

1 Current setup

1.1 `vigra.filters.gaussianSmoothing(img, sigma)`

- separated x-,y- convolution
- kernel is initialized from scratch for every call
- kernel element: $k[i] = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{i^2}{2\sigma^2}}$
- kernel radius: $r = \text{round}(3 * \sigma)$

1.2 `vigra.filters.laplacianOfGaussian(img, sigma)`

1.3 `vigra.filters.gaussianGradientMagnitude(img, sigma)`

1.4 `vigra.filters.gaussianSmoothing(img, sigma)` - `vigra.filters.gaussianSmoothing(img, 0.66*sigma)`

1.5 `vigra.filters.structureTensorEigenvalues(img, sigma, sigma/2.0)`

1.6 `vigra.filters.hessianOfGaussianEigenvalues(img, sigma)`

2 HiWi HCI Nots

2.1 26.01.2016 (3h)

Base on CUDA samples. `separableConvolution` and `textureConvolution` are already implemented.

2.1.1 snippets

Listing 1: "print out gaussian 1D kernel"

```
import vigra

sigma = 3

kernel = vigra.filters.Kernel1D()
kernel.initGaussian(sigma)

for i in range(kernel.left(), kernel.right()+1):
    print kernel[i]
```

2.1.2 Addresses

- bigheron - GPU system
- hciwiki.iwr.uni-heidelberg.de

2.2 22.01.2016 (3h)

2.2.1 Ideas

- Vigna efficiency improvements
- CUDA/OpenCL (CUDA samples!)
- OpenGL
- FFT approach

2.2.2 Commands

```
# source miniconda2/bin/activate ilastik-devel \\
# source miniconda2/bin/activate root \\
# conda create -n ilastik-devel -c ilastik ilastik-everything-but-tracking
```