shading, Half Dome is clearly visible at the right edge of the map.

Here with alpha of 1/2, few other details are visible, because the valley is so dominant that most of the grayscale range is taken up showing its depth, softening the other features. A higher value of alpha will enhance the details further.

Yosemite Valley – Texture Shading (α = 1)



Eastern Yosemite Valley, CA (α = 1) Scale: 4.2 mi. x 2.5 mi.
Source of elevation data: National Park Service (filtered LIDAR DEM)

23

Here's the same map with an alpha of 1, showing not only the texture of the valley walls, but the banks of the meandering river on the valley floor, and the bridges across the river.

Remember, this is not an adaptive transform that enhances parts of the image differently based on areas of high or low relief. This is a linear operator - it applies exactly the same computation at each point - and yet it's able to display clearly both the drop on the sheer face of Half Dome and the contrast along the banks of the river on the same image - a dynamic range of 2 or 3 orders of magnitude. I found that to be remarkable.

Benefits of the New Method

- No inherent characteristic scale
  - May be beneficial for geomorphological interpretation (see later examples)
  - Lets the terrain reveal its own scale parameters, if any
  - Avoids imposing any *a priori* scale assumptions on the data
- A single large map can be precomputed and used at all resolutions
  - Useful for interactive applications
  - No need to compute different shading for each zoom level
  - Simply scale the gray levels according to the values currently in view
- Balance of major features and details
  - Helpful in creating a hierarchical mental framework
  - Major terrain features are most prominent, but details are also visible
  - This same balance is preserved when zooming in or out
- Combines this information into a single color dimension (brightness)
  - Leaves the rest of the color space to show vegetation, geology, etc.

24

One of the advantages of the scale independence of this shading method is that the algorithm contains no scale parameters - such as a blur radius (for relief generalization) or stream density (in channel networks). Thus, any scale information (such as stream density) which is apparent in the resulting map is actually present in the elevation data and not artificially imposed by choices made in the algorithm. Such features may provide information about geomorphological processes in the actual terrain (or they could also be artifacts present in the input data).

The Fractional Laplacian Operator $(-\Delta)^{\alpha/2}$

Frequency domain

Laplacian Δz of $z(x,y)$:

$$-\left(2\pi\|f\|\right)^2 \cdot \hat{z}$$

Fractional Laplacian ($0<\alpha<2$):

$$\left(2\pi\|f\|\right)^\alpha \cdot \hat{z}$$

Spatial domain

$$\Delta z = \frac{\partial^2 z}{\partial x^2} + \frac{\partial^2 z}{\partial y^2}$$

$(-\Delta)^{\alpha/2} z = \ldots$ [much more complex]

- The effect is to enhance high spatial frequencies (details) more than low frequencies (large features)
  - The relative emphasis is controlled by the power law exponent α
- Simple to apply in the frequency domain (after Fourier transform)
- The fractional Laplacian is more complex in the spatial domain …

25

Unlike the ordinary Laplacian, the fractional Laplacian is very difficult to compute directly on the original data in the spatial domain. A Fourier transform is used to convert the data into a set of components of different spatial frequencies; then the function is applied on this data in the frequency domain, and the result is run through another Fourier transform to yield the result. This gives a relatively efficient implementation - the Fourier transforms account for almost all the computation time; the rest is very fast.

Fractional Laplacian Spatial Domain Characteristics

- A nonlocal operator
  - Unlike the gradient operator used for hillshading
  - The result at each point depends on *all* the data, to varying degrees
  - More computation-intensive
- A measure of *relative height* compared to surrounding data
  - Closer data is weighted more heavily
  - Highlights ridges and valleys
  - The contribution at distance r is weighted according to a power law, $r^{-(1+\alpha)}$

26

Compared to computing the slope and illumination at each point, however, texture shading is somewhat computation-intensive.

To be more precise, right now it’s taking about 3 seconds per million grid cells to produce a texture-shaded map on my 3 GHz Mac at home (as long as there’s enough memory). So processing a 10,000 x 10,000 grid takes about 5 minutes. There are a couple of specific things I’m doing inefficiently, so I’m pretty sure those times can be sped up by at least a factor of 4 without much difficulty.

Fractional Laplacian Asymptotic Behavior – Issues

- At very small length scales, features are greatly amplified
  - Proper discretization of this operator is tricky
  - Any artificial discontinuities will produce artifacts
  - Solution: model a continuous surface using a smooth interpolator on the gridded data (see next slide)
- At increasing length scale, weighting gets smaller but height features tend to get larger
  - Data at large distances has a decreasing but nontrivial influence
  - Makes it difficult to merge adjacent maps seamlessly

27

If we examine the fractional Laplacian at the extremes of scale, we run into two problems. First, we need an interpolation scheme to define a continuous, smooth surface from our individual grid points, since the fractional Laplacian will not tolerate sharp discontinuities such as a nearest-neighbor or bilinear interpolation model would produce. This problem has a solution which we'll see in the following slides.

The second issue is related to generating large seamless maps from individual tiles, and is more problematic.

Smooth Bandlimited Interpolators

- A little-known fact: an interpolating cubic spline on a regular grid can be expressed as a convolution [2]
  - Allows it to be applied easily in the frequency domain
- Two close approximations to the cubic spline frequency spectrum form smooth interpolators bandlimited at *twice* the Nyquist frequency:

$$\hat{s}(f) = \boxed{\left(\cos^4 \frac{\pi f}{2}\right) \cdot \frac{2+\cos \pi f}{2+\cos 2\pi f}} \quad \text{or} \quad \hat{s}(f) = \boxed{\frac{\left(1-|f|\right)^4}{\left(1-|f|\right)^4 + f^4}} \quad \left(-1 \le f \le 1\right)$$

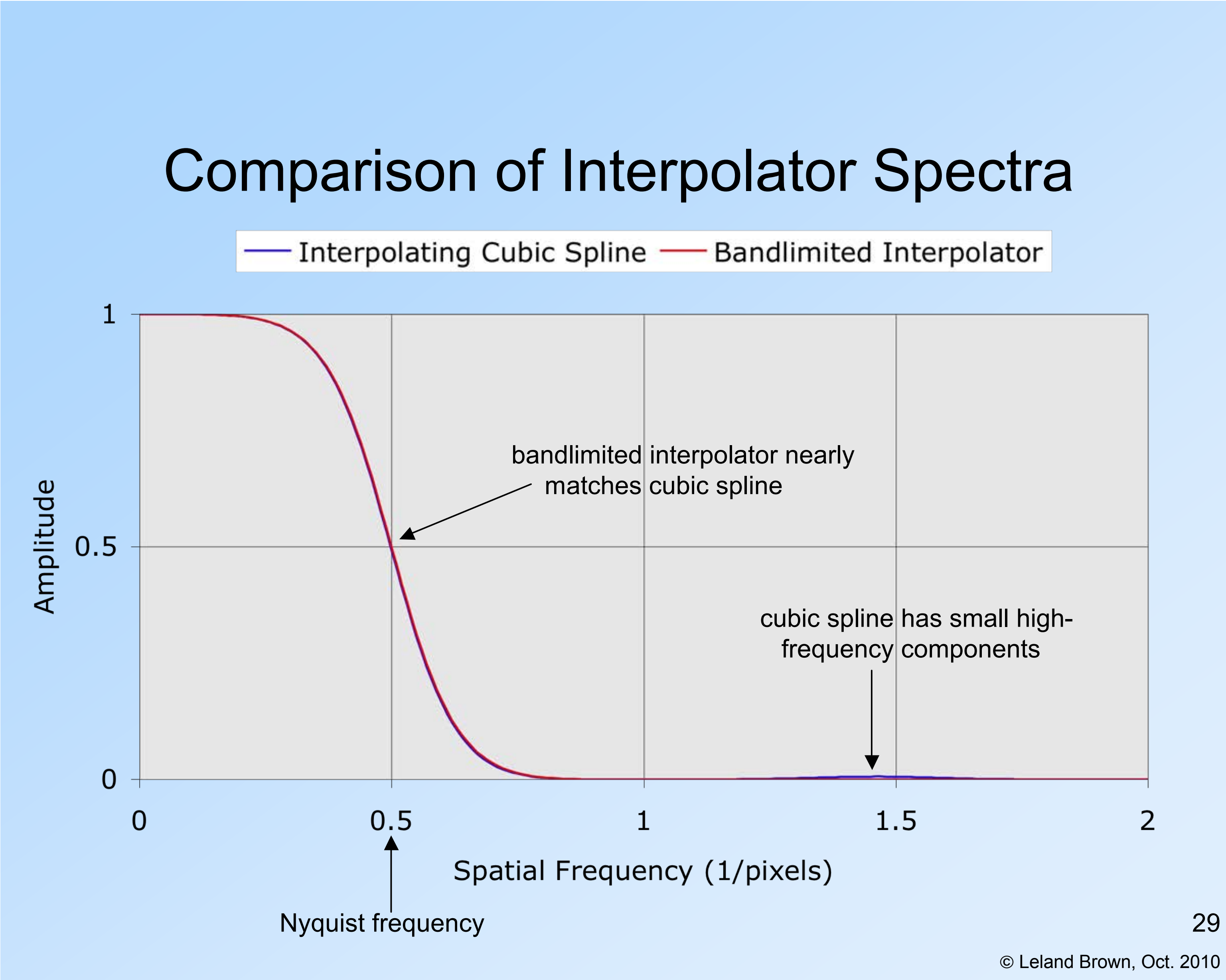
- Frequency limit ensures computation is finite
- Minimal edge effects – spatial-domain kernels decay rapidly ($\sim|x|^{-5}$)
- Both are nearly identical to the cubic spline (see next slide)

[2] N. A. Dodgson, "Image resampling," Tech. Rep. 261,
Univ. Cambridge Comp. Lab., Cambridge, U.K., 1992, 127-130.

