KTH Stockholm
CSC :: CST
Visualization, Autumn 2017, DD2257
Tino Weinkauf

Homework assignment No. 06

Due October 12, 2017

Download the zip-File with the code and place the contents in the same folder as the previous modules. Follow the same process to activate the `IVW_MODULE_DD2257LAB4` module in CMake and build Inviwo. The files `licprocessor.h/cpp`, `noisetexturegenerator.h/cpp` and `integrator.h/cpp` contain additional comments and instructions.

Task 6.1: Line Integral Convolution — 20 P

Implement the Line Integral Convolution algorithm. See also Figure 1. Each of the following features gives a certain number of points, summing up to the total of 20 points:

(a) Random texture generation for textures of user-defined size (number of pixels) in x-direction and y-direction. (1 P)

(b) Base algorithm creating a LIC texture from a loaded vector field using either a randomly generated texture or a loaded image as the texture. You can use the box kernel for the convolution. (8 P)

(c) FastLIC. (5 P)

(d) User-defined kernel size. (2 P)

(e) Contrast enhancement after the convolution by adjusting the mean and standard deviation. (2 P)

(f) Ability to choose between a grayscale texture and a black-white texture for the randomly generated texture. (1 P)

(g) User-defined seed of the random number generator in order to create the same random texture with each run. (1 P)

The assignment comes with several vector fields for testing and Line Integral Convolution workspaces with randomly generated textures (`LIC_randomTexture.inv`) and loaded images (`LIC_loadImage.inv`). You may use other images then the ones provided.

Task 6.2: *(Bonus)* Line Integral Convolution ++ — 5 BP

Color the LIC texture according to the magnitude of the vector field.

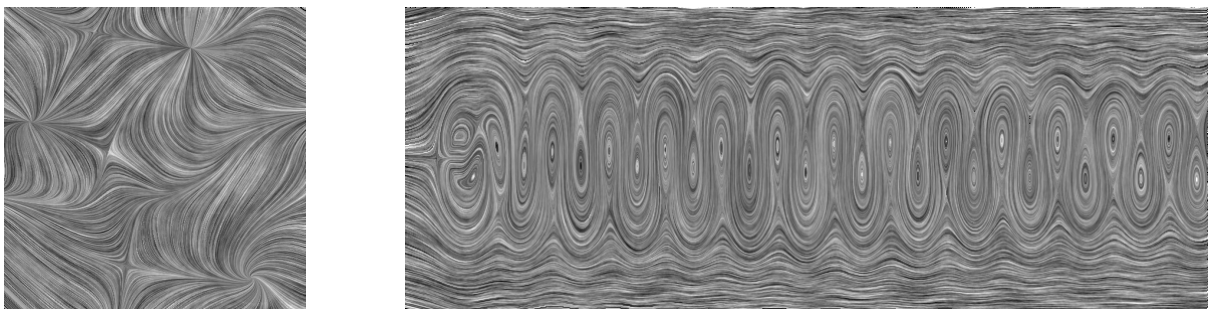

Figure 1: Examples for Line Integral Convolution