Computer Graphics

Visualization II

Emanuele Rodolà
rodola@di.uniroma1.it

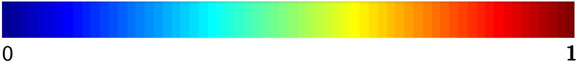
SAPIENZA
UNIVERSITÀ DI ROMA

2nd semester a.y. 2018/2019 · March 28, 2019

Colormap limits

Sometimes it is useful to modify the colormap to get saturation effects

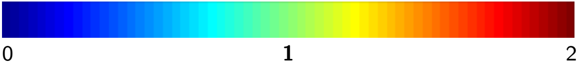
We can increase the maximum value represented in the colormap:



Colormap limits

Sometimes it is useful to modify the colormap to get saturation effects

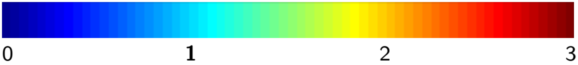
We can increase the maximum value represented in the colormap:



Colormap limits

Sometimes it is useful to modify the colormap to get saturation effects

We can increase the maximum value represented in the colormap:

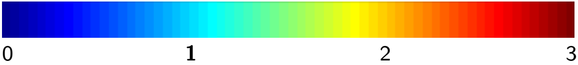


The colors remain fixed, but the mapping from values to colors changes

Colormap limits

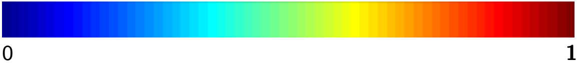
Sometimes it is useful to modify the colormap to get saturation effects

We can increase the maximum value represented in the colormap:



The colors remain fixed, but the mapping from values to colors changes

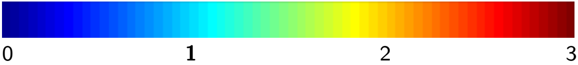
Similarly, we can decrease the max limit:



Colormap limits

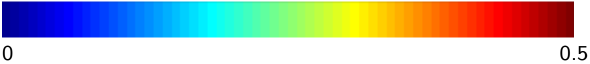
Sometimes it is useful to modify the colormap to get saturation effects

We can increase the maximum value represented in the colormap:



The colors remain fixed, but the mapping from values to colors changes

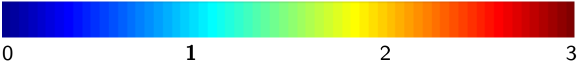
Similarly, we can decrease the max limit:



Colormap limits

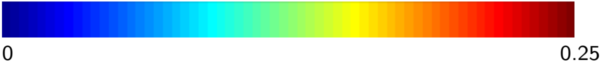
Sometimes it is useful to modify the colormap to get saturation effects

We can increase the maximum value represented in the colormap:



The colors remain fixed, but the mapping from values to colors changes

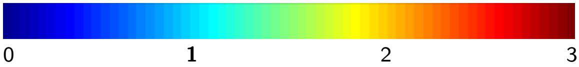
Similarly, we can decrease the max limit:



Colormap limits

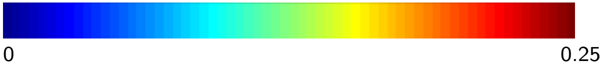
Sometimes it is useful to modify the colormap to get saturation effects

We can increase the maximum value represented in the colormap:



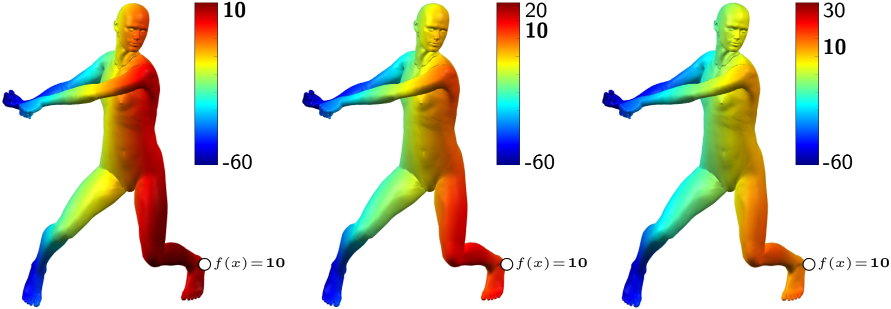
The colors remain fixed, but the mapping from values to colors changes

Similarly, we can decrease the max limit:

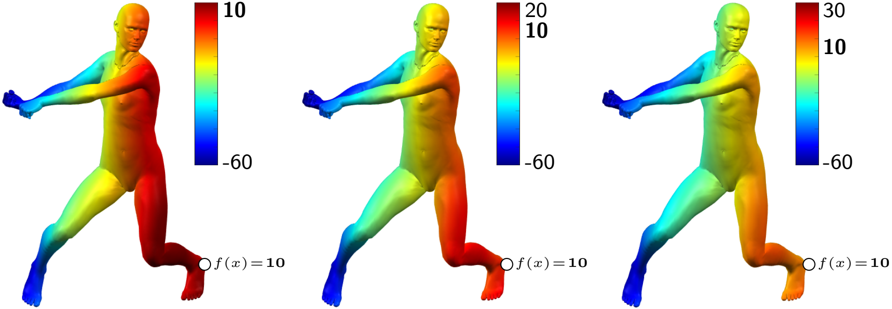


- Values $\geq$ the max limit are mapped to the max
- Values $\leq$ the min limit are mapped to the min
- Values between min and max are linearly interpolated

Increasing the max limit

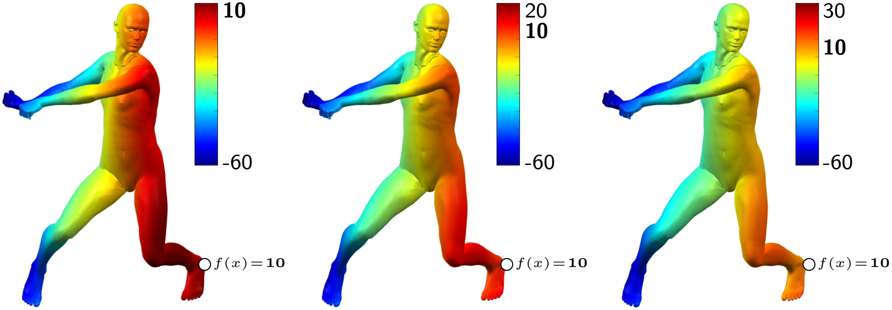


Increasing the max limit



For smooth colormaps, the effect is to “flatten out” the colors and make variations less evident

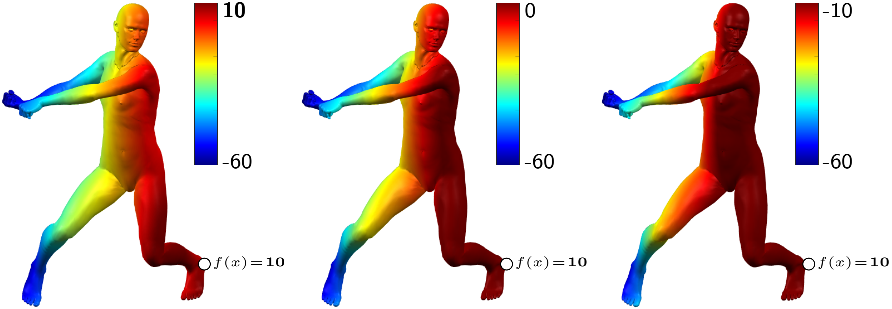
Increasing the max limit



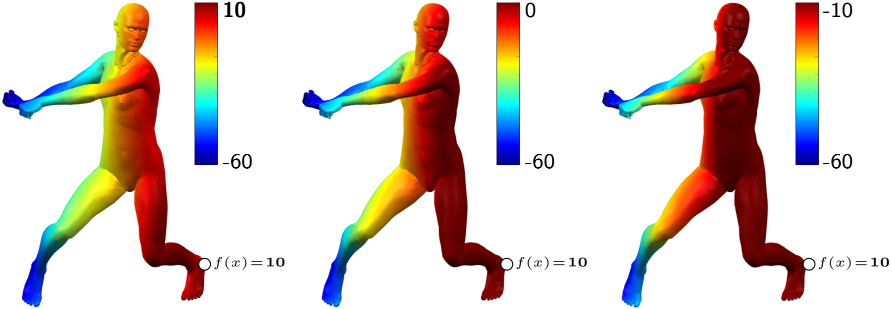
For smooth colormaps, the effect is to “flatten out” the colors and make variations less evident

In Matlab: caxis ([min max]) sets the colormap limits and does all the saturation and interpolation for us