28

A high-quality smooth interpolator appropriate for terrain processing is the cubic spline. If we examine its properties in the frequency domain, we find that (by virtue of being smooth) it eliminates almost all high-frequency components beyond the pixel density of the input grid, as we'll see on the next slide.

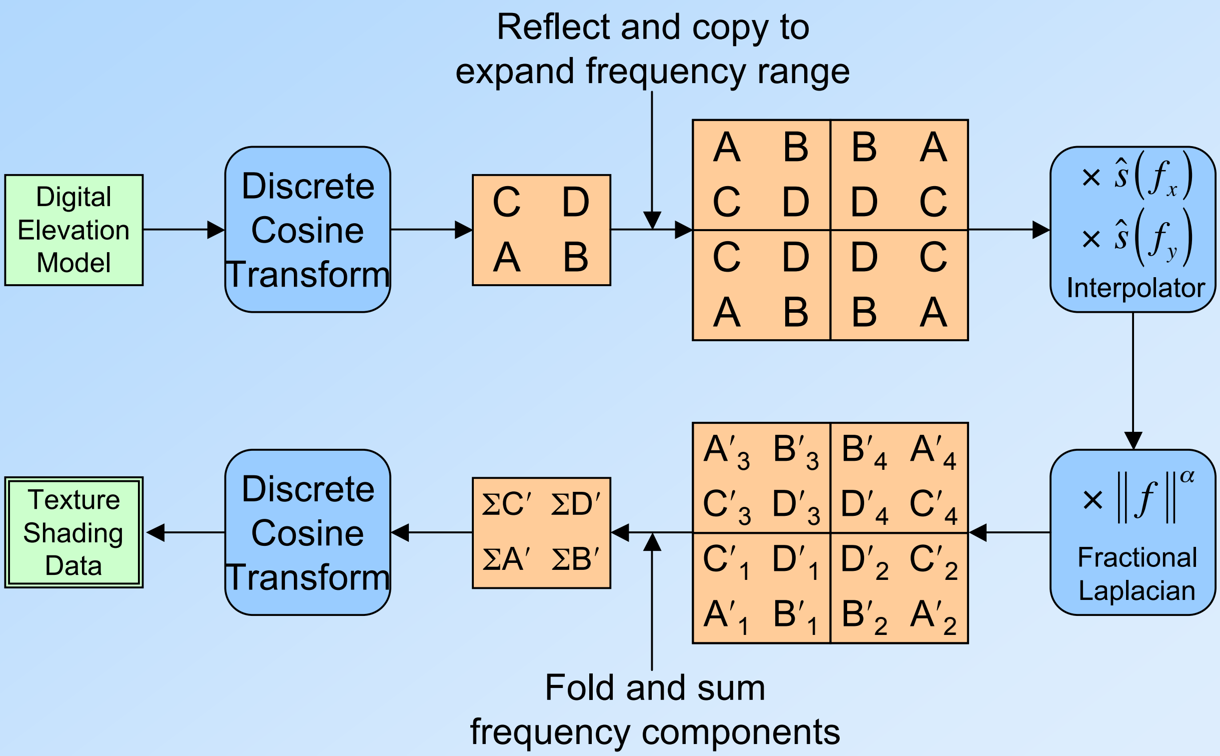
We would prefer, however, to have an interpolator that contains exactly zero of those high frequencies, while retaining the useful properties of the cubic spline. That will allow us to ignore those components in the computations entirely and still get a precise result. It turns out that either of the two functions shown here will achieve that without introducing unwanted artifacts.



This graph shows a comparison of the cubic spline frequency spectrum to one of the functions on the previous slide. If you can't see the difference between the red and blue curves on the graph, that's the point - the two functions are very nearly identical. The only visible difference is the small high-frequency bump of the cubic spline.

We might be tempted to just cut off the spline curve and zero out the high frequencies. But that would result in an "interpolated" surface that doesn't quite pass through the data points, as well as some minor edge effects due to a more abrupt transition at the frequency cutoff. The two alternative functions don't suffer either of these drawbacks, and they are just as easy to compute.

Texture Shading Algorithm



30

Here's a diagram summarizing the texture shading algorithm. The input DEM is passed through an off-the-shelf DCT function, a type of Fourier transform. This produces a new grid containing spatial frequency information for a certain range of frequencies.

This output is expanded by making 4 copies, to cover the frequency range of interest to the interpolator. We then multiply this by one of the interpolator functions, in both x and y directions. These two steps are analogous to performing an interpolation on the original spatial data.

Next we multiply by the fractional Laplacian function, which in the frequency domain is just a simply radial power law. The result is folded over and added back together; this is the analog of re-gridding the data. (In fact at this point we can even re-grid at a different grid pitch, so as to change the resolution either up or down, or adjust the pixel aspect ratio, right here within the texture shading algorithm - eliminating the need for a separate interpolation step.)

Finally, we run it back through a DCT, yielding the texture-shaded map. If all of this sounds complicated, it's actually very easy to implement.

Note on Map Projections

- Weighting height information by a power of distance results in a comingling of vertical and horizontal units
  - Output of algorithm is in units of height × $(\text{pixels})^{-\alpha}$
  - Since pixels measure horizontal distance, this has dimensions of $(\text{length})^{1-\alpha}$
- Projections with significant shape distortion should be avoided
  - Linear units per pixel should be (nearly) the same in X and Y directions
  - Texture shade using a more conformal projection and reproject the result
- Projections with area distortion should be weighted by local scale
  - Multiply output by $A^{\alpha/2}$, where A is (proportional to) the local area scale
  - For Mercator projection, multiply by $(\text{secant of parametric latitude})^{\alpha}$
  - Without this correction, areas of larger scale will appear comparatively flat
  - Not necessary for large-scale maps or equal-area projections

31

This is a technical note of interest to those who want to implement the algorithm, explaining how to get suitable results with certain map projections.

[This slide has been corrected from the original presentation, to specify *parametric* latitude in the Mercator scale formula.]

