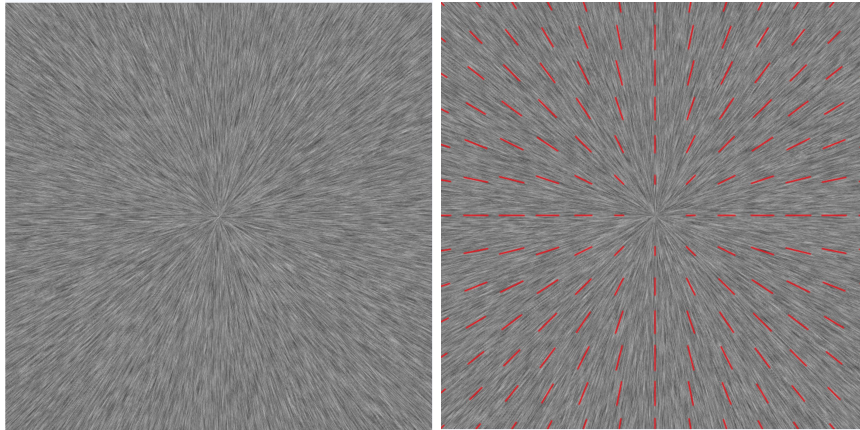


CS 519 Scientific Visualization

Machine Problem 2: Due 10/18/2017 at 11:55pm

Line Integral Convolution (LIC) and a HedgeHog Plot



Programming Language and Tools

For this MP, use HTML5 and JavaScript. If you wish, you can use WebGL, although you can accomplish the MPs just using the HTML5 *canvas* element as you have done in class.

The data set for this MP will be given by the gradient of the Gaussian function you worked with in MP1.

Line Integral Convolution (LIC) and a HedgeHog Plot

You will implement LIC as described in section 6.6 of *Data Visualization: Principles and Practice* by Telea. You can also refer to wikipedia article http://en.wikipedia.org/wiki/Line_integral_convolution

You will also implement a simple hedgehog plot capability. For this, you should allow the user to sample the domain on a uniform grid or to generate N random sample points where N is the square of the grid size parameter. At each point, draw a line segment in the direction of the vector field with the line length scaled by a user specified factor k .

Implementation Requirements:

1. To generate the random noise texture, you can just use `Math.random` and suitable scaling to generate a gray value for each pixel.
2. It is suggested you use the Gaussian function $k(s) = e^{-s^2}$ as the weighting function where s is some measure of distance from the source pixel. However you may use any weighting function that results in images similar to those shown above.

3. Your user interface should allow the user to pick a value L for how far to trace the streamline forward and backward from the source pixel. You will get better results if your weight function reaches 0 or close to it at distance L .
4. Your code should work on the gradient of the Gaussian we have worked with in class. Draw the hedgehog lines in red over the LIC image.
5. In addition to grayscale, offer users the ability to color the pixels according to the magnitude of the vector at the point. You can use whatever colormap you wish (including rainbow). Your user interface should allow users to turn this feature on and off. Keep in mind that you should modulate the luminance of the color according to the LIC value.
6. Your code should consist of the following files:
LIC.html
LIC.js
You may base your code off of the code we have worked on in class.

Submission

Submission will be through Compass. You are required to submit the following:

1. Your source code
2. An image demonstrating LIC applied to the gradient of the Gaussian function.
3. An image demonstrating a hedgehog plot of the gradient of the Gaussian function.
4. An image demonstrating LIC colored by magnitude.

Grading

The MP is worth 15% of your total course grade. It will be graded out of 15 points as follows:

- 1 points: Correct computation of the gradient of the Gaussian function
- 4 points: Correct implementation of hedgehog plotting
- 4 points: Correct implementation of LIC
- 4 points: Correct implementation of coloring by magnitude
- 2 points: Providing a reasonable user interface with the following parameters:
 - L : the LIC line length
 - Color on/off: color according to magnitude
 - k : the hedgehog line scaling factor
 - grid size: determines number of samples for the hedgehog plot
 - Hedgehog on/off: toggle on or off the hedgehog lines