Decreasing the max limit



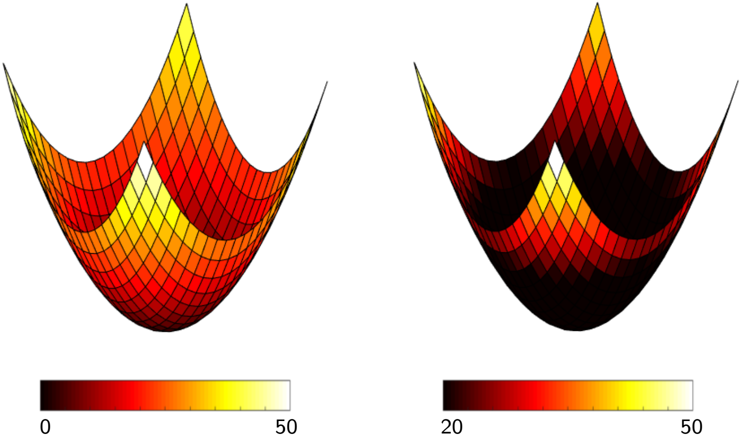
Decreasing the max limit



For smooth colormaps, the effect is to saturate the colors and make variations more evident

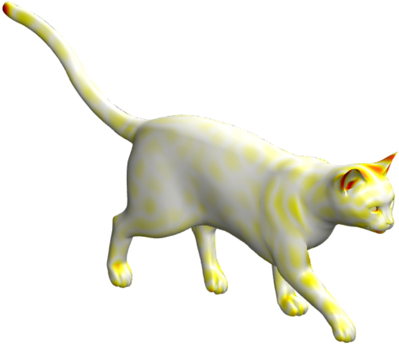
Example: Visualizing maxima

Increasing the min limit can be useful for visualizing maxima

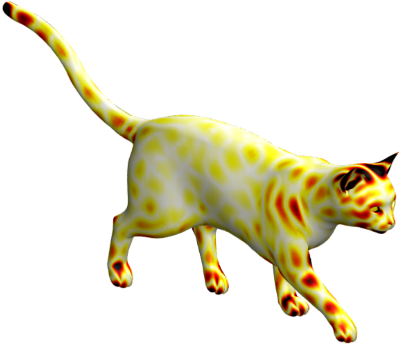


Example: Visualizing point-wise error

Emphasize areas of large error:



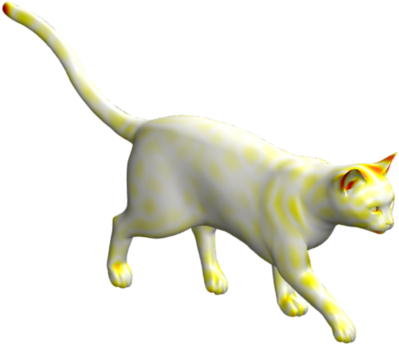
no saturation



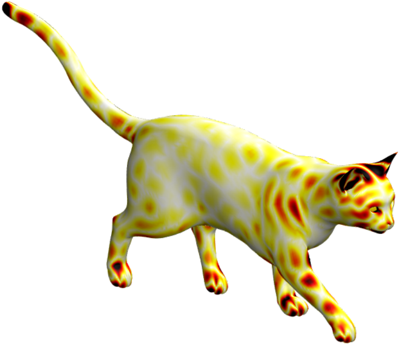
saturate at 0.5

Example: Visualizing point-wise error

Emphasize areas of large error:



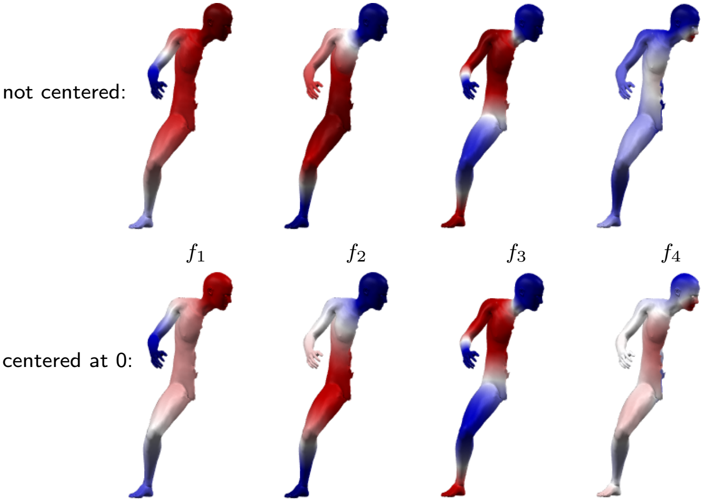
no saturation



saturate at 0.5

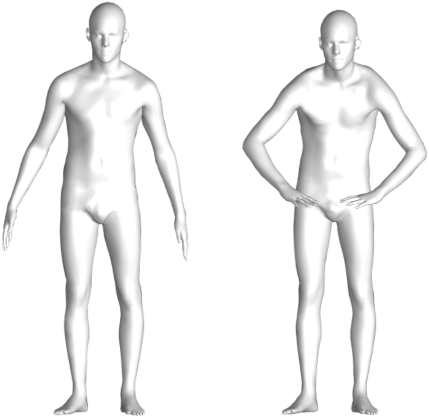
Example: Zero-centered functions

If $f :\to [-1,1]$ but f is not surjective, it can be useful to recenter colors



Coloring shapes

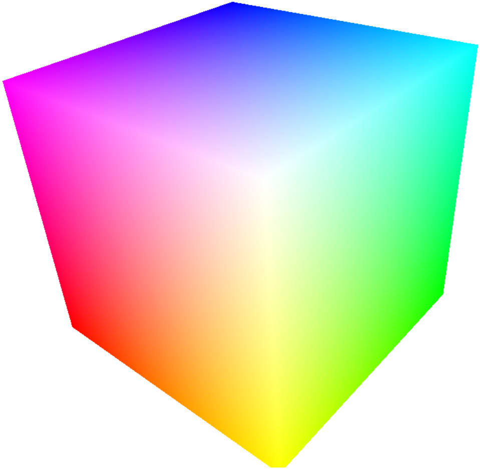
It is often useful to paint our surfaces



We may assign a color to each vertex, and interpolate between neighbors

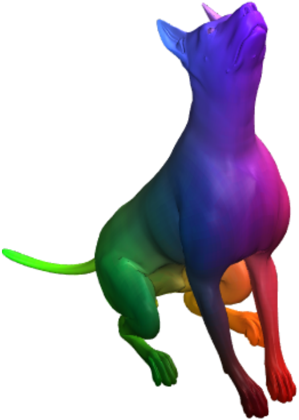
Color cube

Given a color space (here represented by the RGB color cube), we can "carve out" our shapes out of this cube



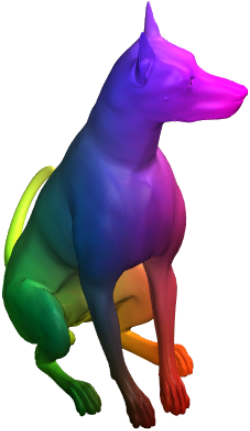
Coloring shapes

Given a color space (here represented by the RGB color cube), we can "carve out" our shapes out of this cube



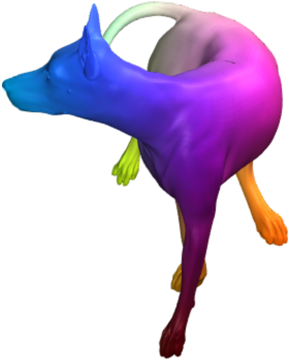
Coloring shapes

Given a color space (here represented by the RGB color cube), we can “carve out” our shapes out of this cube



Coloring shapes

Given a color space (here represented by the RGB color cube), we can "carve out" our shapes out of this cube



Coloring shapes

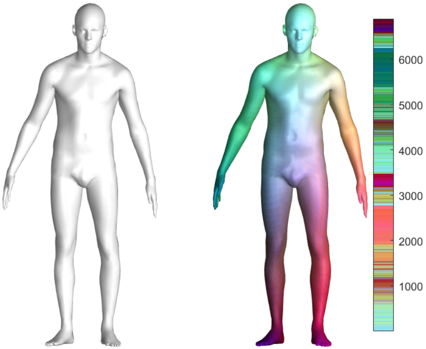
This is easily done by interpreting (x, y, z) as (R, G, B)

- Create a colormap with one color for each point ($n \times 3$ matrix)

Coloring shapes

This is easily done by interpreting (x, y, z) as (R, G, B)

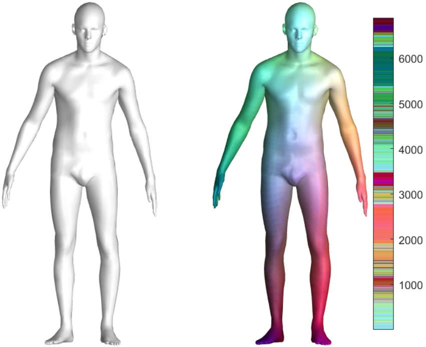
- Create a colormap with one color for each point ($n \times 3$ matrix)
- Plot the vector $(1, 2, \ldots, n)$



Coloring shapes

This is easily done by interpreting (x, y, z) as (R, G, B)

- Create a colormap with one color for each point ($n \times 3$ matrix)
- Plot the vector $(1, 2, \ldots, n)$



Warning: New versions of Matlab require shading flat (not interp)