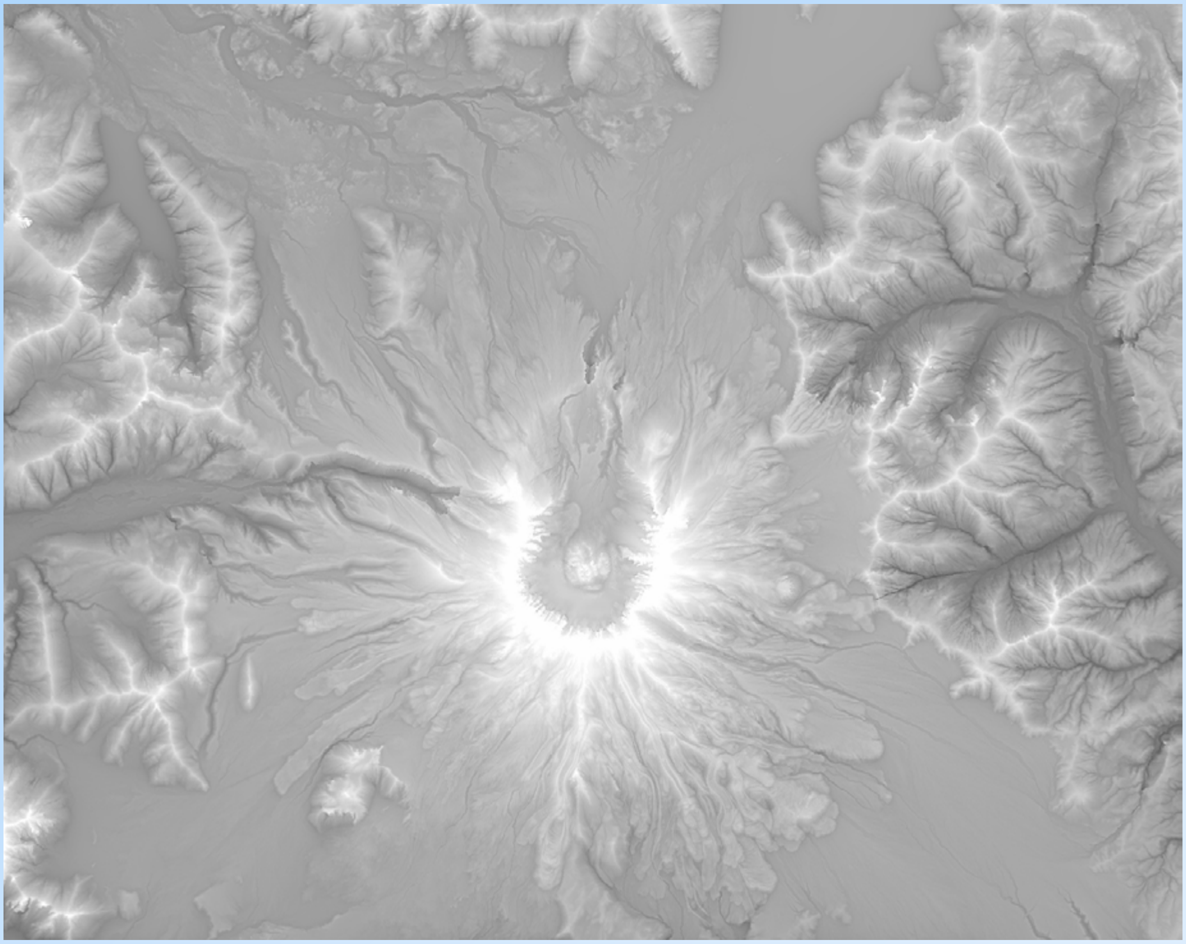
Some Remaining Issues

- Converting output to grayscale
  - Histogram of output values tends to have long tails
  - Min/max values as limits is too wide a range
    - Makes most of the map medium gray
  - Mean ± 2 or 3 standard deviations is often too narrow a range
    - High ridges and peaks can saturate at white and look flattened
  - Possible solutions
    - Use higher-order statistics, and/or use sigmoid function instead of linear mapping
- Generating seamless maps
  - This algorithm ignores the contribution of terrain outside the tile
  - Visual effect is small but will cause vertical offset at tile boundaries
  - Possible solution
    - Incorporate data from a wide area at a much lower resolution
    - This should be sufficient to address the problem

32

These are two practical issues with the technique that are as-yet unresolved - though I have ideas for each that I'm planning to try.

Texture Shading Examples: Mount Saint Helens, WA

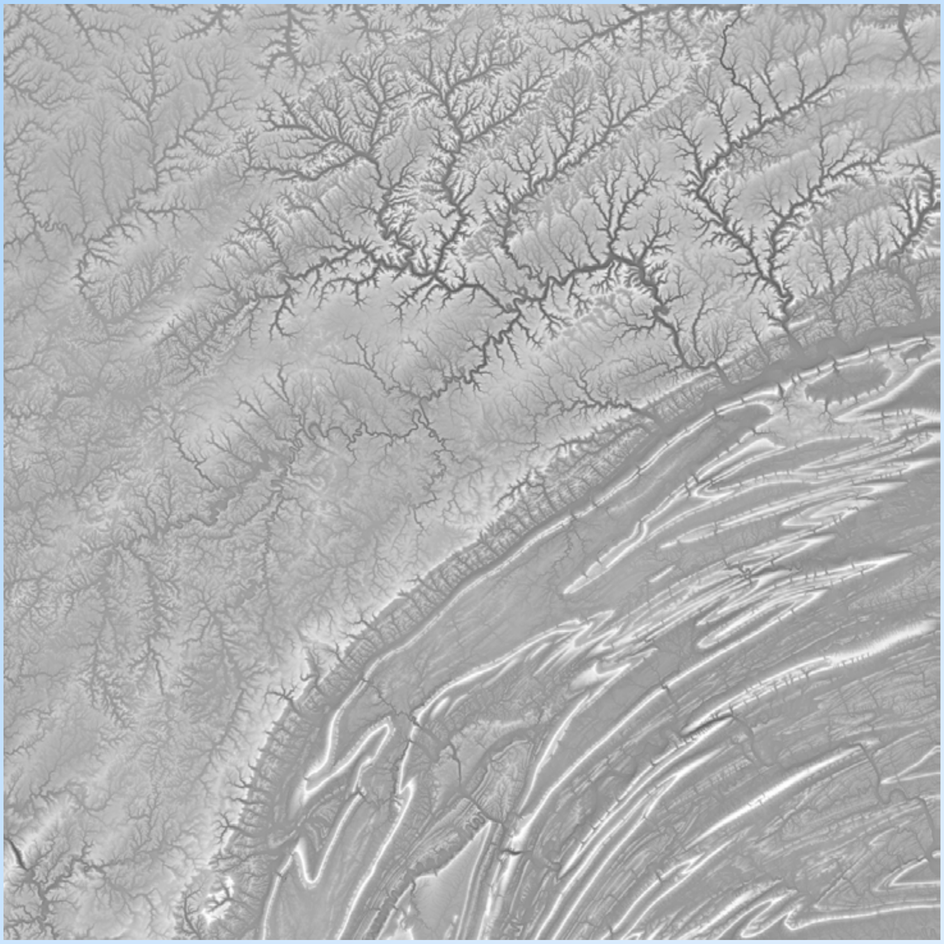


Mount Saint Helens, WA ($\alpha = 2/3$) Scale: 10.9 mi. x 8.6 mi.
Source of elevation data: U.S. Geological Survey (NED 1/9 Arc-Second)

33

In this map of Mt. St. Helens, the crater rim demonstrates the first issue from the preceding slide, having saturated at white. So there's actually more detail available of the rim, which could be brought out by choosing different limits for the grayscale mapping.

Appalachian Plateau, PA

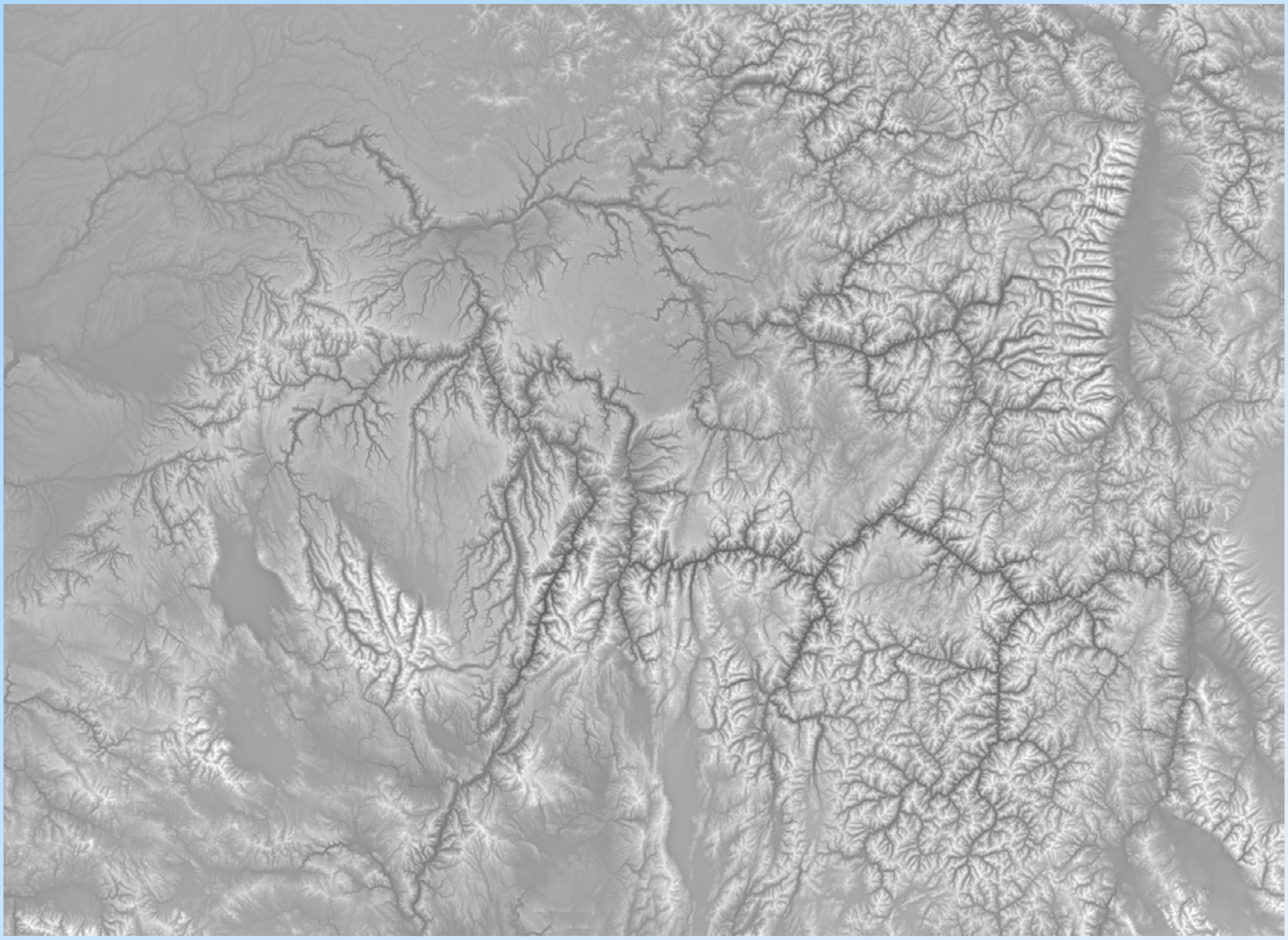


Appalachian Plateau, PA ($\alpha = 1/2$) Scale: 100 mi. x 100 mi.
Source of elevation data: U.S. Geological Survey (NED 1 Arc-Second)

34

This map shows the Appalachian Plateau and the Valley and Ridge region in Pennsylvania. Not knowing much about geology, I was fascinated to see the contrast here.

Salmon River, ID

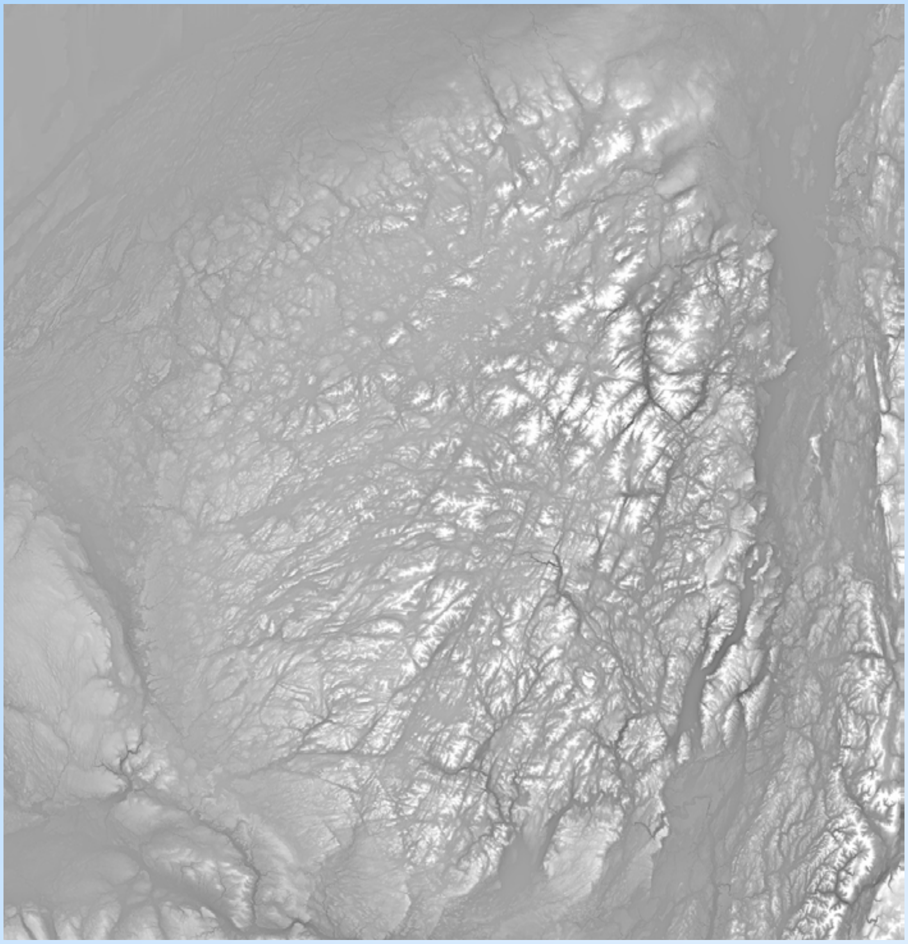


Salmon River, ID ($\alpha = 1/2$) Scale: 270 mi. x 190 mi.
Source of elevation data: U.S. Geological Survey (NED 1 Arc-Second)

35

The Salmon River, Hells Canyon on the Snake River, and the Clearwater River are shown in this map. Notice here that the map spans 270 miles, versus only about 4 miles for the eastern Yosemite Valley map. This shows how the texture shading technique works across a wide range of scales.

Adirondack Mountains, NY

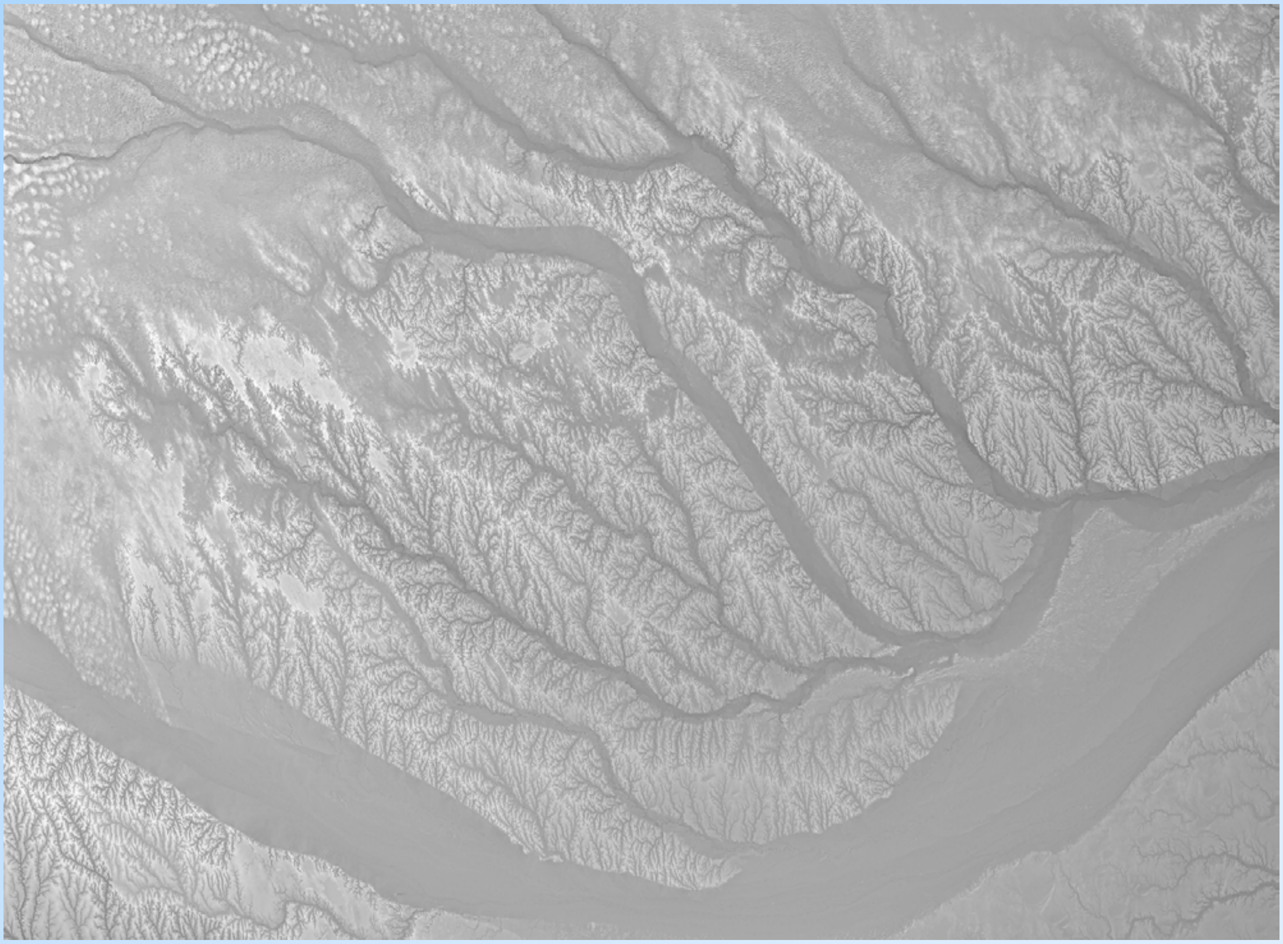


Adirondack Mountains, NY ($\alpha = 1/2$) Scale: 130 mi. x 140 mi.
Source of elevation data: U.S. Geological Survey (NED 1 Arc-Second)

36

With the Adirondacks we see a very different type of topography than the rugged mountains of California.

Central Nebraska

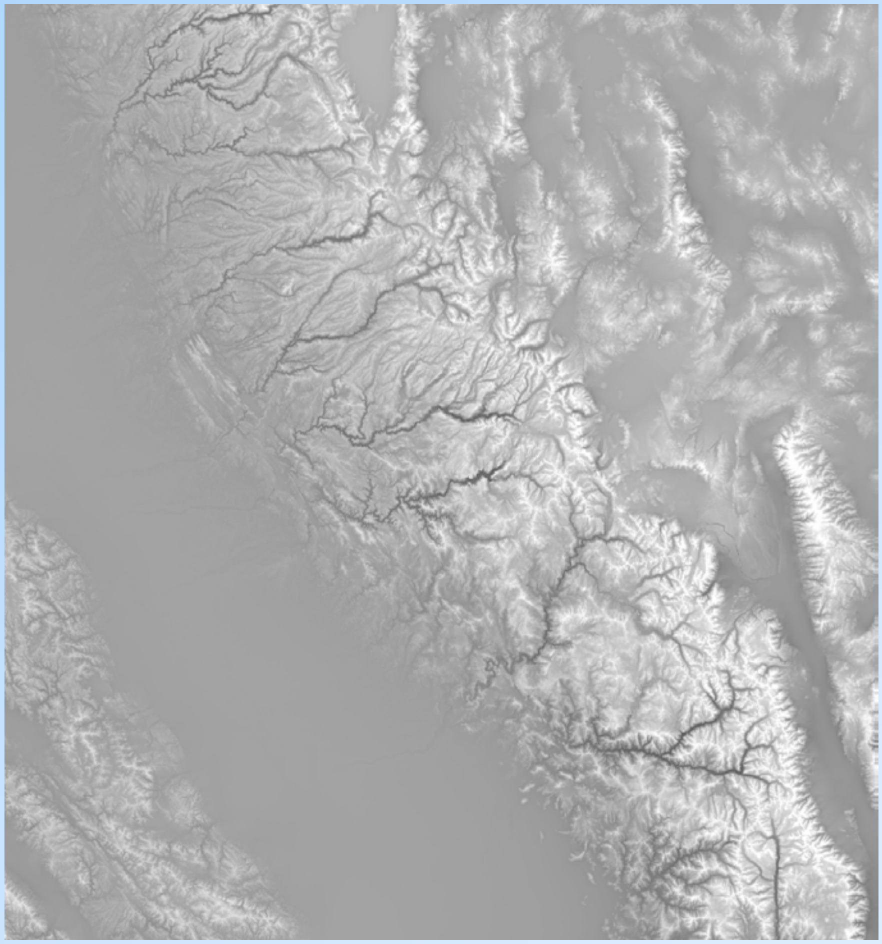


Central Nebraska ($\alpha = 1/2$) Scale: 130 mi. x 95 mi.
Source of elevation data: U.S. Geological Survey (NED 1 Arc-Second)

37

This map of central Nebraska shows that texture shading can also be used on areas of much lower relief.

Sierra Nevada Mountains, CA

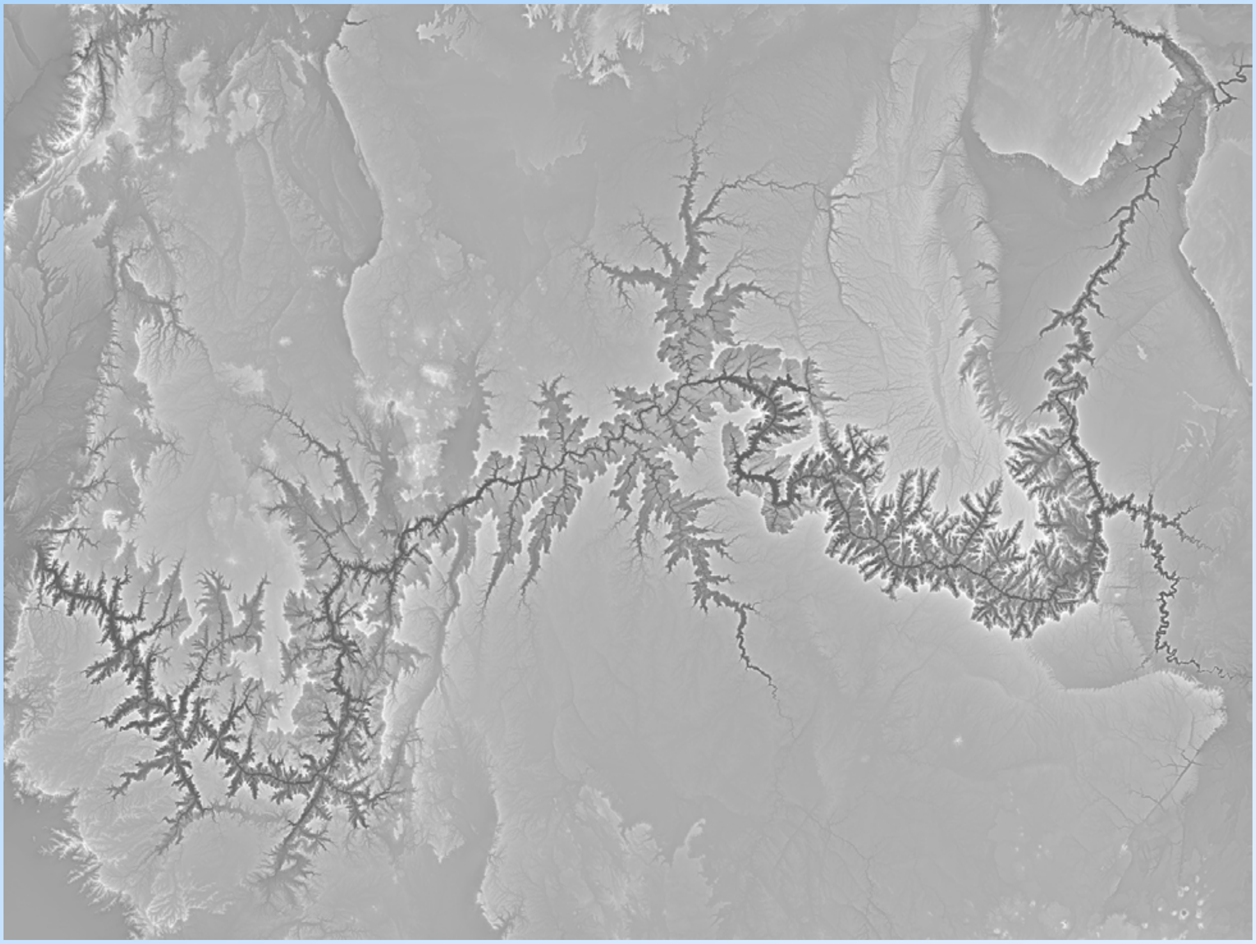


Sierra Nevada Mountains, CA ($\alpha = 1/2$)   Scale: 190 mi. x 210 mi.
Source of elevation data: U.S. Geological Survey (NED 1 Arc-Second)

38

The Sierras look like a speed bump across the state of California. To the left is the Central Valley.

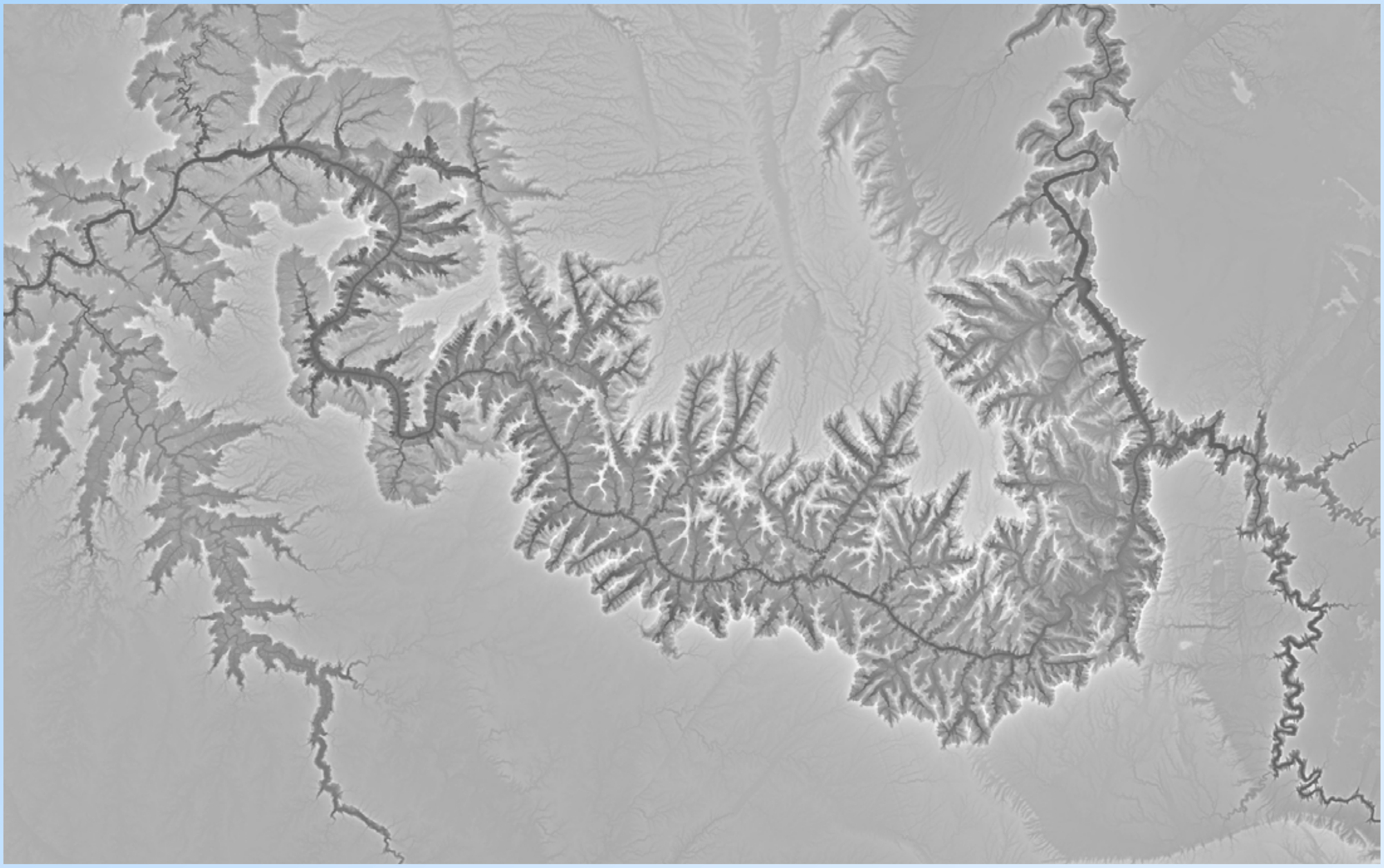
Grand Canyon, AZ



Grand Canyon, AZ ($\alpha = 2/3$) Scale: 140 mi. x 100 mi.
Source of elevation data: U.S. Geological Survey (NED 1 Arc-Second)

39

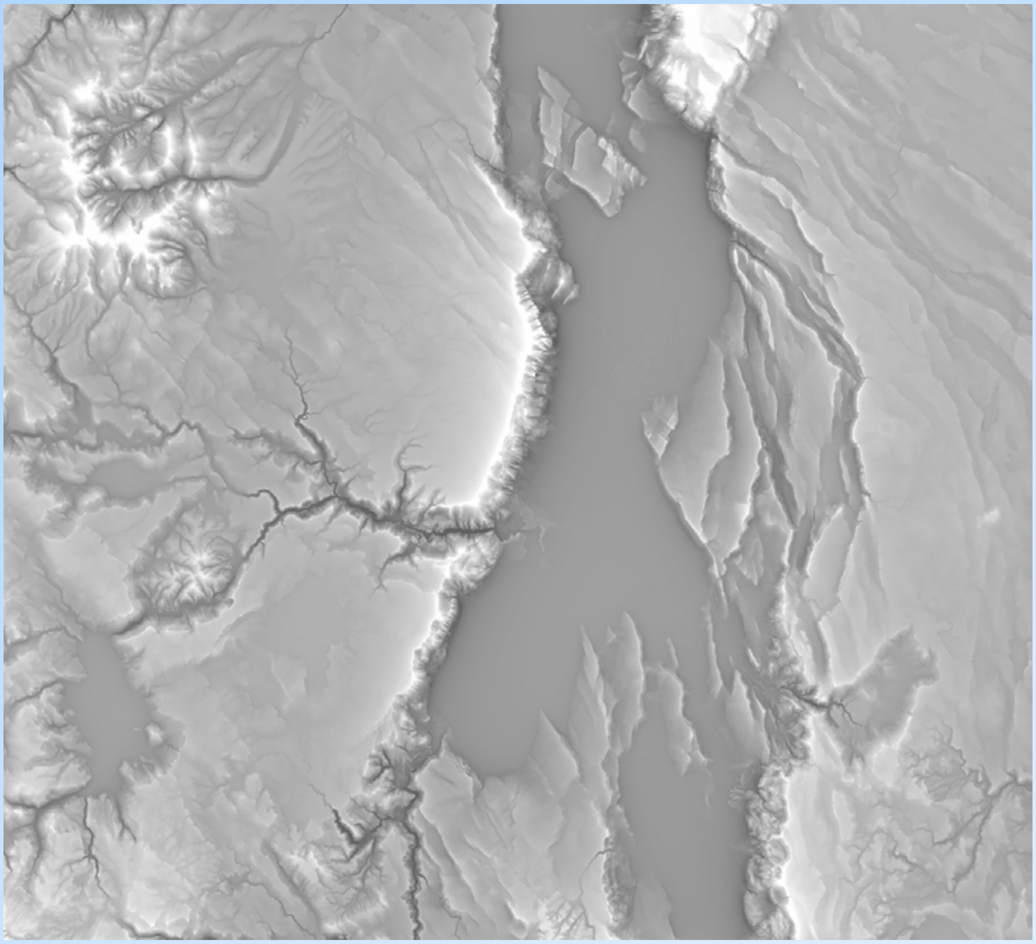
Grand Canyon, AZ



Grand Canyon, AZ ($\alpha = 2/3$) Scale: 66 mi. x 41 mi.
Source of elevation data: U.S. Geological Survey (NED 1 Arc-Second)

40

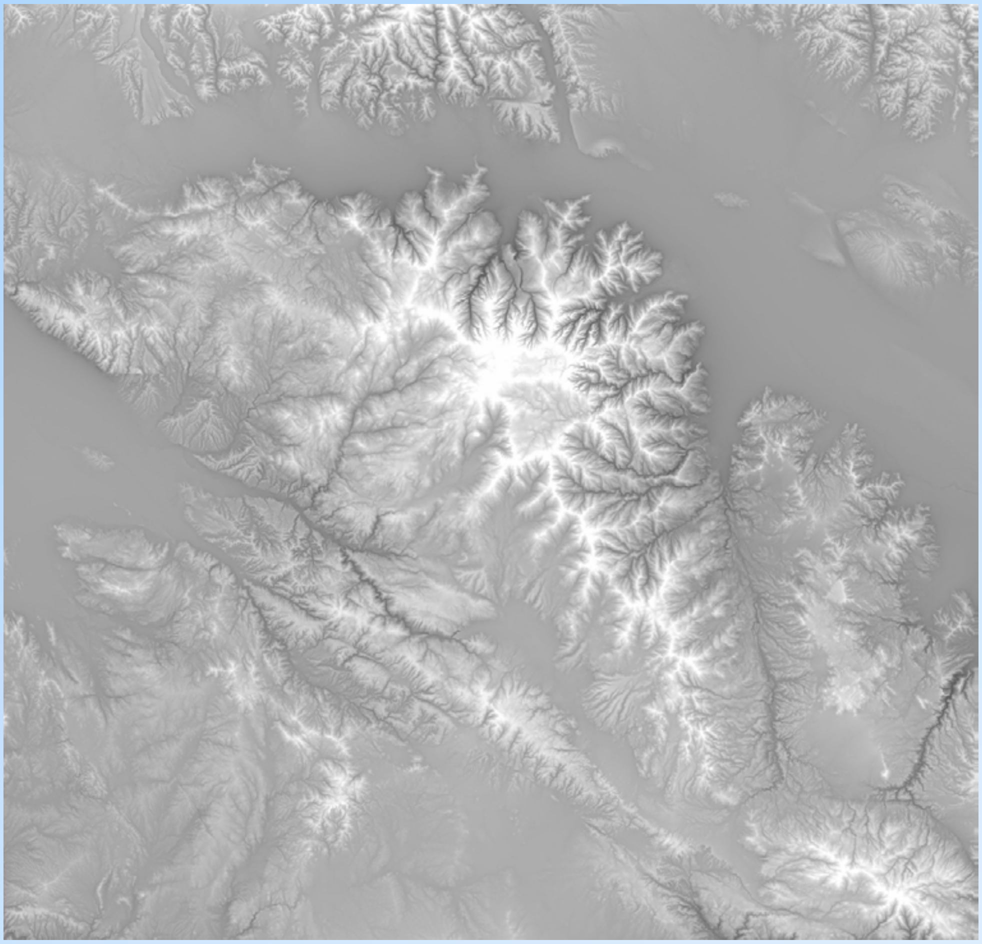
High Lava Plains, OR



High Lava Plains, OR ($\alpha = 2/3$) Scale: 30 mi. x 28 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

41

San Jacinto Mountains, CA

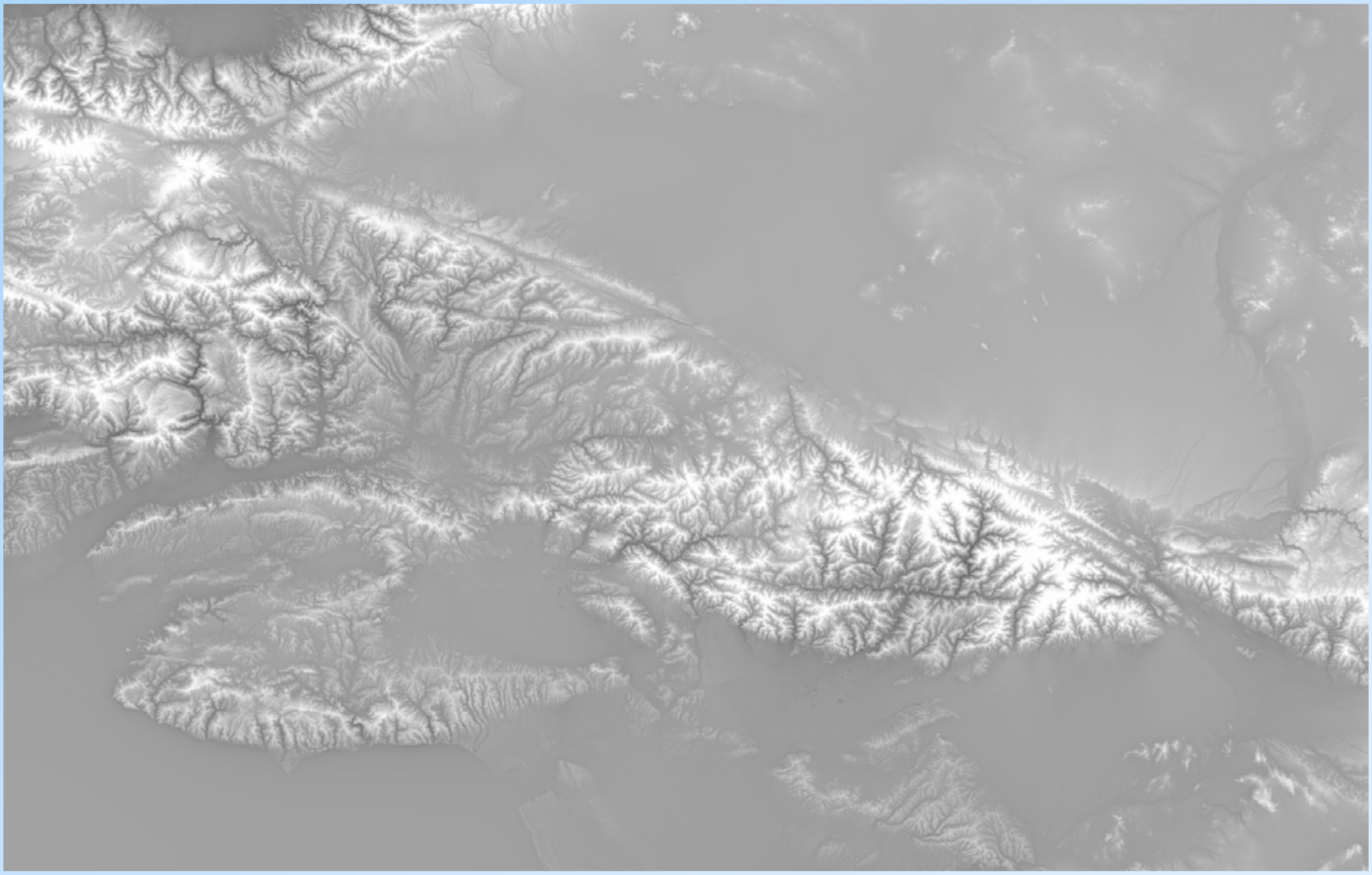


San Jacinto Mountains, CA ($\alpha = 2/3$) Scale: 36 mi. x 35 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

42

Palm Springs is in the valley to the right of the mountain.

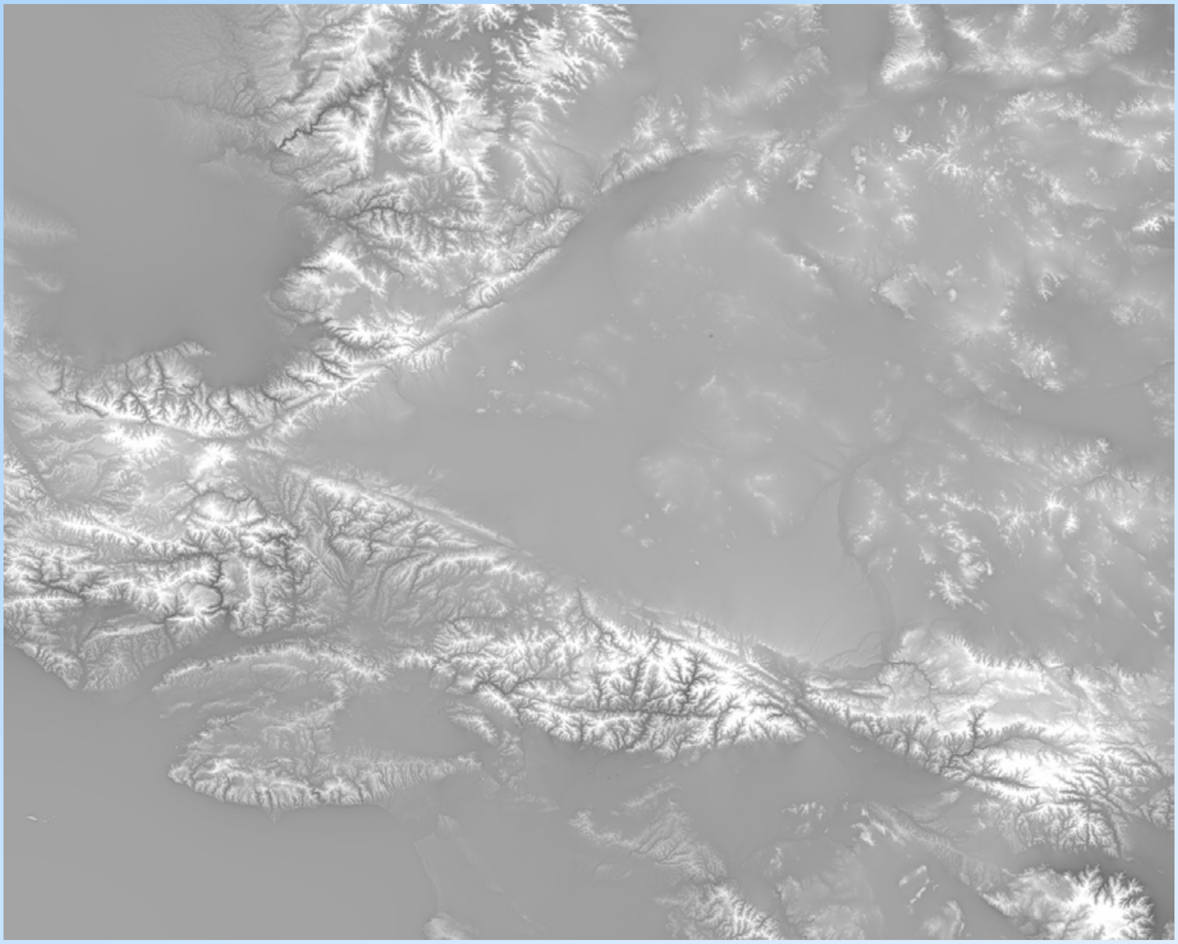
San Andreas Fault, CA



San Andreas Fault, CA ($\alpha = 1/2$)　　Scale: 120 mi. x 80 mi.
Source of elevation data: U.S. Geological Survey (NED 1 Arc-Second)

43

Antelope Valley, CA

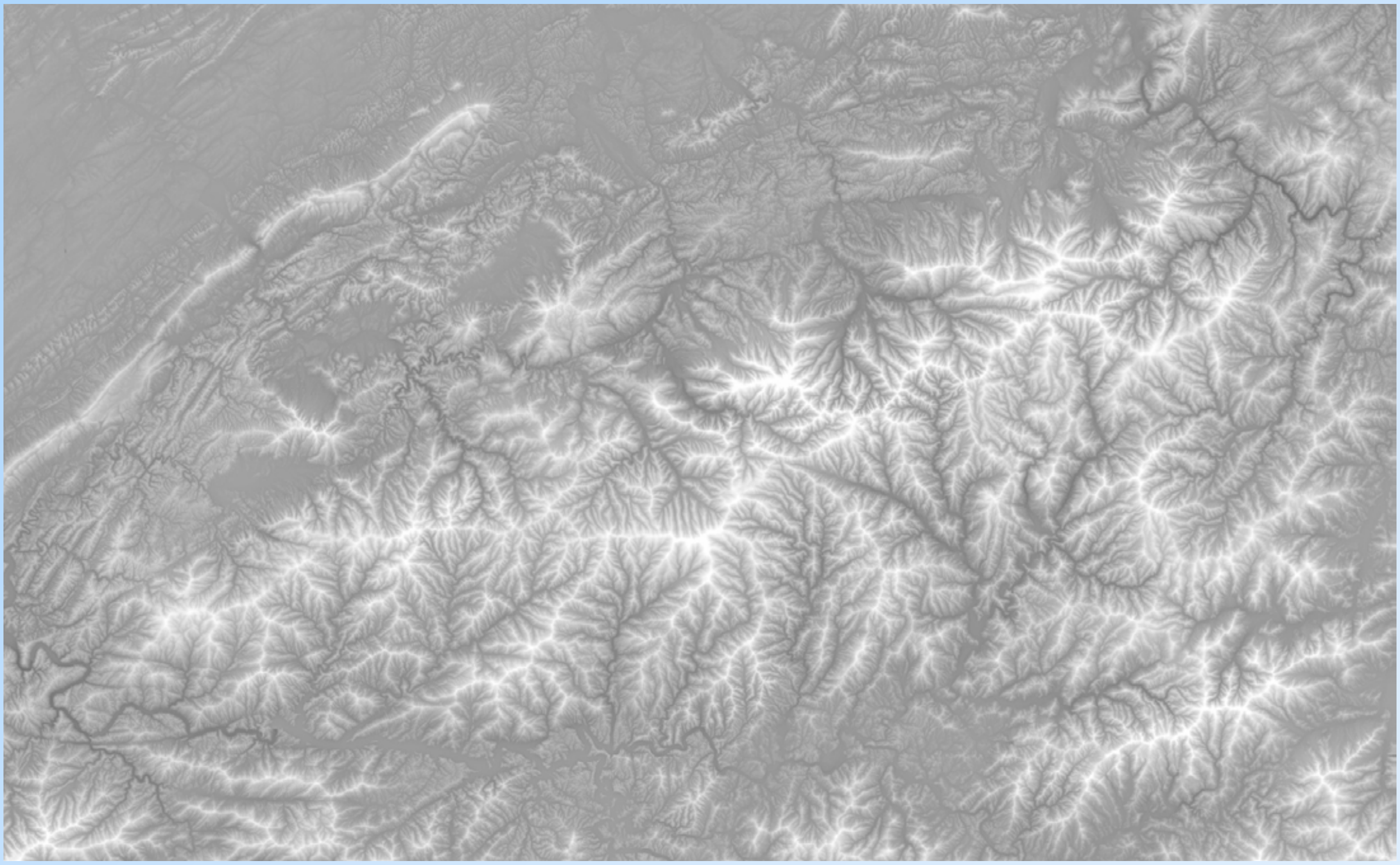


Antelope Valley, CA ($\alpha = 1/2$) Scale: 170 mi. x 140 mi.
Source of elevation data: U.S. Geological Survey (NED 1 Arc-Second)

44

The San Andreas Fault and the Garlock Fault, forming two sides of a triangle.

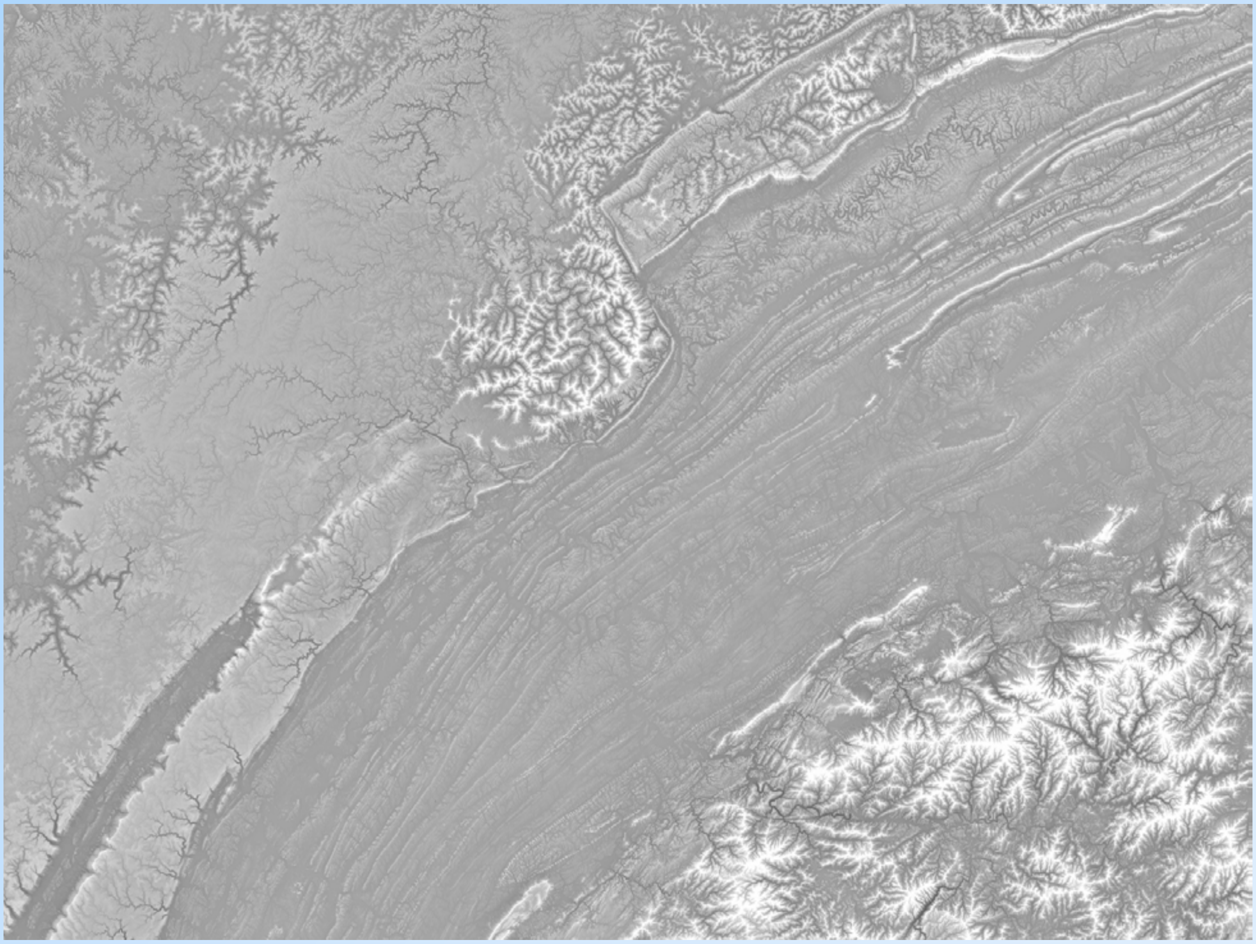
Great Smoky Mountains, TN & NC



Great Smoky Mountains, TN & NC ($\alpha = 1/2$) Scale: 56 mi. x 35 mi.
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

45

Eastern Tennessee

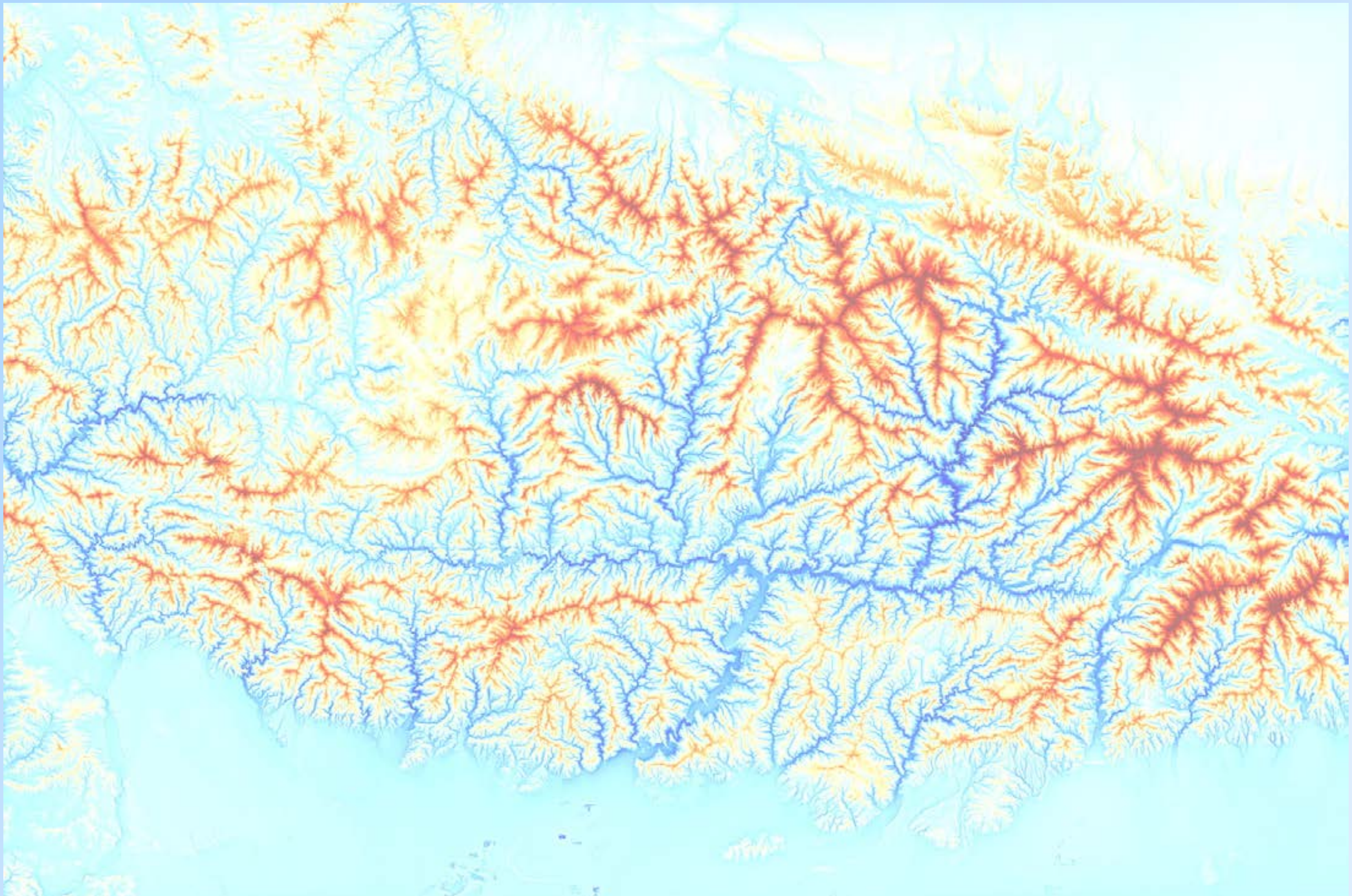


Eastern Tennessee ($\alpha = 1/2$) Scale: 140 mi. x 100 mi.
Source of elevation data: U.S. Geological Survey (NED 1 Arc-Second)

46

Part of the Tennessee Valley in the center, with the Smoky Mountains in the lower right.

Changing the Color Map – Ridges Red, Canyons Blue



San Gabriel Mountains, CA ($\alpha = 2/3$)
Source of elevation data: U.S. Geological Survey (NED 1/3 Arc-Second)

47

As a final illustration of what can be done, this is the same texture-shaded image of the San Gabriel Mountains we saw before, only using a different color map to show canyons in blue and ridges in red.

Questions?

Comments?

Feedback?

48